

C++

AND ALGORITHMIC THINKING

FOR THE COMPLETE BEGINNER

Second Edition

The Answers

Aristides S. Bouras

Copyright ©

Copyright © by Aristides S. Bouras
<https://www.bouraspage.com>

RCode: 221008

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, mechanical or electronic, including photocopying, recording, or by any information storage and retrieval system, without written permission from the authors.

Warning and Disclaimer

This book is designed to provide the answers to all of the review questions, as well as the solutions to all review exercises of the book "C++ AND ALGORITHMIC THINKING FOR THE COMPLETE BEGINNER – Second Edition". Every effort has been taken to make this book compatible with all releases of C++, and it is almost certain to be compatible with any future releases of it.

The information is provided on an "as is" basis. The authors shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the files that may accompany it.

Table of Contents

How to Report Errata.....	7
Chapter 1.....	8
1.7 Review Questions: True/False.....	8
1.8 Review Questions: Multiple Choice.....	8
Review in “Introductory Knowledge”	9
Review Crossword Puzzles	9
Chapter 4.....	11
4.16 Review Questions: True/False.....	11
4.17 Review Questions: Multiple Choice.....	11
Chapter 5.....	12
5.8 Review Questions: True/False.....	12
5.9 Review Questions: Multiple Choice.....	12
5.10 Review Exercises	12
Chapter 6.....	13
6.4 Review Questions: True/False.....	13
6.5 Review Questions: Multiple Choice.....	13
Chapter 7.....	14
7.7 Review Questions: True/False.....	14
7.8 Review Questions: Multiple Choice.....	14
7.9 Review Exercises	14
Chapter 8.....	15
8.2 Review Questions: True/False.....	15
8.3 Review Exercises	15
Chapter 9.....	17
9.4 Review Exercises	17
Review in “Getting Started with C#”	19
Review Crossword Puzzles	19
Chapter 10	20
10.2 Review Exercises	20
Chapter 11	27
11.3 Review Questions: True/False.....	27
11.4 Review Questions: Multiple Choice.....	27
11.5 Review Exercises	27

Chapter 12	30
12.2 Review Exercises	30
Chapter 13	33
13.2 Review Exercises	33
Chapter 14	36
14.4 Review Questions: True/False	36
14.5 Review Questions: Multiple Choice	36
14.6 Review Exercises	37
Review in “Sequence Control Structures”	40
Review Crossword Puzzle	40
Chapter 15	41
15.9 Review Questions: True/False	41
15.10 Review Questions: Multiple Choice	41
15.11 Review Exercises	41
Chapter 16	43
16.2 Review Questions: True/False	43
16.3 Review Questions: Multiple Choice	43
16.4 Review Exercises	43
Chapter 17	50
17.2 Review Questions: True/False	50
17.3 Review Questions: Multiple Choice	50
17.4 Review Exercises	50
Chapter 18	59
18.2 Review Questions: True/False	59
18.3 Review Exercises	59
Chapter 19	69
19.2 Review Questions: True/False	69
19.3 Review Exercises	69
Chapter 20	77
20.3 Review Questions: True/False	77
20.4 Review Exercises	77
Chapter 21	83
21.4 Review Exercises	83
Chapter 22	90
22.9 Review Questions: True/False	90

22.10 Review Questions: Multiple Choice	90
22.11 Review Exercises.....	91
Chapter 23	95
23.6 Review Exercises	95
Review in “Decision Control Structures”	116
Review Crossword Puzzle.....	116
Chapter 24	117
24.3 Review Questions: True/False.....	117
Chapter 25	118
25.4 Review Questions: True/False.....	118
25.5 Review Questions: Multiple Choice.....	118
25.6 Review Exercises	118
Chapter 26	131
26.3 Review Questions: True/False.....	131
26.4 Review Questions: Multiple Choice.....	131
26.5 Review Exercises	131
Chapter 27	147
27.3 Review Questions: True/False.....	147
27.4 Review Questions: Multiple Choice.....	147
27.5 Review Exercises	147
Chapter 28	156
28.8 Review Questions: True/False.....	156
28.9 Review Questions: Multiple Choice.....	156
28.10 Review Exercises.....	156
Chapter 29	162
29.4 Review Exercises	162
Chapter 30	171
30.7 Review Questions: True/False.....	171
30.8 Review Exercises	171
Review in “Loop Control Structures”	197
Review Crossword Puzzle	197
Chapter 31	198
31.13 Review Questions: True/False	198
31.14 Review Questions: Multiple Choice	198
31.15 Review Exercises.....	198

Chapter 32	213
32.7 Review Questions: True/False	213
32.8 Review Questions: Multiple Choice	213
32.9 Review Exercises	214
Chapter 33	227
33.7 Review Questions: True/False	227
33.8 Review Questions: Multiple Choice	227
33.9 Review Exercises	227
Chapter 34	238
34.7 Review Questions: True/False	238
34.8 Review Exercises	239
Review in “Data Structures in C#”	277
Review Crossword Puzzle	277
Chapter 35	278
35.4 Review Questions: True/False	278
Chapter 36	279
36.8 Review Questions: True/False	279
36.9 Review Exercises	279
Chapter 37	292
37.10 Review Questions: True/False	292
37.11 Review Exercises	292
Chapter 38	301
38.3 Review Exercises	301
Review in “Subprograms”	318
Review Crossword Puzzle	318
Chapter 39	319
39.8 Review Questions: True/False	319
39.9 Review Exercises	319
Review in “Object Oriented Programming”	333
Review Crossword Puzzle	333
Some Final Words from the Author	334

How to Report Errata

Although I have taken great care to ensure the accuracy of the content of this book, mistakes do occur. If you find a mistake in this book, either in the text or the code, I encourage you to report it to me. By doing so, you can save other readers from frustration and, of course, help me to improve the next release of this book. If you find any errata, please feel free to report them by visiting the following address:

<https://www.bouraspage.com/report-errata>

Once your errata are verified, your submission will be accepted and the errata will be uploaded to my website, and added to any existing list of errata.

Chapter 1

1.7 Review Questions: True/False

- | | |
|----------|-----------|
| 1. false | 12. false |
| 2. false | 13. false |
| 3. true | 14. false |
| 4. false | 15. true |
| 5. false | 16. true |
| 6. true | 17. false |
| 7. true | 18. false |
| 8. false | 19. true |
| 9. false | 20. false |
| 10. true | 21. false |
| 11. true | 22. true |

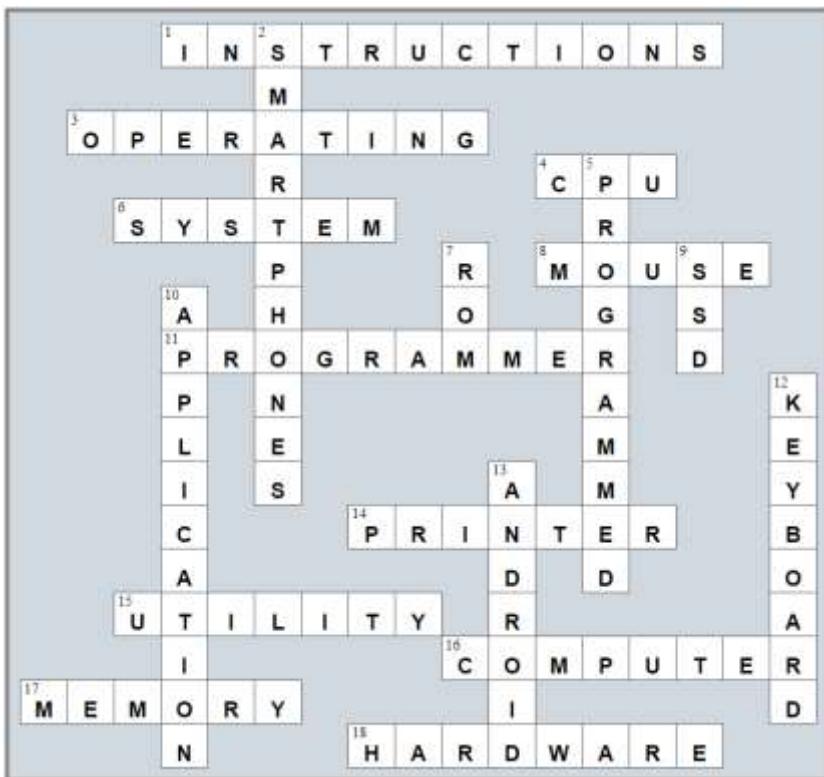
1.8 Review Questions: Multiple Choice

- | | |
|------|-------|
| 1. b | 7. c |
| 2. d | 8. b |
| 3. b | 9. c |
| 4. c | 10. b |
| 5. f | 11. a |
| 6. d | |

Review in "Introductory Knowledge"

Review Crossword Puzzles

1.



2.



3.



Chapter 4

4.16 Review Questions: True/False

- | | |
|-----------|-----------|
| 1. true | 22. true |
| 2. false | 23. false |
| 3. false | 24. false |
| 4. false | 25. false |
| 5. false | 26. false |
| 6. true | 27. true |
| 7. false | 28. true |
| 8. true | 29. false |
| 9. true | 30. false |
| 10. true | 31. false |
| 11. false | 32. false |
| 12. false | 33. false |
| 13. true | 34. true |
| 14. true | 35. false |
| 15. false | 36. false |
| 16. true | 37. false |
| 17. false | 38. false |
| 18. false | 39. true |
| 19. false | 40. true |
| 20. true | 41. false |
| 21. true | |

4.17 Review Questions: Multiple Choice

- | | |
|------|-------|
| 1. c | 6. a |
| 2. b | 7. b |
| 3. c | 8. d |
| 4. a | 9. a |
| 5. a | 10. d |

Chapter 5

5.8 Review Questions: True/False

- | | |
|----------|-----------|
| 1. false | 10. false |
| 2. false | 11. true |
| 3. true | 12. false |
| 4. false | 13. true |
| 5. false | 14. true |
| 6. true | 15. true |
| 7. false | 16. true |
| 8. false | 17. false |
| 9. true | |

5.9 Review Questions: Multiple Choice

- | | |
|------|------|
| 1. e | 5. c |
| 2. a | 6. c |
| 3. d | 7. d |
| 4. b | |

5.10 Review Exercises

- 1 - c, 2 - d, 3 - a, 4 - b
- 1 - d, 2 - c, 3 - b, 4 - a
- 3.

Value	Data Type	Declaration and Initialization
The name of my friend	String	string name = "Mark";
My address	String	string address = "254 Lookout Rd. Wilson, NY 27893";
The average daily temperature	Float	double average = 70.3;
A telephone number	String	string phone_number = "1-891-764-2410";
My Social Security Number (SSN)	String	string ssn = "123-45-6789";
The speed of a car	Float	double speed = 90.5;
The number of children in a family	Integer	int children = 3;

Chapter 6

6.4 Review Questions: True/False

1. true
2. true
3. true
4. false
5. false

6.5 Review Questions: Multiple Choice

1. a
2. c
3. b
4. b

Chapter 7

7.7 Review Questions: True/False

- | | |
|-----------|-----------|
| 1. false | 14. false |
| 2. true | 15. false |
| 3. false | 16. true |
| 4. false | 17. false |
| 5. false | 18. true |
| 6. false | 19. false |
| 7. false | 20. false |
| 8. false | 21. false |
| 9. true | 22. true |
| 10. false | 23. false |
| 11. false | 24. false |
| 12. true | |
| 13. false | |

7.8 Review Questions: Multiple Choice

- | | | |
|------|------|------|
| 1. c | 4. d | 7. d |
| 2. c | 5. b | 8. c |
| 3. b | 6. d | |

7.9 Review Exercises

1. ii, iv, v, ix, x
2. i. String, ii. Boolean, iii. String, iv. String, v. Float, vi. Integer
3. i. d, ii. f, iii. c, iv. e
4. i. 26, ii. 28
5. i. 5, ii. 6
6. i. 1, ii. 0, iii. 1, iv. 1, v. 0, vi. 1
7. i. $2 * 3$, ii. 4
8. i. 2, ii. 0, iii. 1, iv. 0, v. 0, vi. 0
9. i. 2, ii. 10
10. My name is George Malkovich
11. i. (-3) , ii. 1
12. California California

Chapter 8

8.2 Review Questions: True/False

- | | |
|----------|----------|
| 1. false | 3. false |
| 2. true | 4. false |

8.3 Review Exercises

1. Solution

For the input value of 3

Step	Statement	a	b	c	d
1	<code>cin >> a</code>	3	?	?	?
2	<code>a = (a + 1) * (a + 1) + 6 / 3 * 2 + 20</code>	40	?	?	?
3	<code>b = a % 13</code>	40	1	?	?
4	<code>c = b % 7</code>	40	1	1	?
5	<code>d = a * b * c</code>	40	1	1	40
6	<code>cout << a << ", " << b << ", " << c << ", " << d << endl</code>	It displays: 40, 1, 1, 40			

For the input value of 4

Step	Statement	a	b	c	d
1	<code>cin >> a</code>	4	?	?	?
2	<code>a = (a + 1) * (a + 1) + 6 / 3 * 2 + 20</code>	49	?	?	?
3	<code>b = a % 13</code>	49	10	?	?
4	<code>c = b % 7</code>	49	10	3	?
5	<code>d = a * b * c</code>	49	10	3	1470
6	<code>cout << a << ", " << b << ", " << c << ", " << d << endl</code>	It displays: 49, 10, 3, 1470			

For the input value of 1

Step	Statement	a	b	c	d
1	<code>cin >> a</code>	1	?	?	?
2	<code>a = (a + 1) * (a + 1) + 6 / 3 * 2 + 20</code>	28	?	?	?
3	<code>b = a % 13</code>	28	2	?	?
4	<code>c = b % 7</code>	28	2	2	?
5	<code>d = a * b * c</code>	28	2	2	112
6	<code>cout << a << ", " << b << ", " << c << ", " << d << endl</code>	It displays: 28, 2, 2, 112			

2. Solution

For the input values of 8, 4

Step	Statement	a	b	c	d	e
1	<code>cin >> a</code>	8	?	?	?	?
2	<code>cin >> b</code>	8	4	?	?	?

3	c = a + b	8	4	12	?	?
4	d = 1 + a / b * c + 2	8	4	12	27	?
5	e = c + d	8	4	12	27	39
6	c += d + e	8	4	78	27	39
7	e--	8	4	78	27	38
8	d -= c + d % c	8	4	78	-78	38
9	cout << c << ", " << d << ", " << e << endl	It displays: 78, -78, 38				

For the input values of 4, 4

Step	Statement	a	b	c	d	e
1	cin >> a	4	?	?	?	?
2	cin >> b	4	4	?	?	?
3	c = a + b	4	4	8	?	?
4	d = 1 + a / b * c + 2	4	4	8	11	?
5	e = c + d	4	4	8	11	19
6	c += d + e	4	4	38	11	19
7	e--	4	4	38	11	18
8	d -= c + d % c	4	4	38	-38	18
9	cout << c << ", " << d << ", " << e << endl	It displays: 38, -38, 18				

Chapter 9

9.4 Review Exercises

1. Solution

The statement `S = S1 + S3 + SS` is wrong. It must be `S = S1 + S3 + S5`

2. Solution

For the input values of 5, 5

Step	Statement	a	b	c	d	e
1	<code>cin >> a</code>	5	?	?	?	?
2	<code>cin >> b</code>	5	5	?	?	?
3	<code>c = a + b</code>	5	5	10	?	?
4	<code>d = 5 + a / b * c + 2</code>	5	5	10	17	?
5	<code>e = c - d</code>	5	5	10	17	-7
6	<code>c += d + c</code>	5	5	37	17	-7
7	<code>e--</code>	5	5	37	17	-8
8	<code>d += e + c % b</code>	5	5	37	11	-8
9	<code>cout << c << ", " << d << ", " << e << endl</code>	It displays: 37, 11, -8				

For the input values of 4, 2

Step	Statement	a	b	c	d	e
1	<code>cin >> a</code>	4	?	?	?	?
2	<code>cin >> b</code>	4	2	?	?	?
3	<code>c = a + b</code>	4	2	6	?	?
4	<code>d = 5 + a / b * c + 2</code>	4	2	6	19	?
5	<code>e = c - d</code>	4	2	6	19	-13
6	<code>c += d + c</code>	4	2	31	19	-13
7	<code>e--</code>	4	2	31	19	-14
8	<code>d += e + c % b</code>	4	2	31	6	-14
9	<code>cout << c << ", " << d << ", " << e << endl</code>	It displays: 31, 6, -14				

3. Solution

For the input value of 5

Step	Statement	a	b	c
1	<code>cin >> b</code>	?	5	?
2	<code>c = 5</code>	?	5	5
3	<code>c = c * b</code>	?	5	25
4	<code>a = 3 * c % 10</code>	5	5	25

5	cout << a << endl	It displays: 5
---	-------------------	----------------

For the input value of 4

Step	Statement	a	b	c
1	cin >> b	?	4	?
2	c = 5	?	4	5
3	c = c * b	?	4	20
4	a = 3 * c % 10	0	4	20
5	cout << a << endl	It displays: 0		

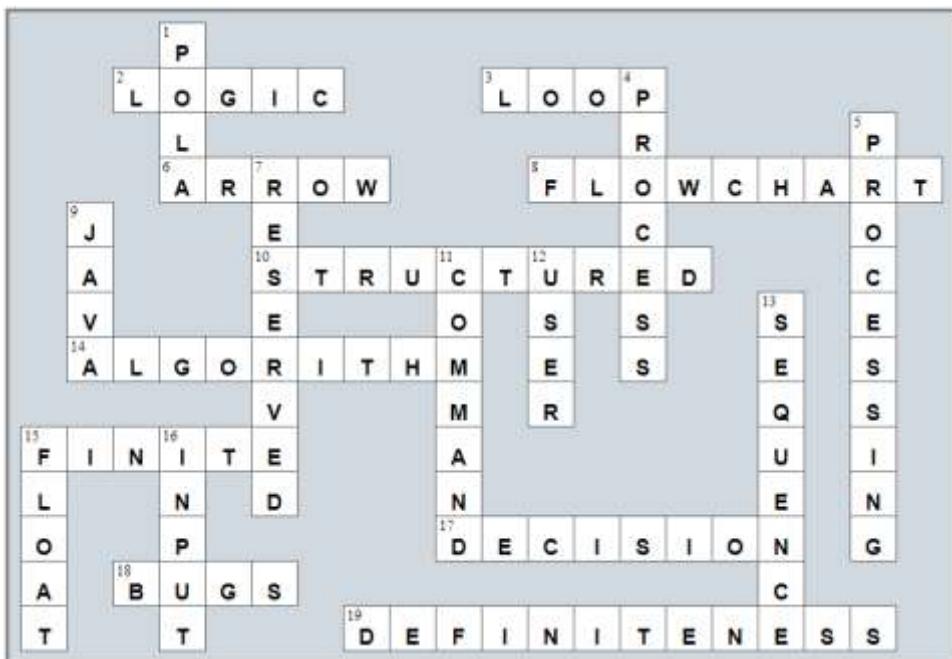
For the input value of 15

Step	Statement	a	b	c
1	cin >> b	?	15	?
2	c = 5	?	15	5
3	c = c * b	?	15	75
4	a = 3 * c % 10	5	15	75
5	cout << a << endl	It displays: 5		

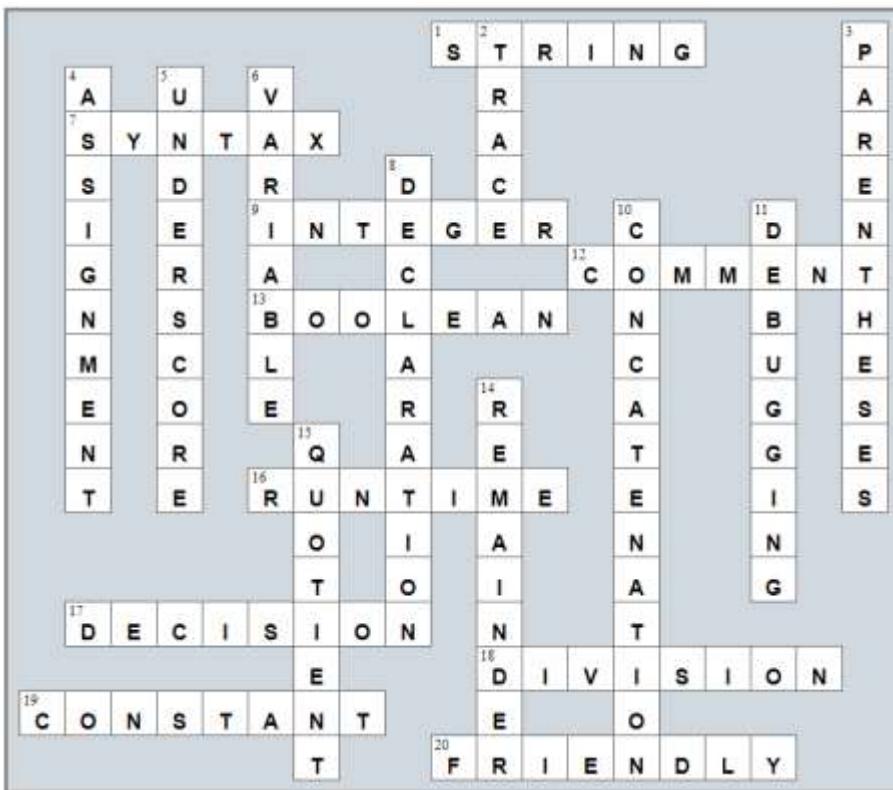
Review in “Getting Started with C#”

Review Crossword Puzzles

1.



2.



Chapter 10

10.2 Review Exercises

1. Solution

```
#include <iostream>
using namespace std;
int main() {
    double b, h, area;

    cout << "Enter base: ";
    cin >> b;
    cout << "Enter height: ";
    cin >> h;

    area = 0.5 * b * h;

    cout << area << endl;
    return 0;
}
```

2. Solution

```
#include <iostream>
using namespace std;
int main() {
    double angle1, angle2, angle3;

    cout << "Enter 1st angle: ";
    cin >> angle1;
    cout << "Enter 2nd angle: ";
    cin >> angle2;

    angle3 = 180 - angle1 - angle2;

    cout << angle3 << endl;
    return 0;
}
```

3. Solution

```
#include <iostream>
using namespace std;
int main() {
    int g1, g2, g3, g4;
    double average;

    cout << "Enter 1st grade: ";
    cin >> g1;
    cout << "Enter 2nd grade: ";
    cin >> g2;
```

```
    cout << "Enter 3rd grade: ";
    cin >> g3;
    cout << "Enter 4th grade: ";
    cin >> g4;

    average = (g1 + g2 + g3 + g4) / 4.0;

    cout << average << endl;
    return 0;
}
```

4. Solution

```
#include <iostream>
using namespace std;
const double PI = 3.14159;

int main() {
    double r, perimeter;

    cout << "Enter radius: ";
    cin >> r;

    perimeter = 2 * PI * r;

    cout << perimeter << endl;
    return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
const double PI = 3.14159;

int main() {
    double d, radius, volume;

    cout << "Enter diameter (in meters): ";
    cin >> d;

    radius = d / 2;
    volume = 4 / 3 * PI * radius * radius * radius;

    cout << volume << endl;
    return 0;
}
```

6. Solution

Only a), e) and g) are syntactically correct. The latter is more user friendly.

7. Solution

```
#include <iostream>
using namespace std;
const double PI = 3.14159;

int main() {
    double d, radius, perimeter, area, volume;

    cout << "Enter diameter: ";
    cin >> d;

    radius = d / 2;
    perimeter = 2 * PI * radius;
    area = PI * radius * radius;
    volume = 4 / 3 * PI * radius * radius * radius;

    cout << radius << " " << perimeter << " " << area << " " << volume << endl;
    return 0;
}
```

8. Solution

```
#include <iostream>
using namespace std;
int main() {
    double charge, tip, tax, total;

    cout << "Enter charge for a meal: ";
    cin >> charge;

    tip = charge * 10 / 100;
    tax = charge * 7 / 100;

    total = charge + tip + tax;

    cout << total << endl;
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, t, s;

    cout << "Enter acceleration in m/sec2: ";
    cin >> a;
    cout << "Enter time traveled in sec: ";
    cin >> t;

    s = 0.5 * a * t * t;
```

```
    cout << s << endl;
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
int main() {
    double f, c;

    cout << "Enter temperature in Fahrenheit: ";
    cin >> f;

    c = 5 / 9 * (f - 32);

    cout << c << endl;
    return 0;
}
```

11. Solution

```
#include <iostream>
using namespace std;
int main() {
    int w, h;
    double bmi;

    cout << "Enter weight in pounds: ";
    cin >> w;
    cout << "Enter height in inches: ";
    cin >> h;

    bmi = w * 703.0 / (h * h);

    cout << bmi << endl;
    return 0;
}
```

12. Solution

```
#include <iostream>
using namespace std;
int main() {
    double s_total, g_rate, tip, total;

    cout << "Enter subtotal: ";
    cin >> s_total;
    cout << "Enter gratuity rate (0 - 100): ";
    cin >> g_rate;

    tip = s_total * g_rate / 100;

    total = s_total + tip;
```

```
    cout << "Tip is $" << tip << " and total is $" << total << endl;
    return 0;
}
```

13. Solution

```
#include <iostream>
using namespace std;
const double VAT = 0.20;

int main() {
    double btax_price1, btax_price2, btax_price3, atax_price1, atax_price2, atax_price3, avg;

    cout << "Enter before-tax price 1: ";
    cin >> btax_price1;
    cout << "Enter before-tax price 2: ";
    cin >> btax_price2;
    cout << "Enter before-tax price 3: ";
    cin >> btax_price3;

    atax_price1 = btax_price1 + btax_price1 * VAT;
    atax_price2 = btax_price2 + btax_price2 * VAT;
    atax_price3 = btax_price3 + btax_price3 * VAT;

    avg = (atax_price1 + atax_price2 + atax_price3) / 3;

    cout << avg << endl;
    return 0;
}
```

14. Solution

```
#include <iostream>
using namespace std;
const int VAT = 0.20;

int main() {
    double atax_price, btax_price;

    cout << "Enter after-tax price: ";
    cin >> atax_price;

    btax_price = atax_price / (1 + VAT);

    cout << btax_price << endl;
    return 0;
}
```

15. Solution

```
#include <iostream>
using namespace std;
int main() {
```

```
double i_price, discount, f_price, saved;

cout << "Enter price: ";
cin >> i_price;
cout << "Enter discount: ";
cin >> discount;

f_price = i_price - i_price * discount / 100;
saved = i_price - f_price;

cout << f_price << " " << saved << endl;
return 0;
}
```

16. Solution

```
#include <iostream>
using namespace std;
const int VAT = 0.20;

int main() {
    int i_kWh, f_kWh, kWh_consumed;
    double cost;

    cout << "Enter kWh at the beginning of the month: ";
    cin >> i_kWh;
    cout << "Enter kWh at the end of the month: ";
    cin >> f_kWh;

    kWh_consumed = f_kWh - i_kWh;

    cost = kWh_consumed * 0.06;
    cost += cost * VAT;

    cout << kWh_consumed << " " << cost << endl;
    return 0;
}
```

17. Solution

```
#include <iostream>
using namespace std;
int main() {
    int day, month, days_passed, days_left;

    cout << "Enter current month: ";
    cin >> month;
    cout << "Enter current day: ";
    cin >> day;

    days_passed = (month - 1) * 30 + day;
    days_left = 360 - days_passed;

    cout << days_left << endl;
```

```
| return 0;  
| }
```

Chapter 11

11.3 Review Questions: True/False

- | | | |
|----------|----------|-----------|
| 1. true | 5. false | 9. true |
| 2. false | 6. false | 10. true |
| 3. false | 7. true | 11. false |
| 4. false | 8. true | 12. false |

11.4 Review Questions: Multiple Choice

- | | | |
|------|------|------|
| 1. d | 3. b | 5. a |
| 2. d | 4. c | 6. b |

11.5 Review Exercises

1. Solution

For the input value of 9

Step	Statement	a	b	c
1	cin >> a	9.0	?	?
2	a += 6 / sqrt(a) * 2 + 20.4	33.4	?	?
3	b = round(a) % 4	33.4	1.0	?
4	c = b % 3	33.4	1.0	1.0
5	cout << a << ", " << b << ", " << c << endl	It displays: 33.4, 1, 1		

For the input value of 4

Step	Statement	a	b	c
1	cin >> a	4.0	?	?
2	a += 6 / sqrt(a) * 2 + 20.4	30.4	?	?
3	b = round(a) % 4	30.4	2.0	?
4	c = b % 3	30.4	2.0	2.0
5	cout << a << ", " << b << ", " << c << endl	It displays: 30.4, 2, 2		

2. Solution

For the input value of -2

Step	Statement	a	b	c
1	cin >> a	-2	?	?
2	b = abs(a) % 4 + pow(a, 4)	-2	18	?
3	c = b % 5	-2	18	3
4	cout << b << ", " << c << endl	It displays: 18, 3		

For the input value of -3

Step	Statement	a	b	c
1	cin >> a	-3	?	?
2	b = abs(a) % 4 + pow(a, 4)	-3	84	?
3	c = b % 5	-3	84	4
4	cout << b << ", " << c << endl	It displays: 84, 4		

3. Solution

```
#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double degrees, radians;

    cout << "Enter angle in radians: ";
    cin >> radians;

    degrees = radians * 180 / M_PI;

    cout << degrees << endl;
    return 0;
}
```

4. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double a, b, hypotenuse;

    cout << "Enter right angle side A of a right-angled triangle: ";
    cin >> a;
    cout << "Enter right angle side B of a right-angled triangle: ";
    cin >> b;

    hypotenuse = sqrt(pow(a, 2) + pow(b, 2));

    cout << hypotenuse << endl;
    return 0;
}
```

5. Solution

```
#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
using namespace std;
int main() {
```

```
double adjacent, opposite, th;  
  
cout << "Enter angle  $\theta$  (in degrees) of a right-angled triangle: ";  
cin >> th;  
cout << "Enter length of adjacent side: ";  
cin >> adjacent;  
  
opposite = tan(th * M_PI / 180) * adjacent;  
  
cout << opposite << endl;  
return 0;  
}
```

Chapter 12

12.2 Review Exercises

1. Solution

- i. a, e, g, h
- ii. c, f

2. Solution

- i. $y = \text{pow}(x + 3, 5 * w) / (7 * (x - 4))$
- ii. $y = \text{pow}(3 * \text{pow}(x, 2) - \text{pow}(x, 3) / 4, 1 / 5.0)$
- iii. $y = \text{sqrt}(\text{pow}(x, 4) - 2 * \text{pow}(x, 3) - 7 * x * x + x) / \text{pow}(4 * (7 * \text{pow}(x, 4) - 3 / 4.0 * \text{pow}(x, 3)) * (7 * x * x + x), 1 / 3.0)$
- iv. $y = x / (x - 3 * (x - 1)) + x * \text{pow}(x - 1, 1 / 5.0) / ((\text{pow}(x, 3) - 2) * \text{pow}(x - 1, 3))$
- v. $y = \text{pow}(\sin(M_PI / 3) - \cos(M_PI / 2 * w), 2)$
- vi. $y = \text{pow}(\sin(M_PI / 2 * x) + \cos(3 * M_PI / 2 * w), 3) / \text{pow}(\tan(2 * M_PI / 3 * w) - \sin(M_PI / 2 * x), 0.5) + 6$

3. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double x, y;

    cout << "Enter value for x: ";
    cin >> x;

    y = sqrt(x) * (pow(x, 3) + pow(x, 2));

    cout << y << endl;
    return 0;
}
```

4. Solution

```
#include <iostream>
using namespace std;
int main() {
    double x, y;

    cout << "Enter value for x: ";
    cin >> x;

    y = 7 * x / (2 * x + 4 * (x * x + 4));

    cout << y << endl;
    return 0;
}
```

5. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double w, x, y;

    cout << "Enter value for x: ";
    cin >> x;
    cout << "Enter value for w: ";
    cin >> w;

    y = pow(x, x + 1) / pow(tan(2 * w / 3 + 5) - tan(x / 2 + 1), 3);

    cout << y << endl;
    return 0;
}
```

6. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double w, x, y;

    cout << "Enter value for x: ";
    cin >> x;
    cout << "Enter value for w: ";
    cin >> w;

    y = (3 + w) / (6 * x - 7 * (x + 4)) + x * pow(3 * w + 1, 1 / 5) * (5 * x + 4) / ((pow(x, 3) + 3) *
pow(x - 1, 7));

    cout << y << endl;
    return 0;
}
```

7. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double w, x, y;

    cout << "Enter value for x: ";
    cin >> x;
    cout << "Enter value for w: ";
    cin >> w;
```

```
    y = pow(x, x) / pow(sin(2 * w / 3 + 5) - x, 2) + pow(sin(3 * x) + w, x + 1) / pow(sqrt(7 * w), 3 /
2);

    cout << y << endl;
    return 0;
}
```

8. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double a, b, c, area, semi;

    cout << "Enter length A: ";
    cin >> a;
    cout << "Enter length B: ";
    cin >> b;
    cout << "Enter length C: ";
    cin >> c;

    semi = (a + b + c) / 2;
    area = sqrt(semi * (semi - a) * (semi - b) * (semi - c));

    cout << area << endl;
    return 0;
}
```

Chapter 13

13.2 Review Exercises

1. Solution

```
#include <iostream>
using namespace std;
int main() {
    int last_digit, n, result;

    cout << "Enter an integer: ";
    cin >> n;

    last_digit = n % 10;
    result = last_digit * 8;

    cout << result << endl;
    return 0;
}
```

2. Solution

```
#include <iostream>
using namespace std;
int main() {
    int digit1, digit2, digit3, digit4, digit5, number, r, reversed_number;

    cout << "Enter a five-digit integer: ";
    cin >> number;

    digit5 = number % 10;
    r = (int)(number / 10);

    digit4 = r % 10;
    r = (int)(r / 10);

    digit3 = r % 10;
    r = (int)(r / 10);

    digit2 = r % 10;
    digit1 = (int)(r / 10);

    reversed_number = digit5 * 10000 + digit4 * 1000 + digit3 * 100 + digit2 * 10 + digit1;
    cout << reversed_number << endl;
    return 0;
}
```

3. Solution

```
#include <iostream>
using namespace std;
int main() {
```

```
int n, result;

cout << "Enter an integer: ";
cin >> n;

result = n % 2;

cout << result << endl;
return 0;
}
```

4. Solution

```
#include <iostream>
using namespace std;
int main() {
    int n, result;

    cout << "Enter an integer: ";
    cin >> n;

    result = 1 - n % 2;

    cout << result << endl;
    return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
int main() {
    int days, hours, minutes, number, r, seconds, weeks;

    cout << "Enter an elapsed time in seconds: ";
    cin >> number;

    weeks = (int)(number / 604800); // 60 * 60 * 24 * 7 = 604800
    r = number % 604800;

    days = (int)(r / 86400); // 60 * 60 * 24 = 86400
    r = r % 86400;

    hours = (int)(r / 3600);
    r = r % 3600;

    minutes = (int)(r / 60);
    seconds = r % 60;

    cout << weeks << " week(s) " << days << " day(s) " << hours << " hour(s) ";
    cout << minutes << " minute(s) and " << seconds << " second(s)" << endl;
    return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
int main() {
    int amount, r, usd1, usd10, usd20, usd5;

    cout << "Enter amount of money to withdraw: ";
    cin >> amount;

    usd20 = (int)(amount / 20);
    r = amount % 20;

    usd10 = (int)(r / 10);
    r = r % 10;

    usd5 = (int)(r / 5);
    usd1 = r % 5;

    cout << usd20 << " note(s) of $20 " << usd10 << " note(s) of $10 ";
    cout << usd5 << " note(s) of $5 and " << usd1 << " note(s) of $1" << endl;
    return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
int main() {
    int distance, feet, inches, miles, r, steps, yards;

    cout << "Enter number of steps: ";
    cin >> steps;

    distance = steps * 25;

    miles = (int)(distance / 63360);
    r = distance % 63360;

    yards = (int)(r / 36);
    r = r % 36;

    feet = (int)(r / 12);
    inches = r % 12;

    cout << miles << " mile(s) " << yards << " yard(s) ";
    cout << feet << " foot/feet and " << inches << " inch(es)" << endl;
    return 0;
}
```

Chapter 14

14.4 Review Questions: True/False

- | | | |
|----------|-----------|-----------|
| 1. true | 7. true | 13. false |
| 2. false | 8. false | 14. true |
| 3. false | 9. true | 15. true |
| 4. true | 10. false | 16. true |
| 5. true | 11. false | |
| 6. false | 12. true | |

14.5 Review Questions: Multiple Choice

1. d
2. b
3. a
4. d
5. b
6. b
7. c
8. a
9. c

14.6 Review Exercises

1. Solution

```
#include <iostream>
using namespace std;
int main() {
    string first_name, last_name, middle_name, title;

    cout << "First name: ";
    cin >> first_name;
    cout << "Middle name: ";
    cin >> middle_name;
    cout << "Last name: ";
    cin >> last_name;
    cout << "Title: ";
    cin >> title;

    cout << title << " " << first_name << " " << middle_name << " " << last_name << endl;
    cout << first_name << " " << middle_name << " " << last_name << endl;
    cout << last_name << ", " << first_name << endl;
    cout << last_name << ", " << first_name << " " << middle_name << endl;
    cout << last_name << ", " << first_name << " " << middle_name << ", " << title << endl;
    cout << first_name << " " << last_name << endl;
    return 0;
}
```

2. Solution

```
#include <iostream>
#include <ctime>
#include <cstdlib>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string alphabet_lower, alphabet_upper;

    srand(time(NULL));

    alphabet_lower = "abcdefghijklmnopqrstuvwxyz";
    alphabet_upper = to_upper_copy(alphabet_lower);

    cout << alphabet_upper[rand() % 26] <<
        alphabet_lower[rand() % 26] <<
        alphabet_lower[rand() % 26] <<
        alphabet_lower[rand() % 26] <<
        alphabet_lower[rand() % 26];
    return 0;
}
```

3. Solution

```
#include <iostream>
#include <ctime>
#include <cstdlib>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string name, x;
    srand(time(NULL));

    cout << "Enter name: ";
    cin >> name;

    x = replace_all_copy(to_lower_copy(name), " ", "");

    cout << x[rand() % x.length()] <<
        x[rand() % x.length()] <<
        x[rand() % x.length()] <<
        (1000 + rand() % (9999 - 1000 + 1));

    return 0;
}
```

4. Solution

First Approach

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    int number, reversed_number;
    string s_number, digit1, digit2, digit3;

    cout << "Enter a three-digit integer: ";
    cin >> number;

    s_number = to_string(number);

    digit1 = s_number[0];
    digit2 = s_number[1];
    digit3 = s_number[2];

    reversed_number = 100 * stoi(digit3) + 10 * stoi(digit2) + stoi(digit1);

    cout << reversed_number << endl;
    return 0;
}
```

Second Approach

```
#include <iostream>
#include <string>
using namespace std;
```

```
int main() {
    int number, reversed_number;
    string s_number, digit1, digit2, digit3;

    cout << "Enter a three-digit integer: ";
    cin >> number;

    s_number = to_string(number);

    digit1 = s_number[0];
    digit2 = s_number[1];
    digit3 = s_number[2];

    reversed_number = stoi(digit3 + digit2 + digit1);

    cout << reversed_number << endl;
    return 0;
}
```

Review in "Sequence Control Structures"

Review Crossword Puzzle

1.



Chapter 15

15.9 Review Questions: True/False

- | | | |
|----------|-----------|-----------|
| 1. true | 9. true | 17. false |
| 2. false | 10. true | 18. true |
| 3. false | 11. true | 19. true |
| 4. false | 12. true | 20. false |
| 5. false | 13. true | 21. true |
| 6. false | 14. true | 22. true |
| 7. false | 15. true | 23. true |
| 8. true | 16. false | |

15.10 Review Questions: Multiple Choice

- | | | |
|------|------|------|
| 1. b | 3. a | 5. c |
| 2. a | 4. a | 6. d |

15.11 Review Exercises

1. Solution

- i. c, e, g
- ii. a, j
- iii. d, f
- iv. b, h, i

2. Solution

a	b	c	a != 1	b > a	c / 2 > 2 * a
3	-5	8	true	false	false
1	10	20	false	true	true
-4	-2	-9	true	true	true

3. Solution

Boolean Expression1 (BE1)	Boolean Expression2 (BE2)	BE1 BE2	BE1 && BE2	!(BE2)
false	false	false	false	true
false	true	true	false	false
true	false	true	false	true
true	true	true	true	false

4. Solution

a	b	c	a > 3 c > b && c > 1	a > 3 && c > b c > 1
4	-6	2	true	true

-3	2	-4	false	false
2	5	5	false	true

5. Solution

Expression	Value
<code>pow(x + y, 3)</code>	8
<code>(x + y) / (pow(x, 2) - 14)</code>	1
<code>x - 1 == y + 5</code>	true
<code>x > 2 && y == 1</code>	false
<code>x == 1 y == -2 && !(flag == false)</code>	true
<code>!(x >= 3) && (x % 2 > 1)</code>	false

6. Solution

- i. false
- ii. true

7. Solution

- i. `age < 12 && age != 8`
- ii. `age >= 6 && age <= 9 || age == 11`
- iii. `age > 7 && age != 10 && age != 12`
- iv. `age == 6 || age == 9 || age == 11`
- v. `age >= 6 && age <= 12 && age != 8`
- vi. `age != 7 && age != 10`

8. Solution

- i. `x != 4 || y == 3`
- ii. `x + 4 > 0`
- iii. `!(x <= 5) && y != 4`
- iv. `x == false`
- v. `!(x < 4 && z <= 4)`
- vi. `x == 2 || x < -5`

9. Solution

- i. `!(x < 4 || y == 10)`
- ii. `!(x - 2 < 9)`
- iii. `!(! (x < 2) && y == 4)`
- iv. `!(x == false && y != 3)`
- v. First approach: `!(! (x < 2 || y < 2))`
Second approach: `x < 2 || y < 2`
- vi. `!(x == -2 || x > 2)`

Chapter 16

16.2 Review Questions: True/False

- | | |
|----------|----------|
| 1. false | 5. false |
| 2. false | 6. false |
| 3. true | 7. true |
| 4. false | 8. false |

16.3 Review Questions: Multiple Choice

- | | |
|------|------|
| 1. b | 4. d |
| 2. c | 5. d |
| 3. d | |

16.4 Review Exercises

1. Solution

The corrections/additions are in red

```
#include <iostream>
using namespace std;
int main() {
    double x, y;
    cin >> x;
    y = -5;
    if (x * y / 2 > 20)
        y *= 2;
    x += 4 * x * x;
}
cout << x << y << endl;
return 0;
}
```

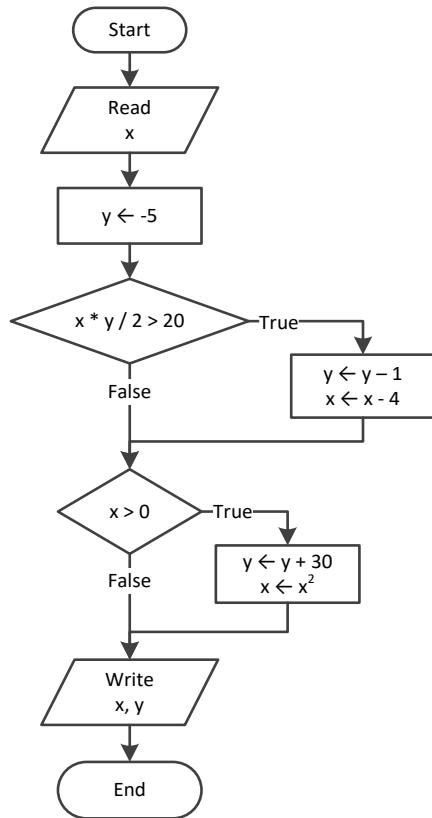
2. Solution

For the input value of 10

Step	Statement	x	y
1	cin >> x	10.0	?
2	y = -5	10.0	-5.0
3	if (x * y / 2 > 20)		false
4	if (x > 0)		true
5	y += 30	10.0	25.0
6	x = pow(x, 2)	100.0	25.0
7	cout << x << ", " << y << endl	It displays: 100, 25	

For the input value of -10

Step	Statement	x	y
1	<code>cin >> x</code>	-10.0	?
2	<code>y = -5</code>	-10.0	-5.0
3	<code>if (x * y / 2 > 20)</code>		true
4	<code>y--</code>	-10.0	-6.0
5	<code>x -= 4</code>	-14.0	-6.0
6	<code>if (x > 0)</code>		false
7	<code>cout << x << ", " << y << endl</code>	It displays: -14, -6	



3. Solution

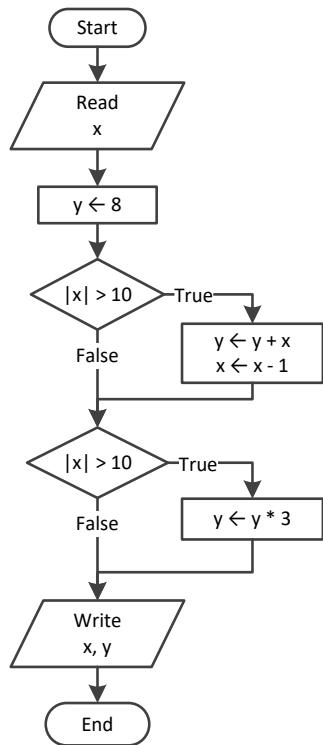
For the input value of -11

Step	Statement	x	y
1	<code>cin >> x</code>	-11	?
2	<code>y = 8</code>	-11	8
3	<code>if (abs(x) > 10)</code>		true
4	<code>y += x</code>	-11	-3
5	<code>x--</code>	-12	-3
6	<code>if (abs(x) > 10)</code>		true
7	<code>y *= 3</code>	-12	-9

8	<code>cout << x << ", " << y << endl</code>	It displays: -12, -9	
----------	---	----------------------	--

For the input value of 11

Step	Statement	x	y
1	<code>cin >> x</code>	11	?
2	<code>y = 8</code>	11	8
3	<code>if (abs(x) > 10)</code>		true
4	<code>y += x</code>	11	19
5	<code>x--</code>	10	19
6	<code>if (abs(x) > 10)</code>		false
7	<code>cout << x << ", " << y << endl</code>	It displays: 10, 19	



4. Solution

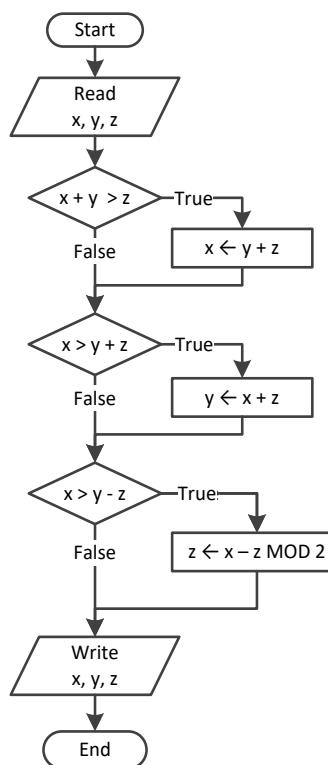
For input values of 1, 2 and 3

Step	Statement	x	y	z
1	<code>cin >> x</code>	1	?	?
2	<code>cin >> y</code>	1	2	?
3	<code>cin >> z</code>	1	2	3
4	<code>if (x + y > z)</code>			false
5	<code>if (x > y + z)</code>			false
6	<code>if (x > y - z)</code>			true
7	<code>z = x - z % 2</code>	1	2	0

8	<code>cout << x << ", " << y << ", " << z << endl</code>	It displays: 1, 2, 0		
----------	--	----------------------	--	--

For input values of 4, 2 and 1

Step	Statement	x	y	z
1	<code>cin >> x</code>	4	?	?
2	<code>cin >> y</code>	4	2	?
3	<code>cin >> z</code>	4	2	1
4	<code>if (x + y > z)</code>			true
5	<code>x = y + z</code>	3	2	1
6	<code>if (x > y + z)</code>			false
7	<code>if (x > y - z)</code>			true
8	<code>z = x - z % 2</code>	3	2	2
9	<code>cout << x << ", " << y << ", " << z << endl</code>			It displays: 3, 2, 2



5. Solution

```

#include <iostream>
using namespace std;
int main() {
    double x;

    cout << "Enter a number: ";
    cin >> x;

    if (x > 0) {
        cout << "Positive" << endl;
    }
}
  
```

```
    }
    return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
int main() {
    double x, y;

    cout << "Enter a number: ";
    cin >> x;
    cout << "Enter a second number";
    cin >> y;

    if (x > 0 && y > 0) {
        cout << "Positive" << endl;
    }
    return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
int main() {
    double x, y;

    cout << "Enter your age: ";
    cin >> x;

    if (x > 14) {
        cout << "You can drive a car in Kansas (USA)" << endl;
    }
    return 0;
}
```

8. Solution

```
#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string str;

    cout << "Enter a string: ";
    cin >> str;

    if (str == to_upper_copy(str)) {
        cout << "Uppercase" << endl;
    }
}
```

```
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    string str;

    cout << "Enter a string: ";
    cin >> str;

    if (str.length() > 20) {
        cout << "Many characters" << endl;
    }
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
int main() {
    double n1, n2, n3, n4;

    cout << "Enter 1st number: ";
    cin >> n1;
    cout << "Enter 2nd number: ";
    cin >> n2;
    cout << "Enter 3rd number: ";
    cin >> n3;
    cout << "Enter 4th number: ";
    cin >> n4;

    if (n1 < 0 || n2 < 0 || n3 < 0 || n4 < 0) {
        cout << "Among the given numbers, there is a negative one!" << endl;
    }
    return 0;
}
```

11. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, b, c;

    cout << "Enter 1st number: ";
    cin >> a;
    cout << "Enter 2nd number: ";
    cin >> b;
```

```
if (a > b) {  
    c = a;  
    a = b;  
    b = c;  
}  
  
cout << a << ", " << b << endl;  
return 0;  
}
```

12. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    double average, t1, t2, t3;  
  
    cout << "Enter 1st temperature: ";  
    cin >> t1;  
    cout << "Enter 2nd temperature: ";  
    cin >> t2;  
    cout << "Enter 3rd temperature: ";  
    cin >> t3;  
  
    average = (t1 + t2 + t3) / 3;  
  
    if (average > 60) {  
        cout << "Heat Wave" << endl;  
    }  
    return 0;  
}
```

Chapter 17

17.2 Review Questions: True/False

- | | |
|----------|----------|
| 1. false | 4. false |
| 2. true | 5. false |
| 3. true | 6. false |

17.3 Review Questions: Multiple Choice

- | |
|------|
| 1. b |
| 2. c |
| 3. c |

17.4 Review Exercises

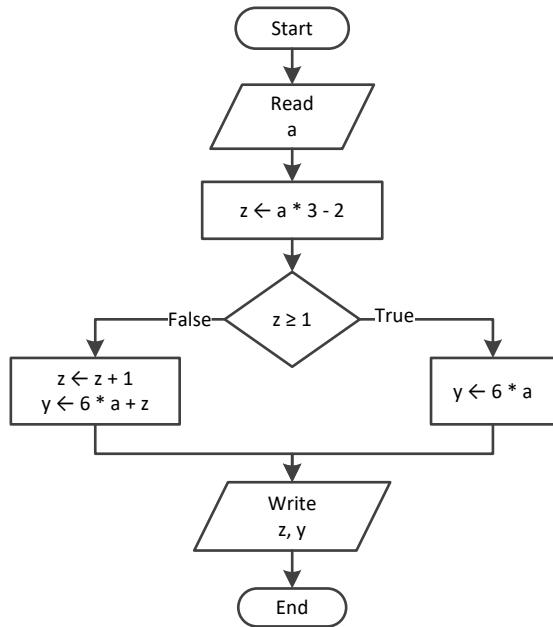
1. Solution

For input value of 3

Step	Statement	a	y	z
1	cin >> a	3.0	?	?
2	z = a * 3 - 2	3.0	?	7.0
3	if (z >= 1)		true	
4	y = 6 * a	3.0	18.0	7.0
5	cout << z << ", " << y << endl	It displays: 7 18		

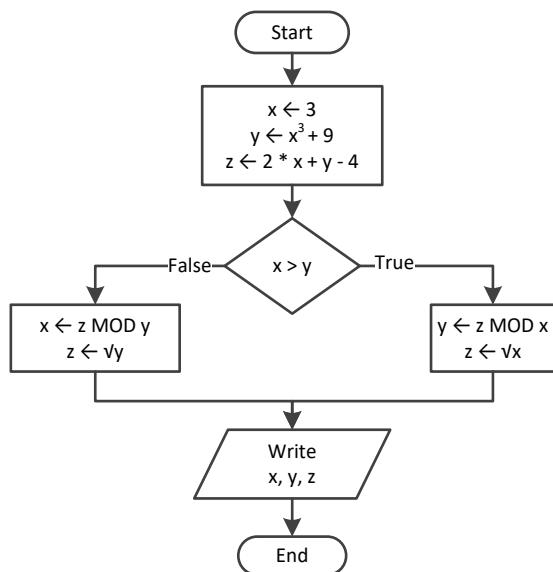
For input value of 0.5

Step	Statement	a	y	z
1	cin >> a	0.5	?	?
2	z = a * 3 - 2	0.5	?	-0.5
3	if (z >= 1)		false	
4	z++	0.5	?	0.5
5	y = 6 * a + z	0.5	3.5	0.5
6	cout << z << ", " << y << endl	It displays: 0.5, 3.5		



2. Solution

Step	Statement	x	y	z
1	$x = 3$	3.0	?	?
2	$y = \text{pow}(x, 3) + 9$	3.0	36.0	?
3	$z = 2 * x + y - 4$	3.0	36.0	38.0
4	$\text{if } (x > y)$			false
5	$x = z \% y$	2.0	36.0	38.0
6	$z = \sqrt{y}$	2.0	36.0	6.0
7	$\text{cout} \ll x \ll ", " \ll y \ll ", " \ll z \ll \text{endl}$	It displays: 2, 36, 6		



3. Solution

```
#include <iostream>
using namespace std;
int main() {
    double w, x, y, z;

    cin >> x;
    w = x * 3 - 15;
    z = (w + 7) * (x + 4) - 10;
    if (w > x && z > x) {
        x++;
        y = x / 2 + 4;
    }
    else {
        y = x / 4 + 2;
    }
    cout << y << endl;
    return 0;
}
```

For input value of 10

Step	Statement	x	y	w	z
1	cin >> x	10.0	?	?	?
2	w = x * 3 - 15	10.0	?	15.0	?
3	z = (w + 7) * (x + 4) - 10	10.0	?	15.0	298.0
4	if (w > x && z > x)		true		
5	x++	11.0	?	15.0	298.0
6	y = x / 2 + 4	11.0	9.5	15.0	298.0
7	cout << y << endl		It displays: 9.5		

For input value of 2

Step	Statement	x	y	w	z
1	cin >> x	2.0	?	?	?
2	w = x * 3 - 15	2.0	?	-9.0	?
3	z = (w + 7) * (x + 4) - 10	2.0	?	-9.0	-22.0
4	if (w > x && z > x)		false		
5	y = x / 4 + 2	2.0	2.5	-9.0	-22.0
6	cout << y << endl		It displays: 2.5		

4. Solution

```
#include <iostream>
using namespace std;
int main() {
    string name1, name2;
    int goals1, goals2;
```

```
cout << "Enter team name 1: ";
cin >> name1;
cout << "Enter team name 2: ";
cin >> name2;

cout << "Enter goals " << name1 << " scored: ";
cin >> goals1;
cout << "Enter goals " << name2 << " scored: ";
cin >> goals2;

if (goals1 > goals2) {
    cout << "Winner: " << name1 << endl;
}
else {
    cout << "Winner: " << name2 << endl;
}
return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
int main() {
    int x;

    cin >> x;
    if (x % 6 == 0) {
        cout << x << " is a multiple of 6" << endl;
    }
    else {
        cout << x << " is not a multiple of 6" << endl;
    }
    return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
int main() {
    int x;

    cin >> x;
    if (x % 6 == 0 || x % 7 == 0) {
        cout << x << " is a multiple of 6 or a multiple of 7" << endl;
    }
    else {
        cout << x << " is neither a multiple of 6 nor a multiple of 7" << endl;
    }
    return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
int main() {
    int x, y;

    cin >> x;

    y = x % 4;
    if (y == 0) {
        cout << x << " is a multiple of 4" << endl;
    }
    else {
        cout << x << " is not a multiple of 4" << endl;
    }

    cout << "The structure is: " << x << " = " << (int)(x / 4) << " x 4 + " << y << endl;
    return 0;
}
```

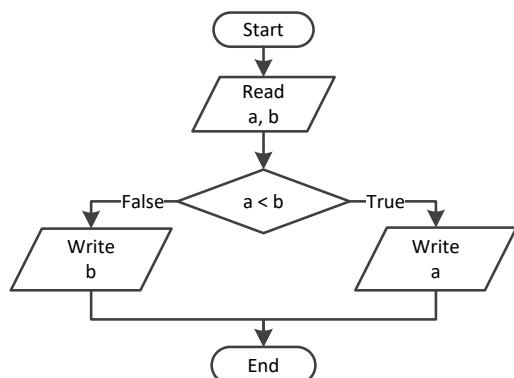
8. Solution

```
#include <iostream>
using namespace std;
int main() {
    int x;

    cin >> x;

    if (x >= 1000 && x <= 9999) {
        cout << x << " is a four-digit integer" << endl;
    }
    else {
        cout << x << " is not a four-digit integer" << endl;
    }
    return 0;
}
```

9. Solution



```
#include <iostream>
using namespace std;
int main() {
    double a, b;

    cin >> a >> b;

    if (a < b) {
        cout << a << endl;
    }
    else {
        cout << b << endl;
    }
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, b, c;

    cin >> a >> b >> c;

    if (a < b + c && b < a + c && c < a + b) {
        cout << "Given numbers can be lengths of the three sides of a triangle" << endl;
    }
    else {
        cout << "Given numbers cannot be lengths of the three sides of a triangle" << endl;
    }
    return 0;
}
```

11. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double a, b, c;

    cin >> a >> b >> c;

    if (pow(a, 2) == pow(b, 2) + pow(c, 2) ||
        pow(b, 2) == pow(a, 2) + pow(c, 2) ||
        pow(c, 2) == pow(a, 2) + pow(b, 2)) {
        cout << "Given numbers can be lengths of the three sides of a right triangle" << endl;
    }
    else {
        cout << "Given numbers cannot be lengths of the three sides of a right triangle" << endl;
    }
    return 0;
}
```

```
}
```

12. Solution

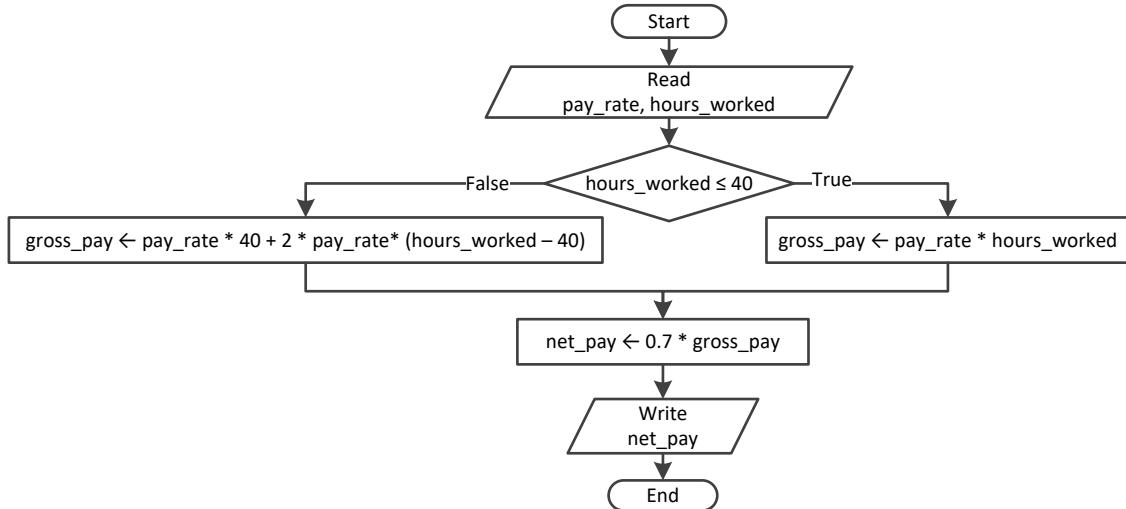
```
#include <iostream>
using namespace std;
int main() {
    double a, average, b, c;

    cout << "Enter 1st jump in meters: ";
    cin >> a;
    cout << "Enter 2nd jump in meters: ";
    cin >> b;
    cout << "Enter 3rd jump in meters: ";
    cin >> c;

    average = (a + b + c) / 3;

    if (average >= 8) {
        cout << "Qualified" << endl;
    }
    else {
        cout << "Disqualified" << endl;
    }
    return 0;
}
```

13. Solution



```
#include <iostream>
using namespace std;
int main() {
    double gross_pay, net_pay, pay_rate;
    int hours_worked;

    cin >> pay_rate >> hours_worked;

    if (hours_worked <= 40) {
```

```
        gross_pay = pay_rate * hours_worked;
    }
    else {
        gross_pay = pay_rate * 40 + 2 * pay_rate * (hours_worked - 40);
    }

    net_pay = 0.7 * gross_pay;
    cout << net_pay << endl;
    return 0;
}
```

14. Solution

```
#include <iostream>
using namespace std;
int main() {
    int miles, miles_left, r;

    cout << "Enter miles traveled: ";
    cin >> miles;

    r = miles % 12000;

    if (r > 6000) {
        miles_left = 12000 - r;
        cout << "Your car needs a major service in " << miles_left << " miles" << endl;
    }
    else {
        miles_left = 6000 - r;
        cout << "Your car needs a minor service in " << miles_left << " miles" << endl;
    }
    return 0;
}
```

15. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double a1, a2, s1, s2, t;

    cout << "Enter the time the two cars traveled: ";
    cin >> t;
    cout << "Enter the acceleration for car A: ";
    cin >> a1;
    cout << "Enter the acceleration for car B: ";
    cin >> a2;

    s1 = 0.5 * a1 * pow(t, 2);
    s2 = 0.5 * a2 * pow(t, 2);

    cout << "Distance between them: " << abs(s1 - s2) << " meters" << endl;
```

```
if (s1 > s2) {  
    cout << "Car A is first" << endl;  
}  
else {  
    cout << "Car B is first" << endl;  
}  
return 0;  
}
```

Chapter 18

18.2 Review Questions: True/False

- | | |
|----------|----------|
| 1. true | 5. false |
| 2. false | 6. true |
| 3. false | 7. false |
| 4. false | 8. true |

18.3 Review Exercises

1. Solution

For input value of 5

Step	Statement	q	b
1	cin >> q	5	?
2	if (q > 0 && q <= 50)		true
3	b = 1	5	1
4	cout << b << endl		It displays: 1

For input value of 150

Step	Statement	q	b
1	cin >> q	150	?
2	if (q > 0 && q <= 50)		false
3	else if (q > 50 && q <= 100)		false
4	else if (q > 100 && q <= 200)		true
5	b = 3	150	3
6	cout << b << endl		It displays: 3

For input value of 250

Step	Statement	q	b
1	cin >> q	250	?
2	if (q > 0 && q <= 50)		false
3	else if (q > 50 && q <= 100)		false
4	else if (q > 100 && q <= 200)		false
5	b = 4	250	4
6	cout << b << endl		It displays: 4

For input value of -1

Step	Statement	q	b
1	cin >> q	-1	?
2	if (q > 0 && q <= 50)		false
3	else if (q > 50 && q <= 100)		false
4	else if (q > 100 && q <= 200)		false

5	b = 4	-1	4
6	cout << b << endl		It displays: 4

2. Solution

For input value of 5

Step	Statement	amount	discount	payment
1	cin >> amount	5.0	?	?
2	discount = 0	5.0	0.0	?
3	if (amount < 20)		true	
4	discount = 0	5.0	0.0	?
5	payment = amount - amount * discount / 100	5.0	0.0	5.0
6	cout << discount << ", " << payment << endl		It displays: 0, 5	

For input value of 150

Step	Statement	amount	discount	payment
1	cin >> amount	150.0	?	?
2	discount = 0	150.0	0.0	?
3	if (amount < 20)		false	
4	else if (amount >= 20 && amount < 60)		false	
5	else if (amount >= 60 && amount < 100)		false	
6	else if (amount >= 100)		true	
7	discount = 15	150.0	15.0	?
8	payment = amount - amount * discount / 100	150.0	15.0	127.5
9	cout << discount << ", " << payment << endl		It displays: 15, 127.5	

For input value of -1

Step	Statement	amount	discount	payment
1	cin >> amount	-1.0	?	?
2	discount = 0	-1.0	0.0	?
3	if (amount < 20)		true	
4	discount = 0	-1.0	0.0	?
5	payment = amount - amount * discount / 100	-1.0	0.0	-1.0
6	cout << discount << ", " << payment << endl		It displays: 0, -1	

3. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, y;
    cin >> a;
```

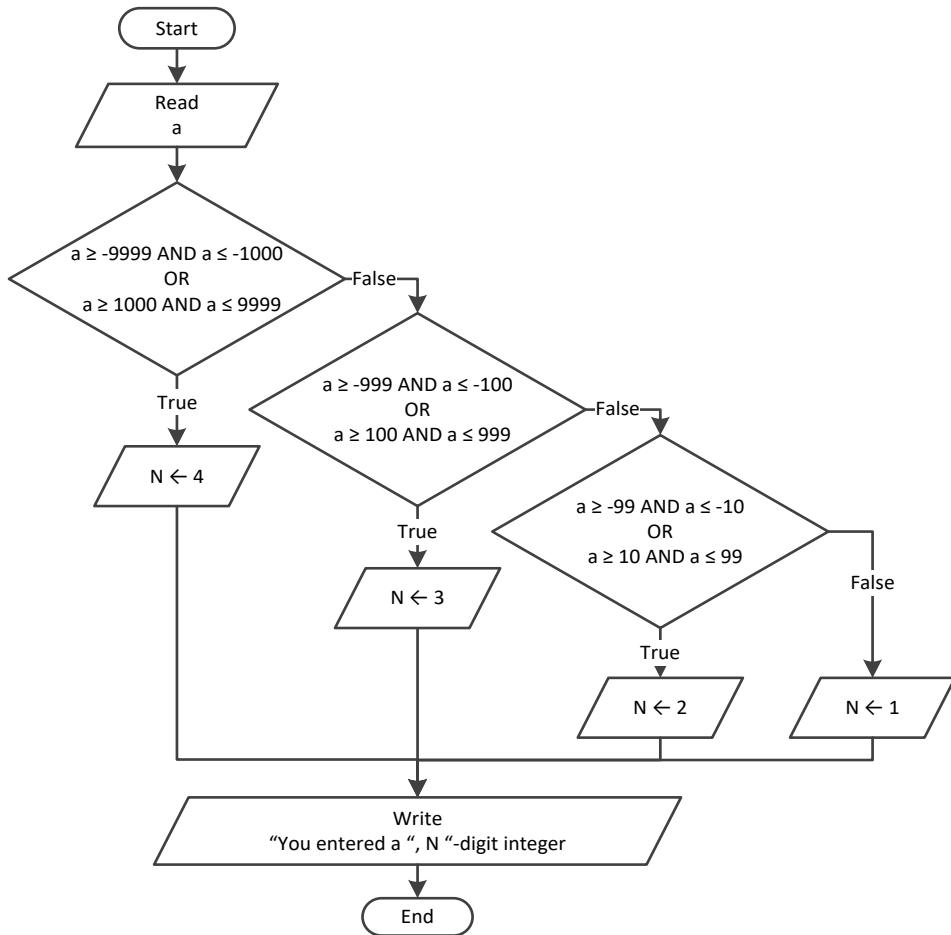
```
if (a < 1) {  
    y = 5 + a;  
    cout << y << endl;  
}  
else if (a < 5) {  
    y = 23 / a;  
    cout << y << endl;  
}  
else if (a < 10) {  
    y = 5 * a;  
    cout << y << endl;  
}  
else {  
    cout << "Error!" << endl;  
}  
return 0;  
}
```

4. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    string name1, name2;  
    int goals1, goals2;  
  
    cout << "Enter team name 1: ";  
    cin >> name1;  
    cout << "Enter team name 2: ";  
    cin >> name2;  
  
    cout << "Enter goals " << name1 << " scored: ";  
    cin >> goals1;  
    cout << "Enter goals " << name2 << " scored: ";  
    cin >> goals2;  
  
    if (goals1 > goals2) {  
        cout << "Winner: " << name1 << endl;  
    }  
    else if (goals2 > goals1) {  
        cout << "Winner: " << name2 << endl;  
    }  
    else {  
        cout << "It's a tie!" << endl;  
    }  
    return 0;  
}
```

5. Solution

First Approach



```

#include <iostream>
using namespace std;
int main() {
    int a, n;

    cin >> a;

    if (a >= -9999 && a <= -1000 || a >= 1000 && a <= 9999) {
        n = 4;
    }
    else if (a >= -999 && a <= -100 || a >= 100 && a <= 999) {
        n = 3;
    }
    else if (a >= -99 && a <= -10 || a >= 10 && a <= 99) {
        n = 2;
    }
    else {
        n = 1;
    }

    cout << "You entered a " << n << "-digit integer" << endl;
  
```

```

    return 0;
}

```

Second Approach

```

#include <iostream>
#include <string>
using namespace std;
int main() {
    int a;
    string a_string;

    cin >> a;

    a_string = to_string(abs(a));
    cout << "You entered a " << a_string.length() << "-digit integer" << endl;
    return 0;
}

```

6. Solution

First Approach

```

#include <iostream>
using namespace std;
int main() {
    int a, n;

    cin >> a;

    if (a >= -9999 && a <= -1000 || a >= 1000 && a <= 9999) {
        cout << "You entered a 4-digit integer" << endl;
    }
    else if (a >= -999 && a <= -100 || a >= 100 && a <= 999) {
        cout << "You entered a 3-digit integer" << endl;
    }
    else if (a >= -99 && a <= -10 || a >= 10 && a <= 99) {
        cout << "You entered a 2-digit integer" << endl;
    }
    else if (a >= -9 && a <= 9) { //Include the value of zero
        cout << "You entered a 1-digit integer" << endl;
    }
    else {
        cout << "Error: Invalid value!" << endl;
    }
    return 0;
}

```

Second Approach

```

#include <iostream>
#include <string>
using namespace std;
int main() {
    int a;
    string a_string;

```

```
    cin >> a;

    if (a >= -9999 && a <= 9999) {
        a_string = to_string(abs(a));
        cout << "You entered a " << a_string.length() << "-digit integer" << endl;
    }
    else {
        cout << "Error: Invalid value!" << endl;
    }
    return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
int main() {
    double cad, eur, gbp, jpy, usd;
    int ch;

    cout << "1. Convert USD to Euro (EUR)" << endl;
    cout << "2. Convert USD to British Pound Sterling (GBP)" << endl;
    cout << "3. Convert USD to Japanese Yen (JPY)" << endl;
    cout << "4. Convert USD to Canadian Dollar (CAD)" << endl;

    cout << "Enter a choice: ";
    cin >> ch;

    cout << "Enter an amount in US dollars: ";
    cin >> usd;

    if (ch == 1) {
        eur = usd * 0.87;
        cout << "$" << usd << " = " << eur << " EUR" << endl;
    }
    else if (ch == 2) {
        gbp = usd * 0.78;
        cout << "$" << usd << " = " << gbp << " GBP" << endl;
    }
    else if (ch == 3) {
        jpy = usd * 108.55;
        cout << "$" << usd << " = " << jpy << " JPY" << endl;
    }
    else {
        cad = usd * 1.33;
        cout << "$" << usd << " = " << cad << " CAD" << endl;
    }
    return 0;
}
```

8. Solution

```
#include <iostream>
```

```
using namespace std;
int main() {
    int m;

    cout << "Enter the number of a month between 1 and 12: ";
    cin >> m;

    if (m <= 2 || m == 12) {
        cout << "Winter" << endl;
    }
    else if (m <= 5) {
        cout << "Spring" << endl;
    }
    else if (m <= 8) {
        cout << "Summer" << endl;
    }
    else {
        cout << "Fall (Autumn)" << endl;
    }
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    int m;

    cout << "Enter the number of a month between 1 and 12: ";
    cin >> m;

    if (m < 1 || m > 12) {
        cout << "Error: Invalid value!" << endl;
    }
    else if (m <= 2 || m == 12) {
        cout << "Winter" << endl;
    }
    else if (m <= 5) {
        cout << "Spring" << endl;
    }
    else if (m <= 8) {
        cout << "Summer" << endl;
    }
    else {
        cout << "Fall (Autumn)" << endl;
    }
    return 0;
}
```

10. Solution

```
#include <iostream>
```

```
using namespace std;
int main() {
    double n;
    int x, y;

    cout << "Enter a number between 1.0 and 4.9: ";
    cin >> n;

    x = (int)(n);
    y = (int)(n * 10) % 10;

    if (x == 1) {
        cout << "One";
    }
    else if (x == 2) {
        cout << "Two";
    }
    else if (x == 3) {
        cout << "Three";
    }
    else if (x == 4) {
        cout << "Four";
    }

    cout << " point ";

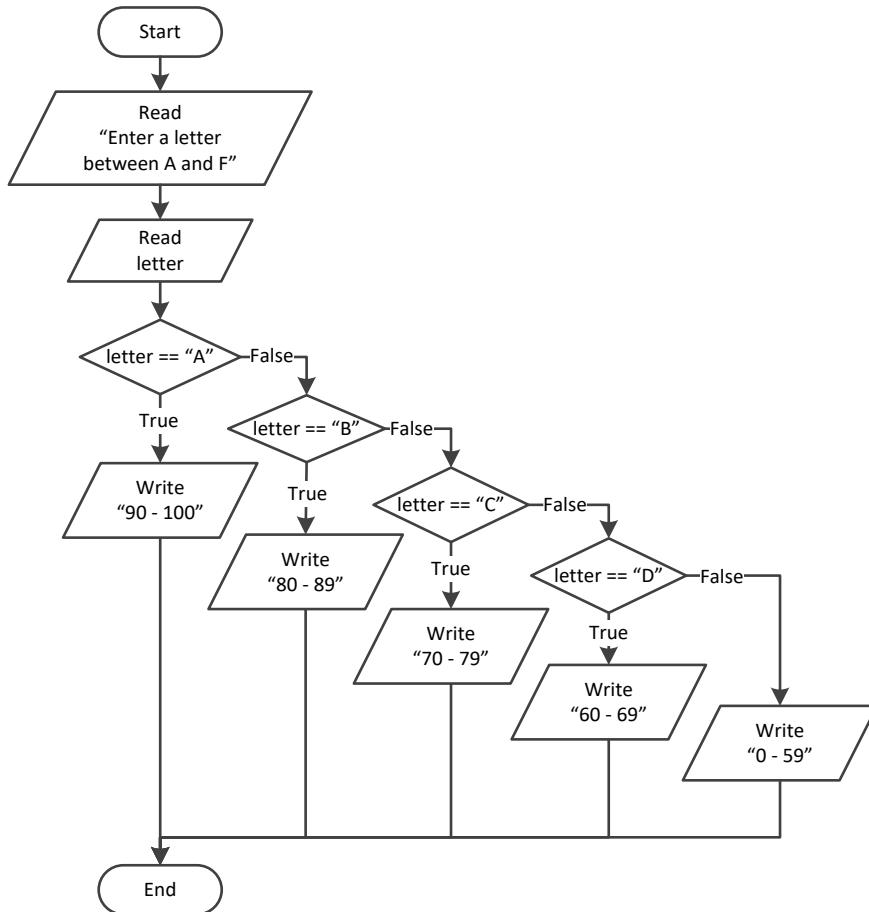
    if (y == 1) {
        cout << "one" << endl;
    }
    else if (y == 2) {
        cout << "two" << endl;
    }
    else if (y == 3) {
        cout << "three" << endl;
    }
    else if (y == 4) {
        cout << "four" << endl;
    }
    else if (y == 5) {
        cout << "five" << endl;
    }
    else if (y == 6) {
        cout << "six" << endl;
    }
    else if (y == 7) {
        cout << "seven" << endl;
    }
    else if (y == 8) {
        cout << "eight" << endl;
    }
    else if (y == 9) {
        cout << "nine" << endl;
```

```

    }
else if (y == 0) {
    cout << "zero" << endl;
}
return 0;
}
}

```

11. Solution



```

#include <iostream>
using namespace std;
int main() {
    string letter;

    cout << "Enter a letter between A and F: ";
    cin >> letter;

    if (letter == "A") {
        cout << "90 - 100" << endl;
    }
    else if (letter == "B") {
        cout << "80 - 89" << endl;
    }
    else if (letter == "C") {
        cout << "70 - 79" << endl;
    }
    else if (letter == "D") {
        cout << "60 - 69" << endl;
    }
    else if (letter == "F") {
        cout << "0 - 59" << endl;
    }
}

```

```
    }
else if (letter == "D") {
    cout << "60 - 69" << endl;
}
else {
    cout << "0 - 59" << endl;
}
}
```

Chapter 19

19.2 Review Questions: True/False

- | | |
|----------|----------|
| 1. true | 5. true |
| 2. false | 6. false |
| 3. true | 7. true |
| 4. false | |

19.3 Review Exercises

1. Solution

For input value of 1

Step	Statement	a	x	y
1	cin >> a	1	?	?
2	x = 0	1	0	?
3	y = 0	1	0	0
4	case a == 1		true	
5	x = x + 5	1	5	0
6	y = y + 5	1	5	5
7	cout << x << ", " << y << endl	It displays: 5, 5		

For input value of 3

Step	Statement	a	x	y
1	cin >> a	3	?	?
2	x = 0	3	0	?
3	y = 0	3	0	0
4	case a == 1		false	
5	case a == 2		false	
6	case a == 3		true	
7	x = x - 9	3	-9	0
8	y = y + 3	3	-9	3
9	cout << x << ", " << y << endl	It displays: -9, 3		

For input value of 250

Step	Statement	a	x	y
1	cin >> a	250	?	?
2	x = 0	250	0	?
3	y = 0	250	0	0
4	case a == 1		false	
5	case a == 2		false	
6	case a == 3		false	

7	x = x + 3	250	3	0
8	y++	250	3	1
9	cout << x << ", " << y << endl	It displays: 3, 1		

2. Solution

For input values of 10, 2, 5

Step	Statement	a	x	y
1	cin >> a	10	?	?
2	cin >> x	10	2	?
3	cin >> y	10	2	5.0
4	case a == 10			true
5	x = x % 2	10	0	5.0
6	y = pow(y, 2)	10	0	25.0
7	cout << x << ", " << y << endl	It displays: 0, 25		

For input values of 5, 2, 3

Step	Statement	a	x	y
1	cin >> a	5	?	?
2	cin >> x	5	2	?
3	cin >> y	5	2	3.0
4	case a == 10			false
5	case a == 3			false
6	case a == 5			true
7	x = x + 4	5	6	3.0
8	y += 7	5	6	10.0
9	cout << x << ", " << y << endl	It displays: 6, 10		

For input values of 4, 6, 2

Step	Statement	a	x	y
1	cin >> a	4	?	?
2	cin >> x	4	6	?
3	cin >> y	4	6	2.0
4	case a == 10			false
5	case a == 3			false
6	case a == 5			false
7	x -= 3	4	3	2.0
8	y++	4	3	3.0
9	cout << x << ", " << y << endl	It displays: 3, 3		

3. Solution

```
#include <iostream>
using namespace std;
int main() {
    int number;

    cout << "Enter the number of a month: ";
    cin >> number;

    switch (number) {
        case 1:
            cout << "JANUARY" << endl;
            break;
        case 2:
            cout << "FEBRUARY" << endl;
            break;
        case 3:
            cout << "MARCH" << endl;
            break;
        case 4:
            cout << "APRIL" << endl;
            break;
        case 5:
            cout << "MAY" << endl;
            break;
        case 6:
            cout << "JUNE" << endl;
            break;
        case 7:
            cout << "JULY" << endl;
            break;
        case 8:
            cout << "AUGUST" << endl;
            break;
        case 9:
            cout << "SEPTEMBER" << endl;
            break;
        case 10:
            cout << "OCTOBER" << endl;
            break;
        case 11:
            cout << "NOVEMBER" << endl;
            break;
        case 12:
            cout << "DECEMBER" << endl;
            break;
        default:
            cout << "Error" << endl;
    }
    return 0;
}
```

```
}
```

4. Solution

```
#include <iostream>
using namespace std;
int main() {
    int choice;
    double feet, inches, miles, yards;

    cout << "1. Convert Miles to Yards" << endl;
    cout << "2. Convert Miles to Feet" << endl;
    cout << "3. Convert Miles to Inches" << endl;

    cout << "Enter a choice: ";
    cin >> choice;

    switch (choice) {
        case 1:
            cout << "Enter miles: ";
            cin >> miles;
            yards = miles * 1760;
            cout << miles << " miles = " << yards << " yards" << endl;
            break;
        case 2:
            cout << "Enter miles: ";
            cin >> miles;
            feet = miles * 5280;
            cout << miles << " miles = " << feet << " feet" << endl;
            break;
        case 3:
            cout << "Enter miles: ";
            cin >> miles;
            inches = miles * 63360;
            cout << miles << " miles = " << inches << " inches" << endl;
            break;
        default:
            cout << "Invalid choice!" << endl;
    }
    return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
int main() {
    int number;

    cout << "Enter a number between 1 and 10: ";
    cin >> number;

    switch (number) {
```

```
case 1:  
    cout << "I" << endl;  
    break;  
case 2:  
    cout << "II" << endl;  
    break;  
case 3:  
    cout << "III" << endl;  
    break;  
case 4:  
    cout << "IV" << endl;  
    break;  
case 5:  
    cout << "V" << endl;  
    break;  
case 6:  
    cout << "VI" << endl;  
    break;  
case 7:  
    cout << "VII" << endl;  
    break;  
case 8:  
    cout << "VIII" << endl;  
    break;  
case 9:  
    cout << "IX" << endl;  
    break;  
case 10:  
    cout << "X" << endl;  
    break;  
default:  
    cout << "Error" << endl;  
}  
return 0;  
}
```

6. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    int total;  
  
    cout << "Enter the total number of CDs purchased in a month: ";  
    cin >> total;  
  
    switch (total) {  
        case 1:  
            cout << "You are awarded 3 points" << endl;  
            break;  
        case 2:  
            cout << "You are awarded 10 points" << endl;
```

```
        break;
    case 3:
        cout << "You are awarded 20 points" << endl;
        break;
    default:
        cout << "You are awarded 45 points" << endl;
    }
    return 0;
}
```

7. Solution

```
#include <iostream>
#include <ctime>
#include <cstdlib>
using namespace std;
int main() {
    int i;
    string name;

    srand(time(NULL));

    cout << "Enter your name: ";
    cin >> name;

    i = rand() % 3;

    switch (i) {
        case 0:
            cout << "Good morning " << name << endl;
            break;
        case 1:
            cout << "Good evening " << name << endl;
            break;
        case 2:
            cout << "Good night " << name << endl;
            break;
    }
    return 0;
}
```

8. Solution

```
#include <iostream>
using namespace std;
int main() {
    int num;

    cin >> num;

    switch (num) {
        case 0:
            cout << "zero" << endl;

```

```
        break;
    case 1:
        cout << "one" << endl;
        break;
    case 2:
        cout << "two" << endl;
        break;
    case 3:
        cout << "three" << endl;
        break;
    case 4:
        cout << "four" << endl;
        break;
    case 5:
        cout << "five" << endl;
        break;
    case 6:
        cout << "six" << endl;
        break;
    case 7:
        cout << "seven" << endl;
        break;
    case 8:
        cout << "eight" << endl;
        break;
    case 9:
        cout << "nine" << endl;
        break;
    default:
        cout << "I don't know this number!" << endl;
    }
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    int b;

    cout << "Enter Beaufort number: ";
    cin >> b;

    switch (b) {
        case 0:
            cout << "Calm" << endl;
            break;
        case 1:
            cout << "Light Air" << endl;
            break;
        case 2:
```

```
    cout << "Light breeze" << endl;
    break;
case 3:
    cout << "Gentle breeze" << endl;
    break;
case 4:
    cout << "Moderate breeze" << endl;
    break;
case 5:
    cout << "Fresh breeze" << endl;
    break;
case 6:
    cout << "Strong breeze" << endl;
    break;
case 7:
    cout << "Moderate gale" << endl;
    break;
case 8:
    cout << "Gale" << endl;
    break;
case 9:
    cout << "Strong gale" << endl;
    break;
case 10:
    cout << "Storm" << endl;
    break;
case 11:
    cout << "Violent storm" << endl;
    break;
case 12:
    cout << "Hurricane force" << endl;
    break;
default:
    cout << "Invalid Beaufort number!" << endl;
}
return 0;
}
```

Chapter 20

20.3 Review Questions: True/False

- | | |
|----------|----------|
| 1. true | 4. false |
| 2. true | 5. true |
| 3. false | |

20.4 Review Exercises

1. Solution

For input values of 20, 1

Step	Statement	x	y
1	cin >> x	20	?
2	cin >> y	20	1
3	if (x < 30)		true
4	case y == 1		true
5	x = x % 3	2	1
6	y = 5	2	5
7	cout << x << ", " << y << endl	It displays: 2, 5	

For input values of 20, 3

Step	Statement	x	y
1	cin >> x	20	?
2	cin >> y	20	3
3	if (x < 30)		true
4	case y == 1		false
5	case y == 2		false
6	case y == 3		true
7	x = x + 5	25	3
8	y += 3	25	6
9	cout << x << ", " << y << endl	It displays: 25, 6	

For input values of 12, 8

Step	Statement	x	y
1	cin >> x	12	?
2	cin >> y	12	8
3	if (x < 30)		true
4	case y == 1		false
5	case y == 2		false
6	case y == 3		false
7	x -= 2	10	8

8	y++	10	9
9	cout << x << ", " << y << endl	It displays: 10, 9	

For input values of 50, 0

Step	Statement	x	y
1	cin >> x	50	?
2	cin >> y	50	0
3	if (x < 30)	false	
4	y++	50	1
5	cout << x << ", " << y << endl	It displays: 50, 1	

2. Solution

For input values of 60, 25

Step	Statement	x	y
1	cin >> x	60	?
2	cin >> y	60	25
3	if ((x + y) / 2 <= 20)	false	
4	if (y < 15)	false	
5	else if (y < 23)	false	
6	x = 2 * x + 5	125	25
7	y += 1	125	26
8	cout << x << ", " << y << endl	It displays: 125, 26	

For input values of 50, 8

Step	Statement	x	y
1	cin >> x	50	?
2	cin >> y	50	8
3	if ((x + y) / 2 <= 20)	false	
4	if (y < 15)	true	
5	x = x % 4	2	8
6	y = 2	2	2
7	cout << x << ", " << y << endl	It displays: 2, 2	

For input values of 20, 15

Step	Statement	x	y
1	cin >> x	20	?
2	cin >> y	20	15
3	if ((x + y) / 2 <= 20)	true	
4	if (y < 10)	false	
5	else if (y < 20)	true	

6	x = x * 5	100	15
7	y += 2	100	17
8	cout << x << ", " << y << endl	It displays: 100, 17	

For input values of 10, 30

Step	Statement	x	y
1	cin >> x	10	?
2	cin >> y	10	30
3	if ((x + y) / 2 <= 20)		true
4	if (y < 10)		false
5	else if (y < 20)		false
6	x = x - 2	8	30
7	y += 3	8	33
8	cout << x << ", " << y << endl	It displays: 8, 33	

3. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a;

    cin >> a;

    if (a > 1000)
        cout << "Big Positive" << endl;
    else {
        if (a > 0)
            cout << "Positive" << endl;
        else {
            if (a < -1000)
                cout << "Big Negative" << endl;
            else {
                if (a < 0)
                    cout << "Negative" << endl;
                else
                    cout << "Zero" << endl;
            }
        }
    }
    return 0;
}
```

4. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
```

```

int main() {
    double a, b, c;

    cout << "Enter the three sides of a triangle: ";
    cin >> a >> b >> c;

    if (a >= b + c || b >= a + c || c >= a + b) {
        cout << "Given numbers cannot be lengths of the three sides of a triangle" << endl;
    }
    else {
        if (a == b && b == c) {
            cout << "Equilateral" << endl;
        }
        else if (pow(a, 2) == pow(b, 2) + pow(c, 2) ||
                  pow(b, 2) == pow(a, 2) + pow(c, 2) ||
                  pow(c, 2) == pow(a, 2) + pow(b, 2)) {

            cout << "Right (or right-angled)" << endl;
        }
        else {
            cout << "Not special" << endl;
        }
    }
    return 0;
}

```

5. Solution

```

#include <iostream>
using namespace std;
int main() {
    int amount, pin, r, usd1, usd10, usd5;

    cout << "Enter your four-digit PIN : ";
    cin >> pin;
    if (pin != 1234) {
        cout << "Wrong PIN. Enter your four-digit PIN : ";
        cin >> pin;
        if (pin != 1234) {
            cout << "Wrong PIN. Enter your four-digit PIN : ";
            cin >> pin;
        }
    }

    if (pin != 1234) {
        cout << "PIN locked!" << endl;
    }
    else {
        cout << "Enter the amount of money (an integer value) that you want to withdraw: ";
        cin >> amount;
        usd10 = (int) (amount / 10);
        r = amount % 10;
    }
}

```

```
    usd5 = (int)(r / 5);
    usd1 = r % 5;
    cout << usd10 << " note(s) of $10 " << usd5 << " note(s) of $5 ";
    cout << "and " << usd1 << " note(s) of $1" << endl;
}
return 0;
}
```

6. Solution

First Approach

```
#include <iostream>
using namespace std;
int main() {
    double t, w;

    cout << "Enter temperature (in Fahrenheit): ";
    cin >> t;
    cout << "Enter wind speed (in miles/hour): ";
    cin >> w;

    if (t > 75) {
        if (w > 12) {
            cout << "The day is hot and windy" << endl;
        }
        else {
            cout << "The day is hot and not windy" << endl;
        }
    }
    else {
        if (w > 12) {
            cout << "The day is cold and windy" << endl;
        }
        else {
            cout << "The day is cold and not windy" << endl;
        }
    }
    return 0;
}
```

Second Approach

```
#include <iostream>
using namespace std;
int main() {
    double t, w;
    string message1, message2;

    cout << "Enter temperature (in Fahrenheit): ";
    cin >> t;
    cout << "Enter wind speed (in miles/hour): ";
    cin >> w;

    if (t > 75) {
```

```
    message1 = "hot";
}
else {
    message1 = "cold";
}

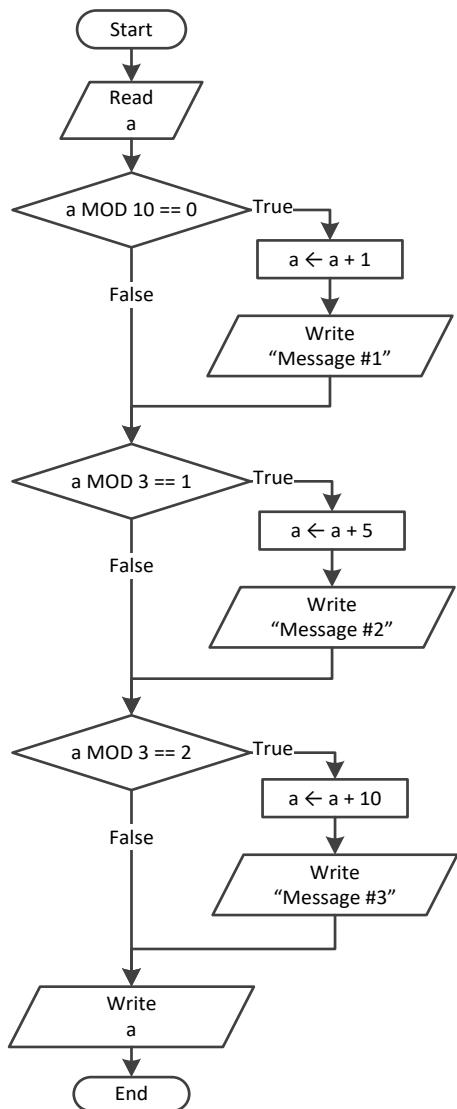
if (w > 12) {
    message2 = "windy";
}
else {
    message2 = "not windy";
}

cout << "The day is " << message1 << " and " << message2 << endl;
return 0;
}
```

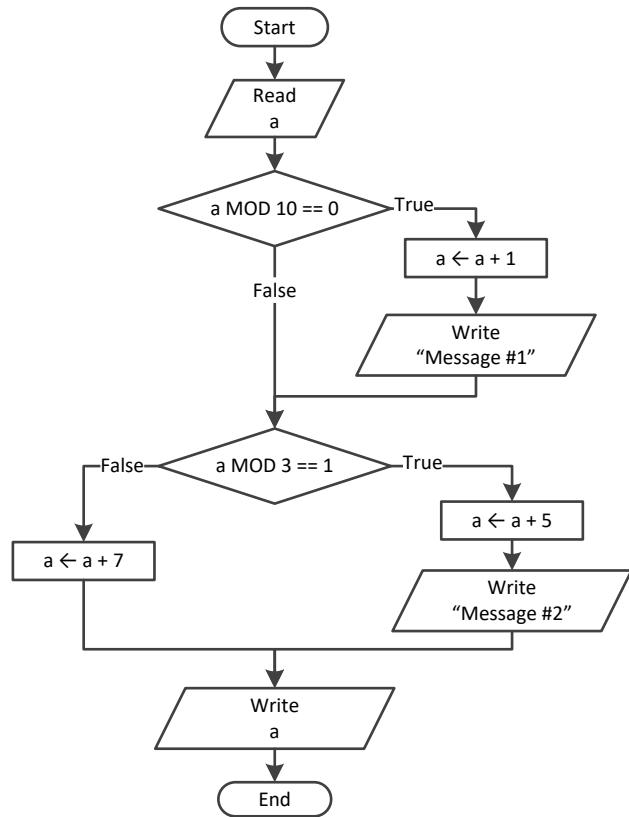
Chapter 21

21.4 Review Exercises

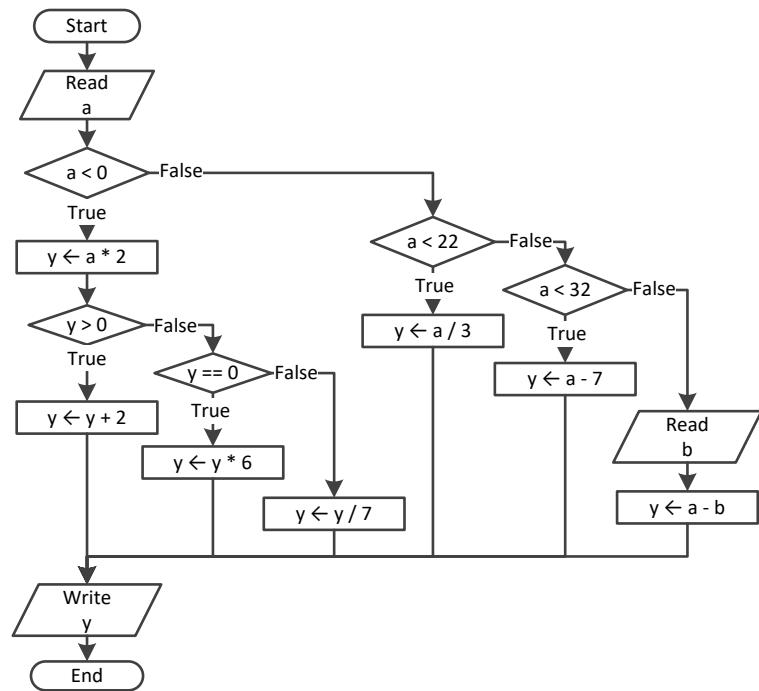
1. Solution



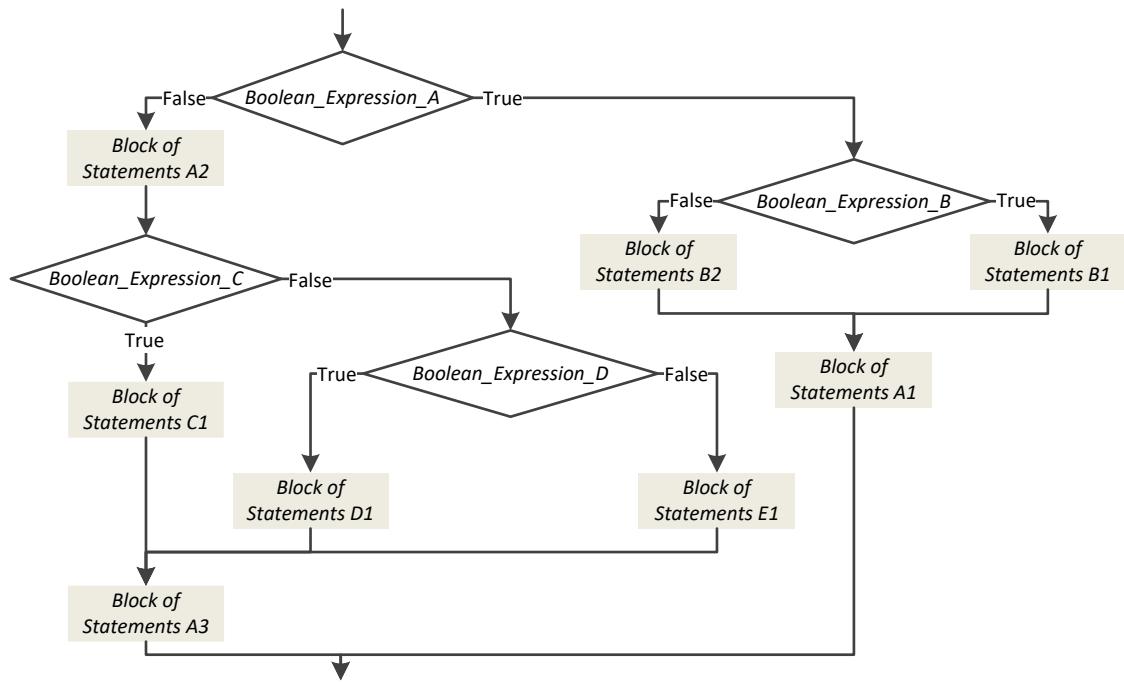
2. Solution



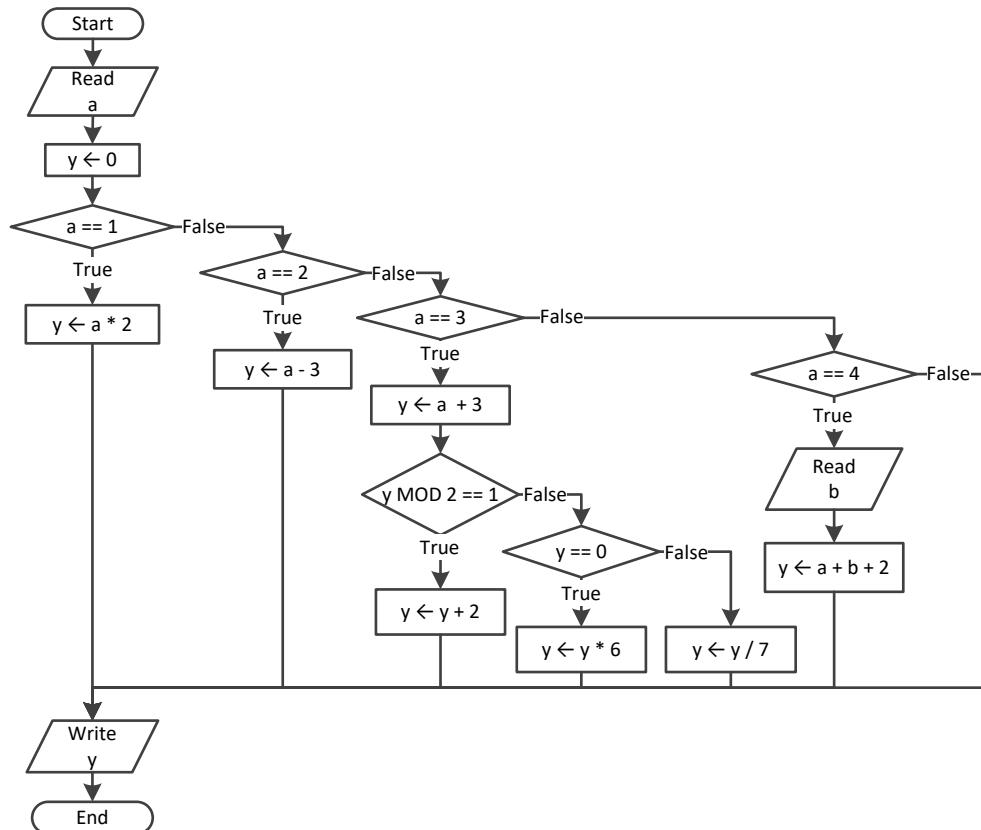
3. Solution



4. Solution



5. Solution



6. Solution

```
#include <iostream>
using namespace std;
int main() {
    double x, y, z;

    cin >> x >> y;

    if (x != 100 || y <= 10) {
        cin >> z;
        if (z <= x + y) {
            x -= 3;
            y = x + 4;
        }
    }
    cout << x << " " << y << endl;
    return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
int main() {
    int x;

    cin >> x;

    if (x == 1) {
        cout << "Good Morning" << endl;
        cout << "How do you do?" << endl;
        cout << "Is everything okay?" << endl;
    }
    else if (x == 2) {
        cout << "Good Evening" << endl;
        cout << "How do you do?" << endl;
        cout << "Is everything okay?" << endl;
    }
    else if (x == 3) {
        cout << "Good Afternoon" << endl;
        cout << "Is everything okay?" << endl;
    }
    else {
        cout << "Good Night" << endl;
    }
    return 0;
}
```

8. Solution

```
#include <iostream>
```

```
using namespace std;
int main() {
    int a, b, c, d, y;

    cin >> a >> b;

    c = a % 2;
    d = (int)(b / 5);

    if (a >= b)
        y = 1;
    else if (d > c && a > 2)
        y = 2;
    else if (d * c > a / b) {
        if (d * c > 10)
            y = 4;
        else
            y = 3;
    }
    else
        y = 5;

    cout << y << endl;
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    int x;

    cin >> x;

    if (x > 0) {
        if (x % 10 == 0) {
            cout << "Last digit equal to 0" << endl;
        }
        else if (x % 10 == 1) {
            cout << "Last digit equal to 1" << endl;
        }
        else {
            cout << "None" << endl;
        }
    }
    else {
        if (x == -1) {
            cout << "Bye" << endl;
        }
        else {
            cout << "Invalid Number" << endl;
        }
    }
}
```

```
    }
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, b, y;

    cin >> a >> b;

    y = a * b;

    if (y > 0) {
        y--;
        y /= 2;
    }
    else {
        y +=10;
        if (y > 0) {
            y /= 2;
        }
        else {
            y *= 2;
        }
    }
    return 0;
}
```

11. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, b, c;

    cin >> a >> b >> c;

    c = a * b + c;
    if (c > 0) {
        c /= 2;
        if (a > b) {
            a *= 2;
            b *= 2;
        }
    }
    else {
        c /= 20;
        if (c <= 10) {
            b *= 2;
        }
    }
}
```

```
    }
else {
    c /= 3;
    c /= 20;
    if (c <= 10) {
        b *= 2;
    }
}
cout << a << " " << b << " " << c << endl;
return 0;
}
```

Chapter 22

22.9 Review Questions: True/False

- | | |
|----------|----------|
| 1. false | 5. true |
| 2. true | 6. false |
| 3. false | 7. false |
| 4. true | |

22.10 Review Questions: Multiple Choice

- | | |
|------|--|
| 1. a | |
| 2. b | |
| 3. a | |
| 4. c | |

22.11 Review Exercises

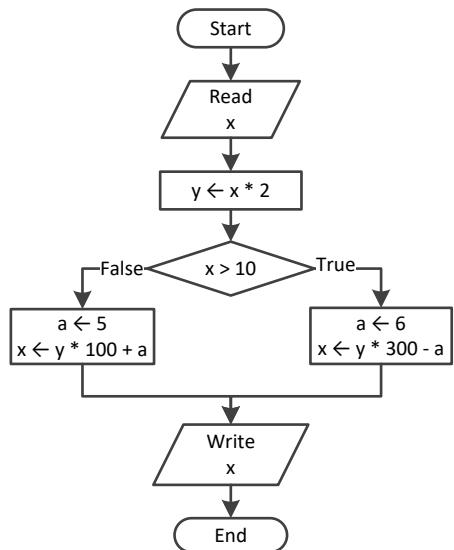
1. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, x, y;

    cin >> y;
    cin >> x;

    if (y > 0) {
        a = x * 4 * y + 1;
    }
    else {
        a = x * 2 * y + 6;
    }
    cout << y << endl;
    cout << a << endl;
    return 0;
}
```

2. Solution



3. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, y;

    cin >> a;

    if (a >= 10) {
```

```

        cout << "Error!" << endl;
    }
else {
    if (a < 1) {
        y = 5 + a;
    }
    else if (a < 5) {
        y = 23 / a;
    }
    else {
        y = 5 * a;
    }
    cout << y << endl;
}
return 0;
}

```

4. Solution

```

#include <iostream>
using namespace std;
int main() {
    int day, month;
    string name;

    cin >> day >> month >> name;

    if (day == 16 && month == 2 && name == "Loukia") {
        cout << "Happy Birthday!!!" << endl;
    }
    else {
        cout << "No match!" << endl;
    }
    return 0;
}

```

5. Solution

It does not operate the same way when variable `a` is less than or equal to 10. The correct program is

```

#include <iostream>
using namespace std;
int main() {
    double a, b, c, d;

    cin >> a >> b >> c;

    if (a > 10) {
        if (c < 2000) {
            d = (a + b + c) / 12;
            cout << "The result is: " << d << endl;
        }
    }
    else {

```

```
        cout << "Error!" << endl;
    }
}
else {
    cout << "Error!" << endl;
}
return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, b, c, d;

    cin >> a >> b >> c;

    if (a > 10 && b < 2000 && c != 10) {
        d = (a + b + c) / 12;
        cout << "The result is: " << d << endl;
    }

    if (a <= 10) {
        cout << "Error!" << endl;
    }
    return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, b, y;

    cin >> a;
    cin >> b;

    y = 3;
    if (a > 0) {
        y = y * a;
        cout << "Hello Zeus" << endl;
    }

    cout << y << " " << b << endl;
    return 0;
}
```

8. Solution

```
#include <iostream>
using namespace std;
```

```
int main() {
    double a, b, y;

    cin >> a;
    cin >> b;

    y = 0;
    if (a > 0) {
        y = y + 7;
    }
    else {
        cout << "Hello Zeus" << endl;
        cout << abs(a) << endl;
    }
    cout << y << endl;
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    string os;

    cout << "What is your tablet's OS? ";
    cin >> os;

    if (os == "iOS") {
        cout << "Apple" << endl;
    }
    else if (os == "Android") {
        cout << "Google" << endl;
    }
    else if (os == "Windows") {
        cout << "Microsoft" << endl;
    }
    return 0;
}
```

Chapter 23

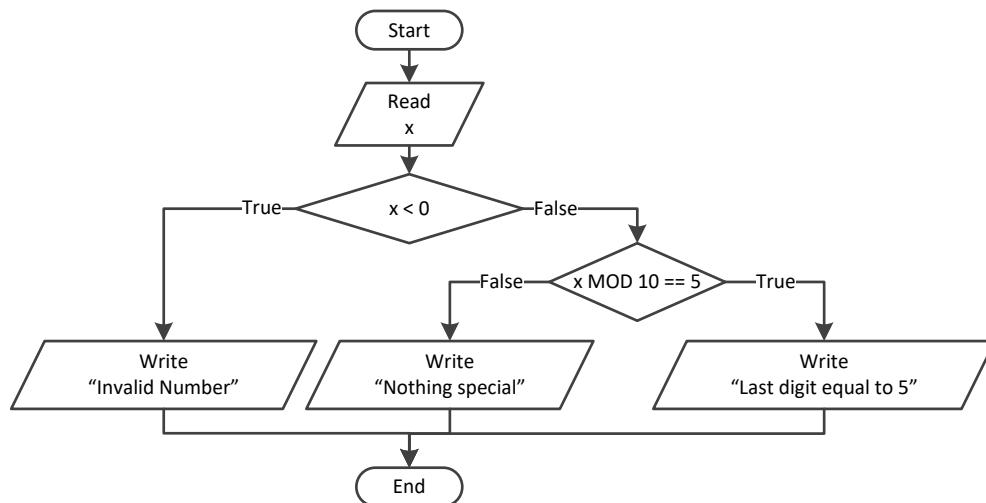
23.6 Review Exercises

1. Solution

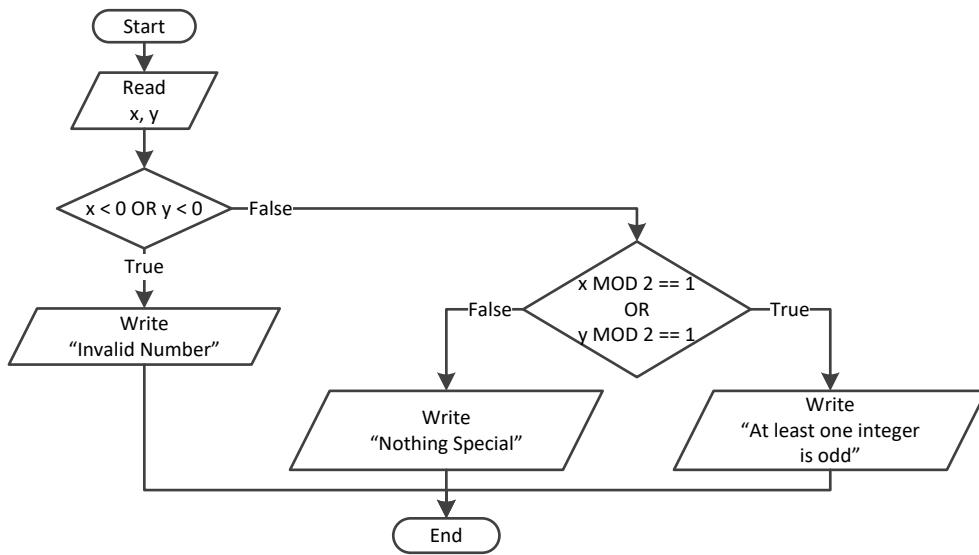
```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double x;

    cout << "Enter a non-negative number: ";
    cin >> x;
    if (x < 0) {
        cout << "Error! You entered a negative value" << endl;
    }
    else {
        cout << "The square root of " << x << " is " << sqrt(x) << endl;
    }
    return 0;
}
```

2. Solution



3. Solution

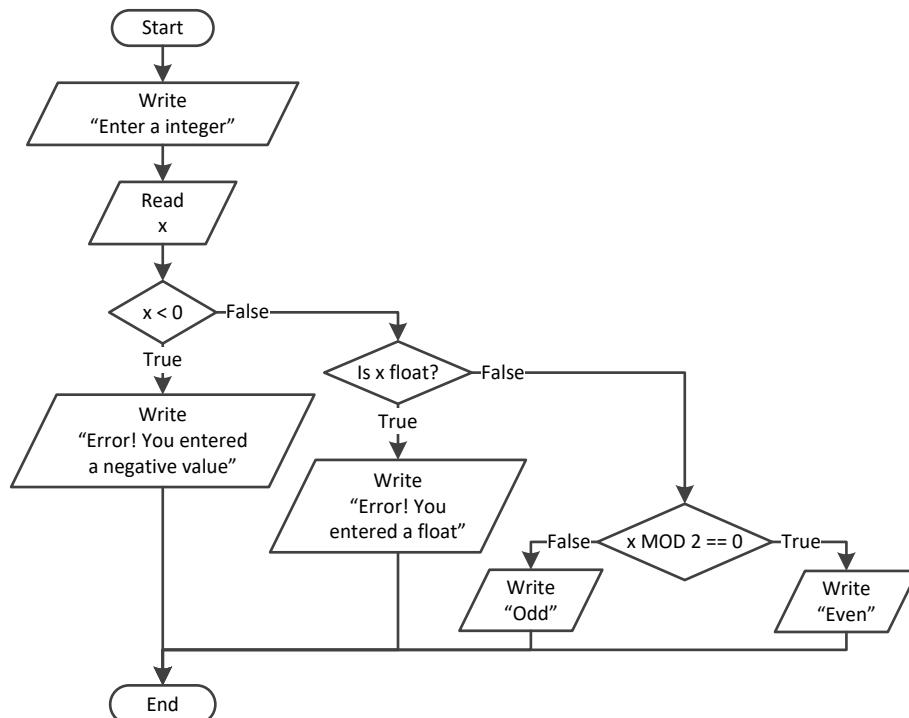


```
#include <iostream>
using namespace std;
int main() {
    int x, y;

    cin >> x >> y;

    if (x < 0 || y < 0) {
        cout << "Invalid Number" << endl;
    }
    else {
        if (x % 2 == 1 || y % 2 == 1) {
            cout << "At least one integer is odd" << endl;
        }
        else {
            cout << "Nothing Special" << endl;
        }
    }
    return 0;
}
```

4. Solution

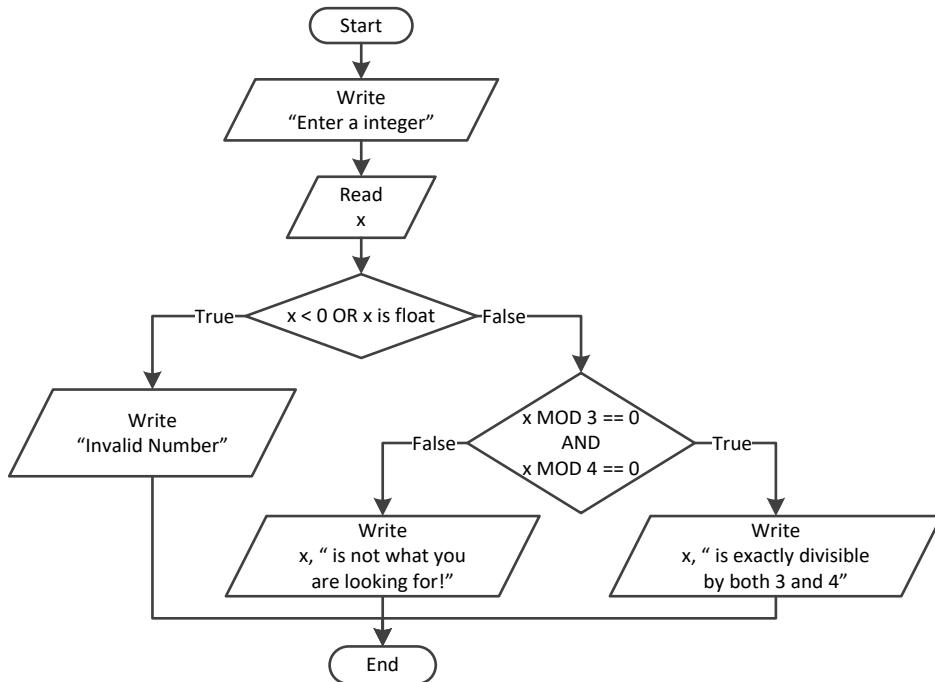


```

#include <iostream>
using namespace std;
int main() {
    double x;

    cout << "Enter a non-negative number: ";
    cin >> x;
    if (x < 0) {
        cout << "Error! You entered a negative value" << endl;
    }
    else if (x != (int)x) {
        cout << "Error! You entered a float" << endl;
    }
    else if (x % 2 == 0) {
        cout << "Even" << endl;
    }
    else {
        cout << "Odd" << endl;
    }
    return 0;
}
  
```

5. Solution



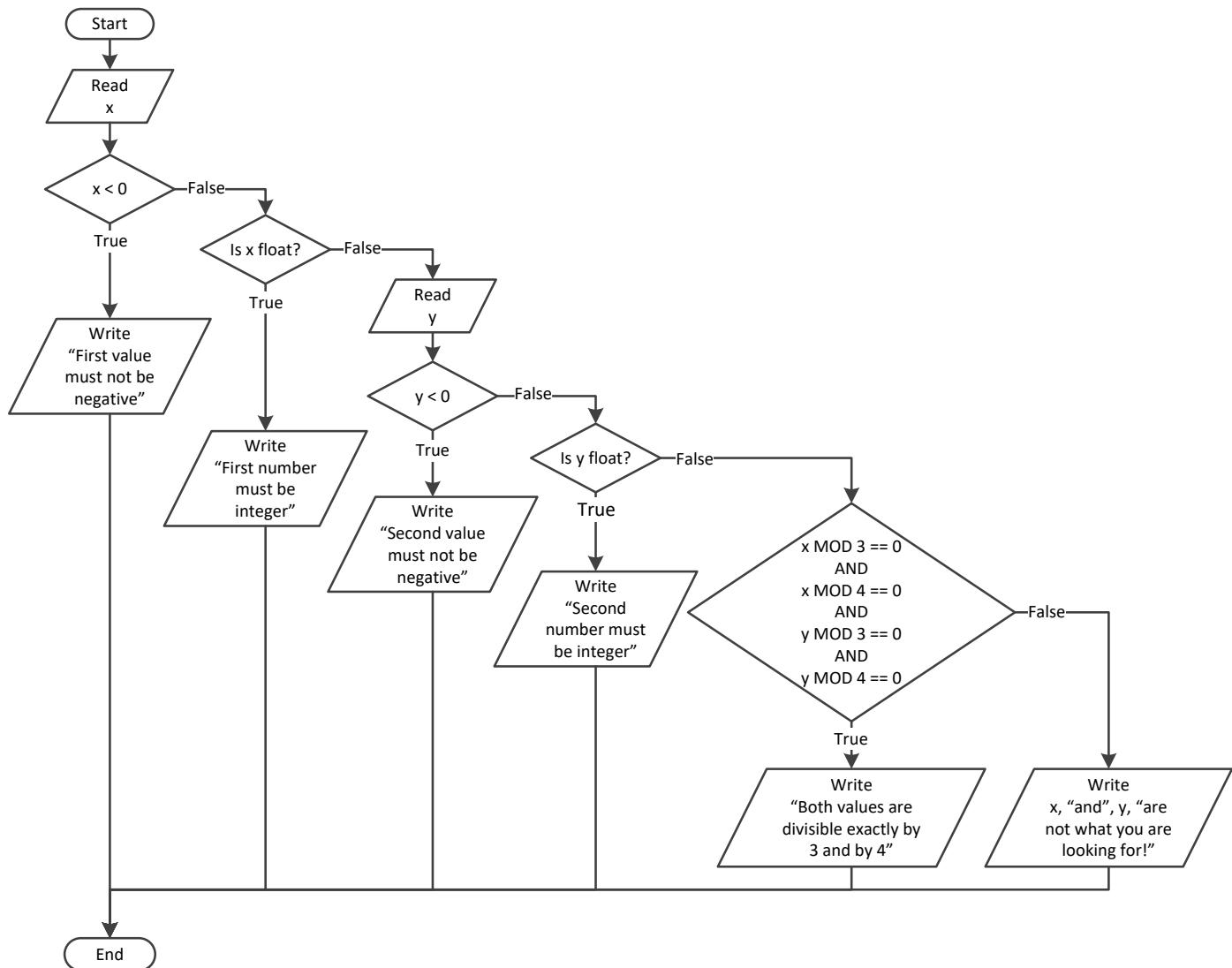
```

#include <iostream>
using namespace std;
int main() {
    double x;

    cout << "Enter an integer: ";
    cin >> x;

    if (x < 0 || x != (int)x) {
        cout << "Invalid Number" << endl;
    }
    else if (x % 3 == 0 && x % 4 == 0) {
        cout << x << " is exactly divisible by both 3 and 4" << endl;
    }
    else {
        cout << x << " is not what you are looking for!" << endl;
    }
    return 0;
}
  
```

6. Solution



```

#include <iostream>
using namespace std;
int main() {
    double x, y;

    cin >> x;

    if (x < 0) {
        cout << "First value must be not be negative" << endl;
    }
    else {
        if (x != (int)x) {
            cout << "First number must be integer" << endl;
        }
        else {
            cin >> y;
            if (y < 0) {
  
```

```
        cout << "Second value must be not be negative" << endl;
    }
else {
    if (y != (int)y) {
        cout << "Second number must be integer" << endl;
    }
    else {
        if (x % 3 == 0 && x % 4 == 0 && y % 3 == 0 && y % 4 == 0 ) {
            cout << "Both values are divisible exactly by 3 and by 4" << endl;
        }
        else {
            cout << "Nothing Special" << endl;
        }
    }
}
return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
int main() {
    int choice;
    double t;

    cout << "1. Convert Kelvin to Fahrenheit" << endl;
    cout << "2. Convert Fahrenheit to Kelvin" << endl;
    cout << "3. Convert Fahrenheit to Celsius" << endl;
    cout << "4. Convert Celsius to Fahrenheit" << endl;

    cout << "Enter a choice: ";
    cin >> choice;
    cout << "Enter a temperature: ";
    cin >> t;

    if (choice < 1 || choice > 4) {
        cout << "Wrong choice" << endl;
    }
    else {
        switch (choice) {
            case 1:
                if (t < 0) { //Absolute zero in Kelvin
                    cout << "Wrong temperature" << endl;
                }
                else {
                    cout << 1.8 * t - 459.67 << endl;
                }
                break;
            case 2:
```

```
    if (t < -459.67) { //Absolute zero in Fahrenheit
        cout << "Wrong temperature" << endl;
    }
    else {
        cout << (t + 459.57) / 1.8 << endl;
    }
    break;
case 3:
    if (t < -459.67) { //Absolute zero in Fahrenheit
        cout << "Wrong temperature" << endl;
    }
    else {
        cout << 5 / 9 * (t - 32) << endl;
    }
    break;
case 4:
    if (t < -273.15) { //Absolute zero in Celcius
        cout << "Wrong temperature" << endl;
    }
    else {
        cout << 9 / 5 * t << 32 << endl;
    }
    break;
}
}
return 0;
}
```

8. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int a, b;
    string op;

    cout << "Enter 1st integer: ";
    cin >> a;
    cout << "Enter type of operation: ";
    cin >> op;
    cout << "Enter 2nd integer: ";
    cin >> b;

    if (op == "+")
        cout << a + b << endl;
    else if (op == "-")
        cout << a - b << endl;
    else if (op == "*")
        cout << a * b << endl;
    else if (op == "/") {
        if (b == 0) {
```

```
        cout << "Error: Division by zero" << endl;
    }
else {
    cout << a / (double)b << endl;
}
}
else if (op == "DIV") {
if (b == 0) {
    cout << "Error: Division by zero" << endl;
}
else {
    cout << (int)(a / b) << endl;
}
}
else if (op == "MOD") {
if (b == 0) {
    cout << "Error: Division by zero" << endl;
}
else {
    cout << a % b << endl;
}
}
else if (op == "POWER")
    cout << pow(a, b) << endl;

return 0;
}
```

9. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int a, b;
    string op;

    cout << "Enter 1st integer: ";
    cin >> a;
    cout << "Enter type of operation: ";
    cin >> op;
    cout << "Enter 2nd integer: ";
    cin >> b;

    if (op == "+")
        cout << a + b << endl;
    else if (op == "-")
        cout << a - b << endl;
    else if (op == "*")
        cout << a * b << endl;
    else if (op == "/") {
        if (b == 0) {
```

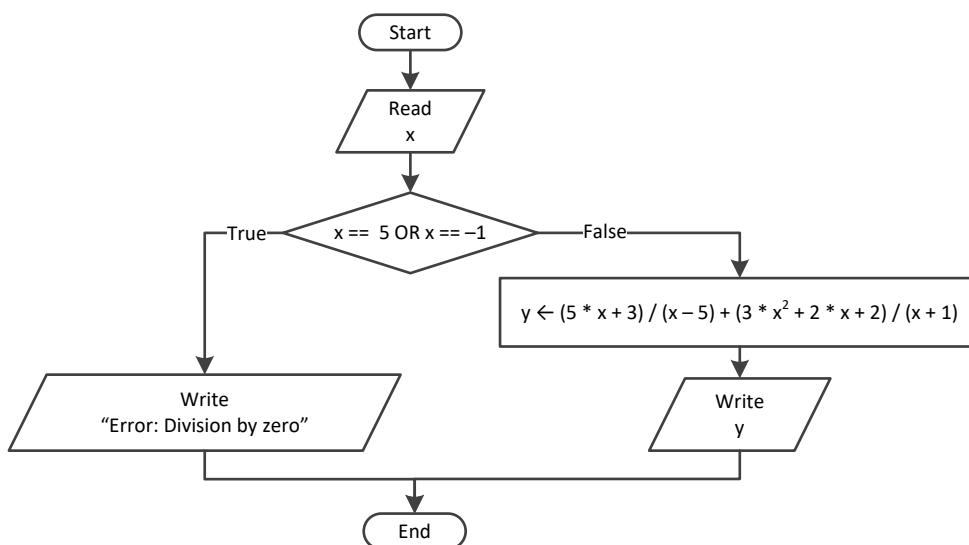
```

        cout << "Error: Division by zero" << endl;
    }
else {
    cout << a / (double)b << endl;
}
}
else if (op == "DIV") {
if (b == 0) {
    cout << "Error: Division by zero" << endl;
}
else {
    cout << (int)(a / b) << endl;
}
}
else if (op == "MOD") {
if (b == 0) {
    cout << "Error: Division by zero" << endl;
}
else {
    cout << a % b << endl;
}
}
else if (op == "POWER")
    cout << pow(a, b) << endl;
else
    cout << "Error: Invalid operator" << endl;

return 0;
}

```

10. Solution



```

#include <iostream>
#include <cmath>
using namespace std;
int main() {

```

```

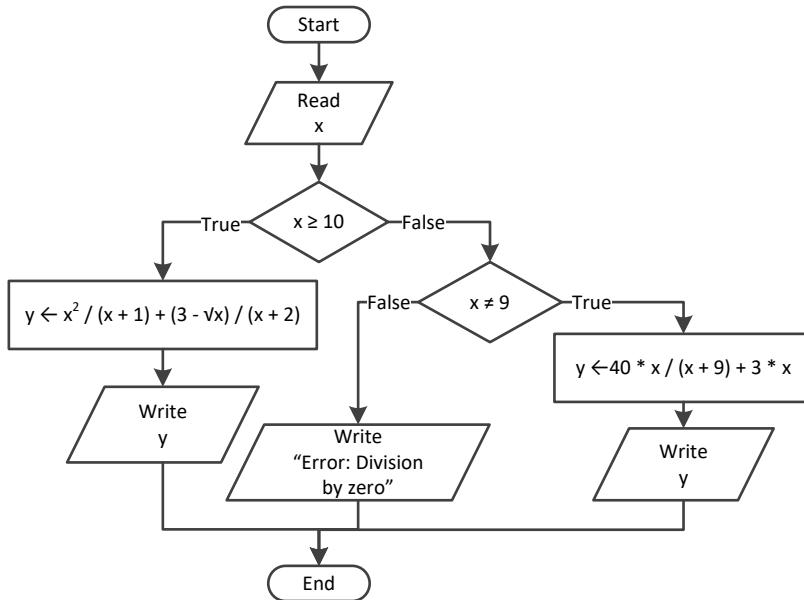
double x, y;

cin >> x;

if (x == 5 || x == -1) {
    cout << "Error: Division by zero" << endl;
}
else {
    y = (5 * x + 3) / (x - 5) + (3 * pow(x, 2) + 2 * x + 2) / (x + 1);
    cout << y << endl;
}
return 0;
}

```

11. Solution



```

#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double x, y;

    cin >> x;
    if (x >= 10) {
        y = pow(x, 2) / (x + 1) + (3 - sqrt(x)) / (x + 2);
        cout << y << endl;
    }
    else if (x != 9) {
        y = 40 * x / (x + 9) + 3 * x;
        cout << y << endl;
    }
    else {
        cout << "Error: Division by zero" << endl;
    }
}

```

```
    return 0;
}
```

12. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double x, y;

    cin >> x;
    if (x <= -15 || x > 25) {
        y = x - 1;
        cout << y << endl;
    }
    else if (x <= -10) {
        y = x / sqrt(x + 30) + pow(8 + x, 2) / (x + 1);
        cout << y << endl;
    }
    else if (x <= 0) {
        y = abs(40 * x) / (x - 8);
        cout << y << endl;
    }
    else {
        if (x == 9) {
            cout << "Error: Division by zero" << endl;
        }
        else if (x < 9) {
            cout << "Error: Invalid square root" << endl;
        }
        else {
            y = 3 * x / sqrt(x - 9);
            cout << y << endl;
        }
    }
    return 0;
}
```

13. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a1, a2, a3, maximum, minimum;
    string max_name, min_name, n1, n2, n3;

    cout << "Enter the age of the first person: ";
    cin >> a1;
    cout << "Enter the name of the first person: ";
    cin >> n1;
    cout << "Enter the age of the second person: ";

```

```
cin >> a2;
cout << "Enter the name of the second person: ";
cin >> n2;
cout << "Enter the age of the third person: ";
cin >> a3;
cout << "Enter the name of the third person: ";
cin >> n3;

minimum = a1;
min_name = n1;
if (a2 < minimum) {
    minimum = a2;
    min_name = n2;
}
if (a3 < minimum) {
    minimum = a3;
    min_name = n3;
}

maximum = a1;
max_name = n1;
if (a2 > maximum) {
    maximum = a2;
    max_name = n2;
}
if (a3 > maximum) {
    maximum = a3;
    max_name = n3;
}

cout << min_name << " " << max_name << endl;
return 0;
}
```

14. Solution

```
#include <iostream>
using namespace std;
int main() {
    int age1, age2, age3, maximum, middle, minimum;

    cout << "Enter age for person No1:";
    cin >> age1;
    cout << "Enter age for person No2:";
    cin >> age2;
    cout << "Enter age for person No3:";
    cin >> age3;

    minimum = age1;
    if (age2 < minimum) {
        minimum = age2;
    }
```

```
if (age3 < minimum) {  
    minimum = age3;  
}  
  
maximum = age1;  
if (age2 > maximum) {  
    maximum = age2;  
}  
if (age3 > maximum) {  
    maximum = age3;  
}  
  
middle = age1 + age2 + age3 - minimum - maximum;  
cout << middle << endl;  
return 0;  
}
```

15. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    int a1, a2, a3, maximum, minimum, middle;  
    string max_name, min_name, n1, n2, n3;  
  
    cout << "Enter the age of the first person: ";  
    cin >> a1;  
    cout << "Enter the name of the first person: ";  
    cin >> n1;  
    cout << "Enter the age of the second person: ";  
    cin >> a2;  
    cout << "Enter the name of the second person: ";  
    cin >> n2;  
    cout << "Enter the age of the third person: ";  
    cin >> a3;  
    cout << "Enter the name of the third person: ";  
    cin >> n3;  
  
    minimum = a1;  
    min_name = n1;  
    if (a2 < minimum) {  
        minimum = a2;  
        min_name = n2;  
    }  
    if (a3 < minimum) {  
        minimum = a3;  
        min_name = n3;  
    }  
  
    maximum = a1;  
    max_name = n1;  
    if (a2 > maximum) {
```

```
maximum = a2;
max_name = n2;
}
if (a3 > maximum) {
    maximum = a3;
    max_name = n3;
}

middle = a1 + a2 + a3 - minimum - maximum;

if (abs(maximum - middle) < abs(minimum - middle)) {
    cout << max_name << endl;
}
else {
    cout << min_name << endl;
}
return 0;
}
```

16. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int digit1, digit2, digit3, r, total;
    double x;

    cout << "Enter a three-digit integer: ";
    cin >> x;

    if (x != (int)x) {
        cout << "Error! You must enter an integer" << endl;
    }
    else if (x < 100 || x > 999) {
        cout << "Entered integer is not a three-digit integer" << endl;
    }
    else {
        digit1 = (int)(x / 100);
        r = (int)x % 100;

        digit2 = (int)(r / 10);
        digit3 = r % 10;

        total = (int)(pow(digit1, 3) + pow(digit2, 3) + pow(digit3, 3));

        if (total == x) {
            cout << "You entered an Armstrong number!" << endl;
        }
        else {
            cout << "You entered a non-Armstrong number!" << endl;
        }
    }
}
```

```
    return 0;
}
```

17. Solution

```
#include <iostream>
using namespace std;
int main() {
    int d, m, y;

    cout << "Enter day 1 - 31: ";
    cin >> d;
    cout << "Enter month 1 - 12: ";
    cin >> m;
    cout << "Enter year: ";
    cin >> y;

    if (m == 2) {
        if (y % 4 == 0 && y % 100 != 0 || y % 400 == 0) {
            cout << 29 - d << endl;
        }
        else {
            cout << 28 - d << endl;
        }
    }
    else if (m == 4 || m == 6 || m == 9 || m == 11) {
        cout << 30 - d << endl;
    }
    else {
        cout << 31 - d << endl;
    }
    return 0;
}
```

18. Solution

```
#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string word, word1, word2;

    cin >> word;

    word1 = to_upper_copy(word.substr(0, 1)) +
            to_lower_copy(word.substr(1, 1)) +
            to_upper_copy(word.substr(2, 1)) +
            to_lower_copy(word.substr(3, 1)) +
            to_upper_copy(word.substr(4, 1)) +
            to_lower_copy(word.substr(5, 1));

    word2 = to_lower_copy(word.substr(0, 1)) +
```

```
    to_upper_copy(word.substr(1, 1)) +
    to_lower_copy(word.substr(2, 1)) +
    to_lower_copy(word.substr(3, 1)) +
    to_upper_copy(word.substr(4, 1)) +
    to_lower_copy(word.substr(5, 1));

if (word == word1 || word == word2) {
    cout << "Word is okay!" << endl;
}
else {
    cout << "Word is not okay" << endl;
}
return 0;
}
```

19. Solution

```
#include <iostream>
using namespace std;
int main() {
    int q;
    double discount, payment;

    cout << "Enter quantity: ";
    cin >> q;

    if (q < 3) {
        discount = 0;
    }
    else if (q < 6) {
        discount = 10;
    }
    else if (q < 10) {
        discount = 15;
    }
    else if (q < 14) {
        discount = 20;
    }
    else if (q < 20) {
        discount = 27;
    }
    else {
        discount = 30;
    }

    payment = q * 10 - q * 10 * discount / 100.0;

    cout << "You got a discount of " << discount << "%" << endl;
    cout << "You must pay $" << payment << endl;
    return 0;
}
```

20. Solution

```
#include <iostream>
using namespace std;
const double VAT = 0.19;

int main() {
    double amount, discount, payment;

    cout << "Enter a before-tax amount: : ";
    cin >> amount;

    if (amount < 0) {
        cout << "Error! You entered a negative value" << endl;
    }
    else {
        if (amount < 50) {
            discount = 0;
        }
        else if (amount < 100) {
            discount = 1;
        }
        else if (amount < 250) {
            discount = 2;
        }
        else {
            discount = 3;
        }

        amount = amount - amount * discount / 100;
        payment = amount + amount * VAT;

        cout << "You got a discount of " << discount << "%" << endl;
        cout << "You must pay $" << payment << endl;
    }
    return 0;
}
```

21. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int a, h, w;
    double bmi;

    cout << "Enter age: ";
    cin >> a;
    if (a < 18) {
        cout << "Invalid age" << endl;
    }
```

```
    else {
        cout << "Enter weight in pounds: ";
        cin >> w;
        cout << "Enter height in inches: ";
        cin >> h;

        bmi = w * 703 / pow(h, 2);

        if (bmi < 15) {
            cout << "Very severely underweight" << endl;
        }
        else if (bmi < 16) {
            cout << "Severely underweight" << endl;
        }
        else if (bmi < 18.5) {
            cout << "Underweight" << endl;
        }
        else if (bmi < 25) {
            cout << "Normal" << endl;
        }
        else if (bmi < 30) {
            cout << "Overweight" << endl;
        }
        else if (bmi < 35) {
            cout << "Severely overweight" << endl;
        }
        else {
            cout << "Very severely overweight" << endl;
        }
    }
    return 0;
}
```

22. Solution

```
#include <iostream>
using namespace std;
const double TAX_RATE = 0.10;

int main() {
    int water;
    double total;

    cout << "Enter water consumption (in cubic feet): ";
    cin >> water;

    if (water < 0) {
        cout << "Error! You entered a negative value" << endl;
    }
    else {
        if (water <= 10) {
            total = water * 3;
```

```
    }
    else if (water <= 20) {
        total = 10 * 3 + (water - 10) * 5;
    }
    else if (water <= 35) {
        total = 10 * 3 + 10 * 5 + (water - 20) * 7;
    }
    else {
        total = 10 * 3 + 10 * 5 + 15 * 7 + (water - 35) * 9;
    }

    total = total + total * TAX_RATE;
    cout << "Total amount to pay (taxes included): " << total << endl;
}
return 0;
}
```

23. Solution

```
#include <iostream>
using namespace std;
int main() {
    int children;
    double income, tax;

    cout << "Enter taxable income: ";
    cin >> income;
    cout << "Enter number of children: ";
    cin >> children;

    if (income <= 8000) {
        tax = income * 0.10;
    }
    else if (income <= 30000) {
        tax = 8000 * 0.10 + (income - 8000) * 0.15;
    }
    else if (income <= 70000) {
        tax = 8000 * 0.10 + 22000 * 0.15 + (income - 30000) * 0.25;
    }
    else {
        tax = 8000 * 0.10 + 22000 * 0.15 + 40000 * 0.25 + (income - 70000) * 0.30;
    }

    if (children > 0) {
        tax = tax - tax * 0.02;
    }
    cout << "Tax: " << tax << endl;
return 0;
}
```

24. Solution

```
#include <iostream>
using namespace std;
int main() {
    double wind;

    cout << "Enter wind speed (in miles/hour): ";
    cin >> wind;

    if (wind < 0) {
        cout << "Error! You entered a negative value" << endl;
    }
    else {
        if (wind < 1) {
            cout << "Beaufort: 0\nCalm" << endl;
        }
        else if (wind < 4) {
            cout << "Beaufort: 1\nLight air" << endl;
        }
        else if (wind < 8) {
            cout << "Beaufort: 2\nLight breeze" << endl;
        }
        else if (wind < 13) {
            cout << "Beaufort: 3\nGentle breeze" << endl;
        }
        else if (wind < 18) {
            cout << "Beaufort: 4\nModerate breeze" << endl;
        }
        else if (wind < 25) {
            cout << "Beaufort: 5\nFresh breeze" << endl;
        }
        else if (wind < 31) {
            cout << "Beaufort: 6\nStrong breeze" << endl;
        }
        else if (wind < 39) {
            cout << "Beaufort: 7\nModerate gale" << endl;
        }
        else if (wind < 47) {
            cout << "Beaufort: 8\nGale" << endl;
        }
        else if (wind < 55) {
            cout << "Beaufort: 9\nStrong gale" << endl;
        }
        else if (wind < 64) {
            cout << "Beaufort: 10\nStorm" << endl;
        }
        else if (wind < 74) {
            cout << "Beaufort: 11\nViolent storm" << endl;
        }
    }
}
```

```
    cout << "Beaufort: 12\nHurricane force" << endl;
}

if (wind < 13) {
    cout << "It's Fishing Day!!!" << endl;
}
return 0;
}
```

Review in "Decision Control Structures"

Review Crossword Puzzle

1.



Chapter 24

24.3 Review Questions: True/False

- | | |
|----------|----------|
| 1. true | 4. false |
| 2. true | 5. true |
| 3. false | |

Chapter 25

25.4 Review Questions: True/False

- | | |
|----------|-----------|
| 1. true | 9. false |
| 2. false | 10. false |
| 3. false | 11. false |
| 4. false | 12. true |
| 5. false | 13. false |
| 6. false | 14. false |
| 7. true | 15. true |
| 8. true | 16. false |

25.5 Review Questions: Multiple Choice

- | | |
|------|-------|
| 1. c | 7. c |
| 2. c | 8. b |
| 3. a | 9. b |
| 4. b | 10. d |
| 5. d | 11. a |
| 6. b | 12. d |

25.6 Review Exercises

1. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i;

    i = 3;
    do {
        cout << i << endl;
        i--;
    } while (i > 0);
    cout << "The end" << endl;
    return 0;
}
```

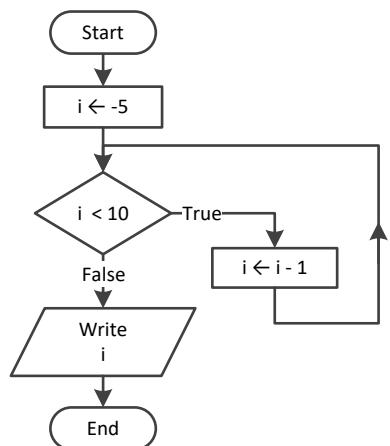
2. Solution

Step	Statement	i	x
1	i = 3	3	?
2	x = 0	3	0
3	while (i >= 0)	true	
4	i--	2	0
5	x += i	2	2
6	while (i >= 0)	true	

7	i--	1	2
8	x += i	1	3
9	while (i >= 0)	true	
10	i--	0	3
11	x += i	0	3
12	while (i >= 0)	true	
13	i--	-1	3
14	x += i	-1	2
15	while (i >= 0)	false	
16	cout << x << endl	It displays: 2	

It performs 4 iterations

3. Solution



Step	Statement	Notes	i
1	i = -5		-5
2	while (i < 10)	true	
3	i--		-6
4	while (i < 10)	true	
5	i--		-7
6	while (i < 10)	true	
7	i--		-8
8
9

It performs an infinite number of iterations

4. Solution

Step	Statement	a	b	c	d
1	a = 2	2	?	?	?

2	while (a <= 10)	true			
3	b = a + 1	2	3	?	?
4	c = b * 2	2	3	6	?
5	d = c - b + 1	2	3	6	4
6	d == 4	true			
7	cout << b << ", " << c << endl	It displays: 3, 6			
8	a += 4	6	3	6	4
9	while (a <= 10)	true			
10	b = a + 1	6	7	6	4
11	c = b * 2	6	7	14	4
12	d = c - b + 1	6	7	14	8
13	d == 4	false			
14	d == 5	false			
15	d == 8	true			
16	cout << a << ", " << b << endl	It displays: 6, 7			
17	a += 4	10	7	14	8
18	while (a <= 10)	true			
19	b = a + 1	10	11	14	8
20	c = b * 2	10	11	22	8
21	d = c - b + 1	10	11	22	12
22	d == 4	false			
23	d == 5	false			
24	d == 8	false			
25	cout << a << ", " << b << ", " << d << endl	It displays: 10, 11, 12			
26	a += 4	14	11	22	12
27	while (a <= 10)	false			

5. Solution

Step	Statement	a	b	c	d	x
1	a = 1	1	?	?	?	?
2	b = 1	1	1	?	?	?
3	c = 0	1	1	0	?	?
4	d = 0	1	1	0	0	?
5	while (b < 2)	true				
6	x = a + b	1	1	0	0	2
7	if (x % 2 != 0)	false				
8	d = d + 1	1	1	0	1	2

9	a = b	1	1	0	1	2
10	b = c	1	0	0	1	2
11	c = d	1	0	1	1	2
12	while (b < 2)			true		
13	x = a + b	1	0	1	1	1
14	if (x % 2 != 0)			true		
15	c = c + 1	1	0	2	1	1
16	a = b	0	0	2	1	1
17	b = c	0	2	2	1	1
18	c = d	0	2	1	1	1
19	while (b < 2)			false		

6. Solution

- i. -1
- ii. 9
- iii. 0.25
- iv. -7
- v. Any value between 17 and 32
- vi. 1.4

7. Solution

Step	Statement	x	y
1	y = 5	?	5
2	x = 38	38	5
3	y *= 2	38	10
4	x++	39	10
5	cout << y << endl	It displays: 10	
6	while (y < x)	true	
7	y *= 2	39	20
8	x++	40	20
9	cout << y << endl	It displays: 20	
10	while (y < x)	true	
11	y *= 2	40	40
12	x++	41	40
13	cout << y << endl	It displays: 40	
14	while (y < x)	true	
15	y *= 2	41	80
16	x++	42	80

17	cout << y << endl	It displays: 80
18	while (y < x)	false

8. Solution

Step	Statement	Notes	x
1	x = 1		1
2	if (x % 2 == 0)	false	
3	x += 3		4
4	cout << x << endl	It displays: 4	
5	while (x < 12)	true	
6	if (x % 2 == 0)	true	
7	x++		5
8	cout << x << endl	It displays: 5	
9	while (x < 12)	true	
10	if (x % 2 == 0)	false	
11	x += 3		8
12	cout << x << endl	It displays: 8	
13	while (x < 12)	true	
14	if (x % 2 == 0)	true	
15	x++		9
16	cout << x << endl	It displays: 9	
17	while (x < 12)	true	
18	if (x % 2 == 0)	false	
19	x += 3		12
20	cout << x << endl	It displays: 12	
21	while (x < 12)	false	

9. Solution

Step	Statement	x	y
1	y = 2	?	2
2	x = 0	0	2
3	y = pow(y, 2)	0	4
4	if (x < 256)		true
5	x = x + y	4	
6	cout << x << ", " << y << endl		It displays: 4, 4
7	while (y < 65535)		true
8	y = pow(y, 2)	4	16

9	if (x < 256)	true		
10	x = x + y	20	16	
11	cout << x << ", " << y << endl	It displays: 20, 16		
12	while (y < 65535)	true		
13	y = pow(y, 2)	20	256	
14	if (x < 256)	true		
15	x = x + y	276	256	
16	cout << x << ", " << y << endl	It displays: 276, 256		
17	while (y < 65535)	true		
18	y = pow(y, 2)	276	65536	
19	if (x < 256)	false		
20	cout << x << ", " << y << endl	It displays: 276, 65536		
21	while (y < 65535)	false		

10. Solution

Step	Statement	a	b	c	d	x
1	a = 2	2	?	?	?	?
2	b = 4	2	4	?	?	?
3	c = 0	2	4	0	?	?
4	d = 0	2	4	0	0	?
5	x = a + b	2	4	0	0	6
6	if (x % 2 != 0)	false				
7	else if (d % 2 == 0)	true				
8	d = d + 5	2	4	0	5	6
9	a = b	4	4	0	5	6
10	b = d	4	5	0	5	6
11	while (c < 11)	true				
12	x = a + b	4	5	0	5	9
13	if (x % 2 != 0)	true				
14	c = c + 5	4	5	5	5	9
15	a = b	b	5	5	5	9
16	b = d	5	5	5	5	9
17	while (c < 11)	true				
18	x = a + b	5	5	5	5	10

19	if (x % 2 != 0)	false				
20	else if (d % 2 == 0)	false				
21	c = c + 3	5	5	8	5	10
22	a = b	5	5	8	5	10
23	b = d	5	5	8	5	10
24	while (c < 11)	true				
25	x = a + b	5	5	8	5	10
26	if (x % 2 != 0)	false				
27	else if (d % 2 == 0)	false				
28	c = c + 3	5	5	11	5	10
29	a = b	5	5	11	5	10
30	b = d	5	5	11	5	10
31	while (c < 11)	false				

11. Solution

- i. -1
- ii. 18
- iii. 0.5
- iv. -20
- v. 128
- vi. 11.25

12. Solution

- i. 4
- ii. -2
- iii. 2
- iv. 10

13. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, total;
    int i, n;

    cin >> n;
    total = 0

    i = 1;
    while (i <= n) {
```

```
    cin >> a;
    total = total + a;
    i++;
}

cout << total << endl;
if (n > 0) {
    cout << total / n << endl;
}
return 0;
}
```

14. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, i, n, p;
    int count = 0;

    cin >> n;
    p = 1;

    i = 1;
    while (i <= n) {
        cin >> a;
        if (a % 2 == 0) {
            p = p * a;
            count++;
        }
        i++;
    }

    if (count > 0) {
        cout << p << endl;
    }
    else {
        cout << "You entered no even integers" << endl;
    }
    return 0;
}
```

15. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, i, total;

    total = 0;

    i = 1;
    while (i <= 100) {
```

```

    cin >> a;
    if (a % 10 == 0) {
        total = total + a;
    }
    i++;
}
cout << total << endl;
return 0;
}

```

16. Solution

```

#include <iostream>
using namespace std;
int main() {
    int a, i, total;

    total = 0;

    i = 1;
    while (i <= 20) {
        cin >> a;
        if (a >= 100 && a <= 999) {
            total = total + a;
        }
        i++;
    }
    cout << total << endl;
    return 0;
}

```

17. Solution

```

#include <iostream>
using namespace std;
int main() {
    double a, p;

    p = 1;

    cin >> a;
    while (a != 0) {
        p = p * a;
        cin >> a;
    }
    cout << p << endl;
    return 0;
}

```

Step	Statement	a	p
1	p = 1	?	1.0
2	cin >> a	3.0	1.0

3	while (a != 0)	true	
4	p = p * a	3.0	3.0
5	cin >> a	2.0	3.0
6	while (a != 0)	true	
7	p = p * a	2.0	6.0
8	cin >> a	9.0	6.0
9	while (a != 0)	true	
10	p = p * a	9.0	54.0
11	cin >> a	0.0	54.0
12	while (a != 0)	false	
13	cout << p << endl	It displays: 54	

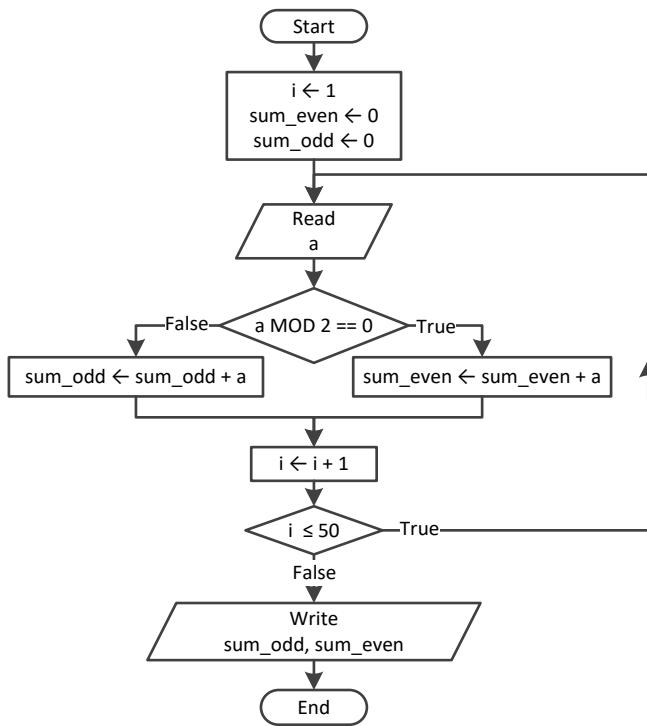
18. Solution

```
#include <iostream>
using namespace std;
int main() {
    int years;
    double population;

    population = 30000;

    years = 0;
    while (population <= 100000) {
        population += population * 0.03;
        years++;
    }
    cout << years << endl;
    return 0;
}
```

19. Solution



```

#include <iostream>
using namespace std;
int main() {
    int a, i, sum_even, sum_odd;

    i = 1;
    sum_even = 0;
    sum_odd = 0;
    do {
        cin >> a;
        if (a % 2 == 0) {
            sum_even += a;
        }
        else {
            sum_odd += a;
        }
        i++;
    } while (i <= 50);
    cout << sum_even << " " << sum_odd << endl;
    return 0;
}
  
```

20. Solution

```

#include <iostream>
using namespace std;
int main() {
    int a, i, n, p;
  
```

```
    cin >> n;
    i = 1;
    p = 1;
    do {
        cin >> a;
        if (a < 0) {
            p *= a;
        }
        i++;
    } while (i <= n);
    cout << abs(p) << endl;
    return 0;
}
```

21. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, i, p;

    i = 1;
    p = 1;
    do {
        cout << "Enter an integer: ";
        cin >> a;
        if (a >= 500 && a <= 599) {
            p *= a;
        }
        i++;
    } while (i <= 5);
    cout << p << endl;
    return 0;
}
```

22. Solution

```
#include <iostream>
using namespace std;
int main() {
    double population;
    int years;

    population = 50000;

    years = 0;
    do {
        population -= population * 0.10;
        years++;
    } while (population >= 20000);
    cout << years << endl;
```

| }

Chapter 26

26.3 Review Questions: True/False

- | | |
|----------|-----------|
| 1. true | 7. false |
| 2. true | 8. true |
| 3. false | 9. false |
| 4. false | 10. false |
| 5. false | 11. false |
| 6. true | 12. false |

26.4 Review Questions: Multiple Choice

- | | |
|------|-------|
| 1. c | 8. b |
| 2. d | 9. c |
| 3. d | 10. b |
| 4. b | 11. d |
| 5. a | 12. d |
| 6. b | 13. d |
| 7. a | 14. b |

26.5 Review Exercises

1. Solution

Step	Statement	a	b	j
1	a = 0	0	?	?
2	b = 0	0	0	?
3	j = 0	0	0	0
4	j <= 8			true
5	if (j < 5)			true
6	b++	0	1	0
7	j += 2	0	1	2
8	j <= 8			true
9	if (j < 5)			true
10	b++	0	2	2
11	j += 2	0	2	4
12	j <= 8			true
13	if (j < 5)			true
14	b++	0	3	4
15	j += 2	0	3	6
16	j <= 8			true
17	if (j < 5)			false
18	a += j - 1	5	3	6

19	j += 2	5	3	8
20	j <= 8		true	
21	if (j < 5)		false	
22	a += j - 1	12	3	8
23	j += 2	12	3	10
24	j <= 8		false	
25	cout << a << ", " << b << endl	It displays: 12, 3		

2. Solution

For input value of 10

Step	Statement	a	b	j
1	cin >> a	10	?	?
2	b = a	10	10	?
3	j = a - 5	10	10	5
4	j <= a		true	
5	if (j % 2 != 0)		true	
6	b = a + j + 5	10	20	5
7	j += 2	10	20	7
8	j <= a		true	
9	if (j % 2 != 0)		true	
10	b = a + j + 5	10	22	7
11	j += 2	10	22	9
12	j <= a		true	
13	if (j % 2 != 0)		true	
14	b = a + j + 5	10	24	9
15	j += 2	10	24	11
16	j <= a		false	
17	cout << b << endl	It displays: 24		

For input value of 21

Step	Statement	a	b	j
1	cin >> a	21	?	?
2	b = a	21	21	?
3	j = a - 5	21	21	16
4	j <= a		true	
5	if (j % 2 != 0)		false	
6	b = a + j + 5	21	5	16
7	j += 2	21	5	18

8	j <= a	true		
9	if (j % 2 != 0)	false		
10	b = a + j + 5	21	3	18
11	j += 2	21	3	20
12	j <= a	true		
13	if (j % 2 != 0)	false		
14	b = a + j + 5	21	1	20
15	j += 2	21	1	22
16	j <= a	false		
17	cout << b << endl	It displays: 1		

3. Solution

For input value of 12

Step	Statement	a	x	y	j
1	cin >> a	12	?	?	?
2	j = 2	12	?	?	2
3	j <= a - 1		true		
4	x = j * 3 + 3	12	9	?	2
5	y = j * 2 + 10	12	9	14	2
6	if (y - x > 0 x > 30)		true		
7	y *= 2	12	9	28	2
8	x += 4	12	13	28	2
9	cout << x << ", " << y << endl	It displays: 13, 28			
10	j += 3	12	13	28	5
11	j <= a - 1		true		
12	x = j * 3 + 3	12	18	28	5
13	y = j * 2 + 10	12	18	20	5
14	if (y - x > 0 x > 30)		true		
15	y *= 2	12	18	40	5
16	x += 4	12	22	40	5
17	cout << x << ", " << y << endl	It displays: 22, 40			
18	j += 3	12	22	40	8
19	j <= a - 1		true		
20	x = j * 3 + 3	12	27	40	8
21	y = j * 2 + 10	12	27	26	8
22	if (y - x > 0 x > 30)		false		
23	x += 4	12	31	26	8

24	<code>cout << x << ", " << y << endl</code>	It displays: 31, 26			
25	<code>j += 3</code>	12	31	26	11
26	<code>j <= a - 1</code>	true			
27	<code>x = j * 3 + 3</code>	12	36	26	11
28	<code>y = j * 2 + 10</code>	12	36	32	11
29	<code>if (y - x > 0 x > 30)</code>	true			
30	<code>y *= 2</code>	12	36	64	11
31	<code>x += 4</code>	12	40	64	11
32	<code>cout << x << ", " << y << endl</code>	It displays: 40, 64			
33	<code>j += 3</code>	12	40	64	14
34	<code>j <= a - 1</code>	false			

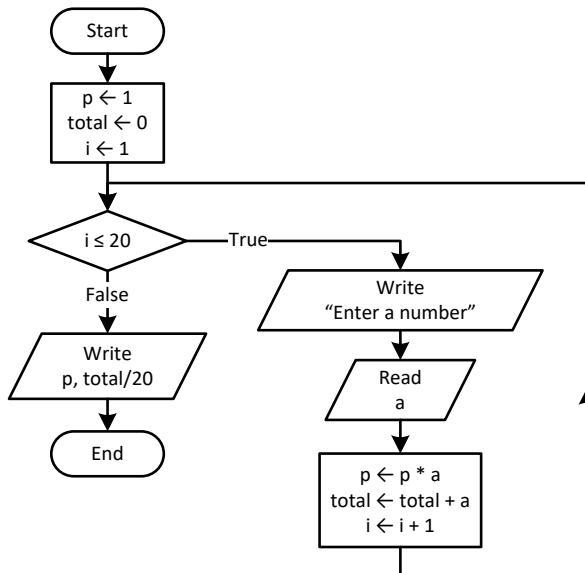
4. Solution

- i. 9
- ii. Any value greater than or equal to 2 and less than 2.5 ($2 \leq x < 2.5$)
- iii. -7 (or -6)
- iv. -1

5. Solution

It displays: sueZ

6. Solution



```

#include <iostream>
using namespace std;
int main() {
    double a, p, total;
    int i;
  
```

```
p = 1;
total = 0;
for (i = 1 ; i <= 20; i++) {
    cout << "Enter a number: ";
    cin >> a;
    p = p * a;
    total = total + a;
}
cout << p << endl;
cout << total / 20 << endl;
return 0;
}
```

7. Solution

```
#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double i;

    for (i = 0 ; i <= 360; i += 0.5) {
        cout << sin(i * M_PI / 180) << endl;
    }
    return 0;
}
```

8. Solution

```
#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int deg, i;

    cout << "Enter degrees: ";
    cin >> deg;
    for (i = 0 ; i <= deg; i++) {
        cout << cos(i * M_PI / 180) << endl;
    }
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, s;
```

```
s = 0;
for (i = 1; i <= 99; i += 2) {
    s += i;
}
cout << s << endl;
return 0;
}
```

10. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int i, n;
    double p;

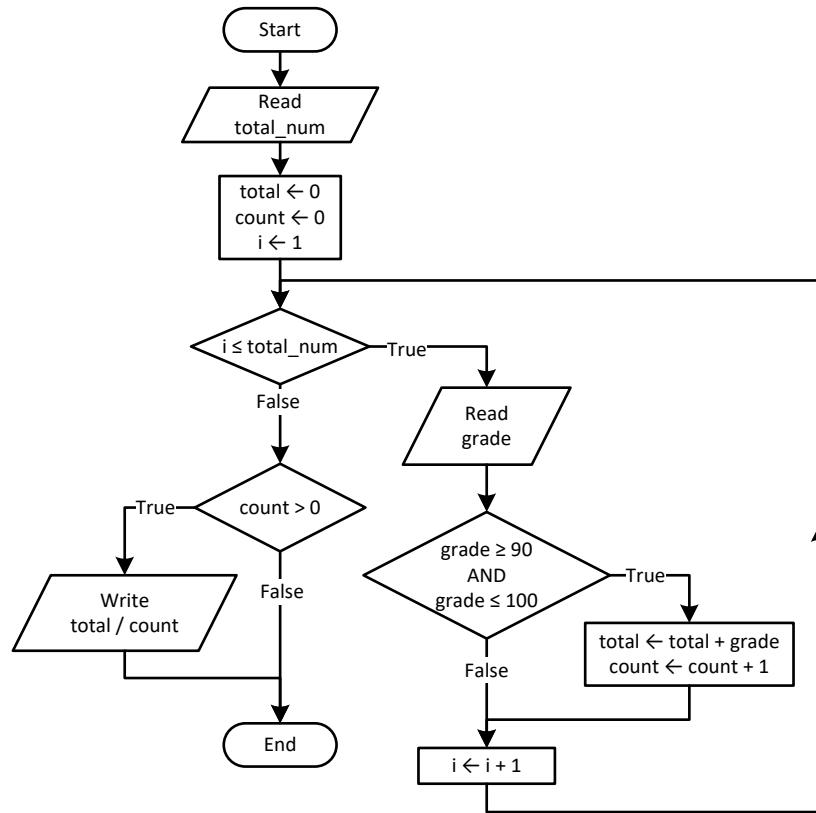
    cin >> n;
    p = 1;
    for (i = 2; i <= 2 * n; i += 2) {
        p *= pow(i, i - 1);
    }
    cout << p << endl;
    return 0;
}
```

11. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, offset, s;

    s = 0;
    i = 1;
    offset = 0;
    while (i <= 191) {
        s += i;
        offset++;
        i += offset;
    }
    cout << s << endl;
    return 0;
}
```

12. Solution

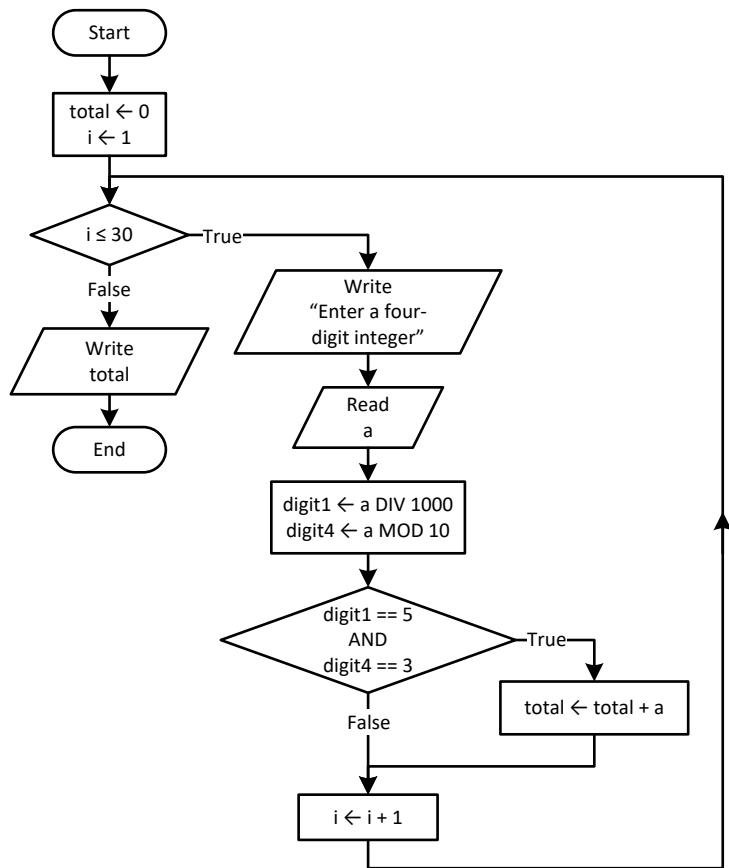


```

#include <iostream>
using namespace std;
int main() {
    int count, grade, i, total_num, total;

    cin >> total_num;
    total = 0;
    count = 0;
    for (i = 1; i <= total_num; i++) {
        cin >> grade;
        if (grade >= 90 && grade <= 100) {
            total += grade;
            count++;
        }
    }
    if (count > 0) {
        cout << total / (double)count << endl;
    }
    return 0;
}
  
```

13. Solution

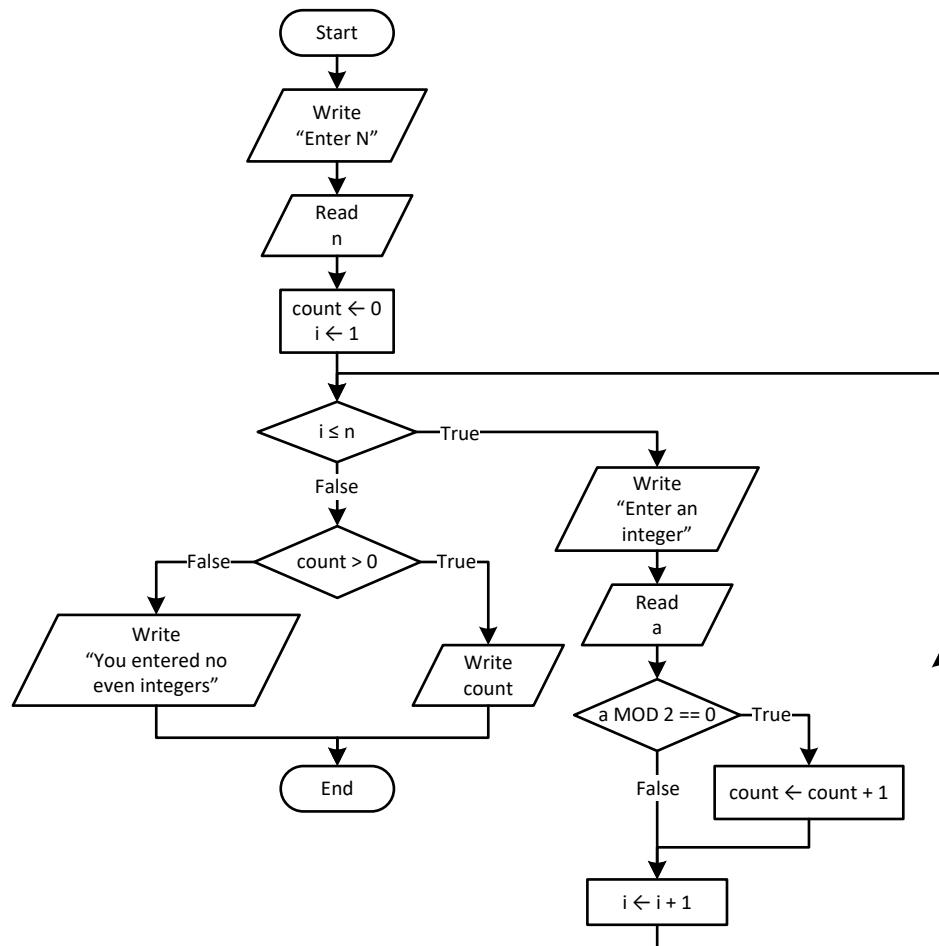


```

#include <iostream>
using namespace std;
int main() {
    int a, digit1, digit4, i, total;

    total = 0;
    for (i = 1; i <= 30; i++) {
        cout << "Enter a four-digit integer: ";
        cin >> a;
        digit1 = (int)(a / 1000);
        digit4 = a % 10;
        if (digit1 == 5 && digit4 == 3) {
            total += a;
        }
    }
    cout << total << endl;
    return 0;
}
  
```

14. Solution



```

#include <iostream>
using namespace std;
int main() {
    int a, count, i, n;

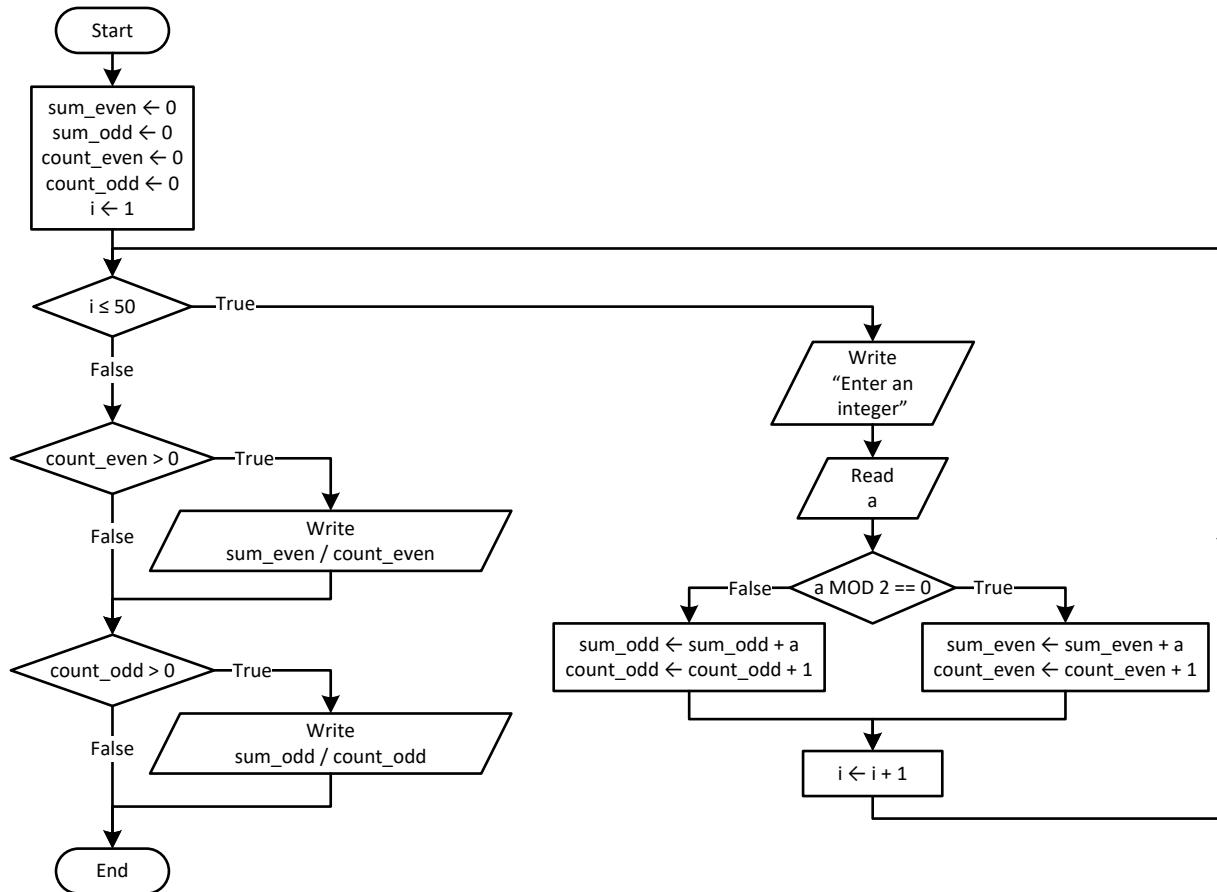
    cout << "Enter N: ";
    cin >> n;
    count = 0;
    for (i = 1; i <= n; i++) {
        cout << "Enter an integer: ";
        cin >> a;
        if (a % 2 == 0) {
            count++;
        }
    }
    if (count > 0) {
        cout << count << endl;
    }
    else {
        cout << "You entered no even integers" << endl;
    }
}
  
```

```

    return 0;
}

```

15. Solution



```

#include <iostream>
using namespace std;
int main() {
    int a, count_even, count_odd, i, sum_even, sum_odd;

    sum_even = 0;
    sum_odd = 0;
    count_even = 0;
    count_odd = 0;
    for (i = 1; i ≤ 50; i++) {
        cout << "Enter an integer: ";
        cin >> a;
        if (a % 2 == 0) {
            sum_even += a;
            count_even++;
        }
        else {
            sum_odd += a;
            count_odd++;
        }
    }
}

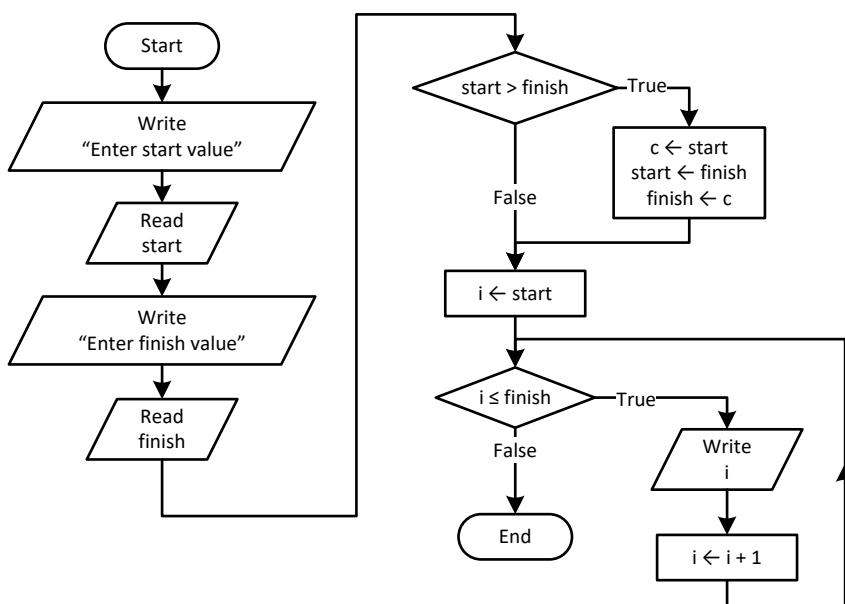
```

```

    }
    if (count_even > 0) {
        cout << sum_even / (double)count_even << endl;
    }
    if (count_odd > 0) {
        cout << sum_odd / (double)count_odd << endl;
    }
    return 0;
}
}

```

16. Solution



```

#include <iostream>
using namespace std;
int main() {
    int c, finish, i, start;

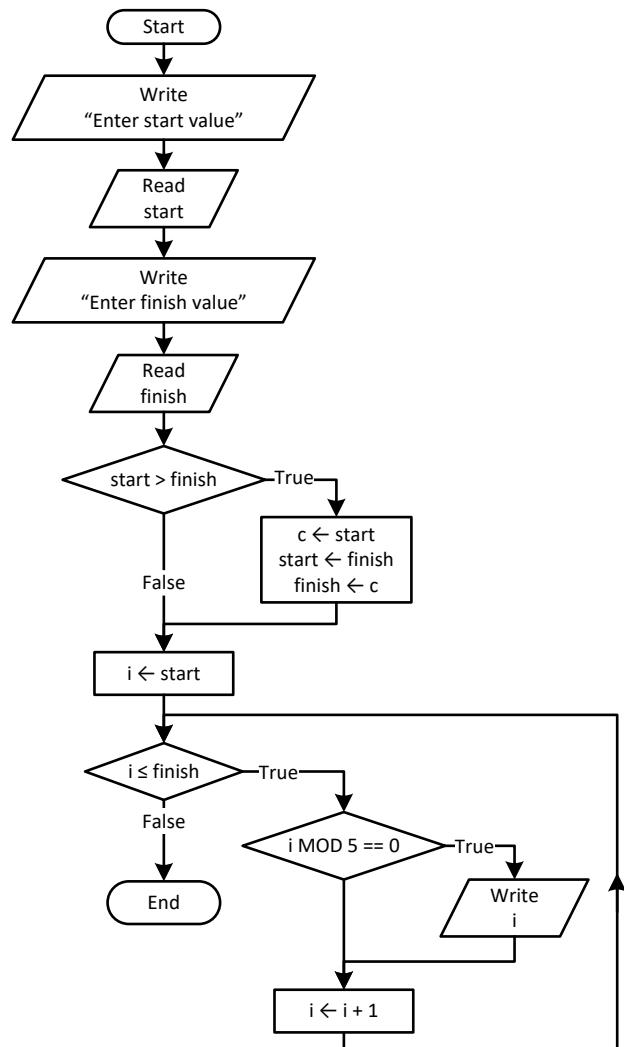
    cout << "Enter start value: ";
    cin >> start;
    cout << "Enter finish value: ";
    cin >> finish;

    if (start > finish) {
        c = start;
        start = finish;
        finish = c;
    }

    for (i = start; i <= finish; i++) {
        cout << i << endl;
    }
    return 0;
}

```

17. Solution



```

#include <iostream>
using namespace std;
int main() {
    int c, finish, i, start;

    cout << "Enter start value: ";
    cin >> start;
    cout << "Enter finish value: ";
    cin >> finish;

    if (start > finish) {
        c = start;
        start = finish;
        finish = c;
    }

    for (i = start; i <= finish; i++) {
        if (i % 5 == 0) {
            
```

```
    cout << i << endl;
}
}
return 0;
}
```

18. Solution

First Approach

```
#include <iostream>
using namespace std;
int main() {
    int exp, i;
    double p, b;

    cout << "Enter a value for base: ";
    cin >> b;
    cout << "Enter an integer for exponent: ";
    cin >> exp;

    p = 1;
    if (exp >= 0) {
        for (i = 1; i <= exp; i++) {
            p *= b;
        }
    } else {
        for (i = 1; i <= -exp; i++) {
            p *= 1 / b;
        }
    }
    cout << p << endl;
    return 0;
}
```

Second Approach

```
#include <iostream>
using namespace std;
int main() {
    int exp, i;
    double p, b;

    cout << "Enter a value for base: ";
    cin >> b;
    cout << "Enter an integer for exponent: ";
    cin >> exp;

    p = 1;
    for (i = 1; i <= abs(exp); i++) {
        p *= b;
    }
    if (exp < 0) {
        p = 1 / p;
    }
}
```

```
    }
    cout << p << endl;
    return 0;
}
```

19. Solution

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    int count, i, words;
    string msg;

    cout << "Enter a message: ";
    getline(cin, msg);

    count = 0;
    for (i = 0; i <= msg.length() - 1; i++) {
        if (msg[i] == ' ') {
            count++;
        }
    }
    words = count + 1;

    cout << "The message entered contains " << words << " words" << endl;
    return 0;
}
```

20. Solution

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    int characters, count, i, words;
    string msg;

    cout << "Enter a message: ";
    getline(cin, msg);

    characters = msg.length();
    count = 0;
    for (i = 0; i <= characters - 1; i++) {
        if (msg[i] == ' ') {
            count++;
        }
    }

    words = count + 1;
    cout << "The average number of letters in each word is ";
    cout << (characters - count) / (double)words << endl;
    return 0;
}
```

```
}
```

21. Solution

```
#include <iostream>
#include <string>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string message;
    char character;
    string consonants = "BCDFGHJKLMNPQRSTVWXYZ";
    int i, count;

    cout << "Enter an English message: ";
    getline(cin, message);
    message = to_upper_copy(message);

    count = 0;
    for (i = 0; i <= message.length() - 1; i++) {
        character = message[i];

        if (consonants.find(character) != -1) { //If character is found in consonants
            count++;
        }
    }
    cout << "Consonants: " << count << endl;
    return 0;
}
```

22. Solution

```
#include <iostream>
#include <string>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string message;
    char character;
    string vowels = "AEIOU";
    string consonants = "BCDFGHJKLMNPQRSTVWXYZ";
    string digits = "0123456789";
    int i, countv, countc, countd;

    cout << "Enter an English message: ";
    getline(cin, message);
    message = to_upper_copy(message);

    countv = countc = countd = 0;
    for (i = 0; i <= message.length() - 1; i++) {
        character = message[i];
```

```
if (vowels.find(character) != -1) { //If character is found in vowels
    countv++;
}
else if (consonants.find(character) != -1) { //If character is found in consonants
    countc++;
}
else if (digits.find(character) != -1) { //If character is found in digits
    countd++;
}
}

cout << "Vowels: " << countv << endl;
cout << "Consonants: " << countc << endl;
cout << "Digits: " << countd << endl;
return 0;
}
```

Chapter 27

27.3 Review Questions: True/False

- | | |
|----------|----------|
| 1. true | 6. false |
| 2. true | 7. true |
| 3. false | 8. true |
| 4. true | 9. true |
| 5. true | 10. true |

27.4 Review Questions: Multiple Choice

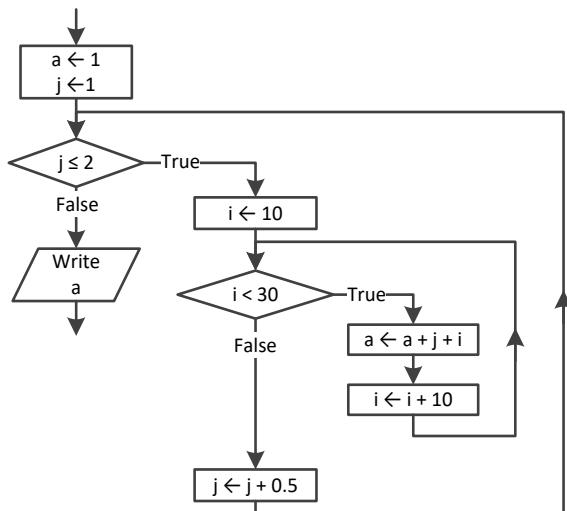
- | | |
|------|------|
| 1. b | 4. a |
| 2. a | 5. b |
| 3. c | |

27.5 Review Exercises

1. Solution

- i. 10
- ii. A value greater than or equal to 4.5 and less than 5.0
- iii. -7 (or -6)
- iv. 138 (or 139)

2. Solution



Step	Statement	a	i	j
1	a = 1	1	?	?
2	j = 1	1	?	1
3	j <= 2	true		
4	i = 10	1	10	1
5	i < 30	true		

6	a = a + j + i	12	10	1
7	i += 10	12	20	1
8	i < 30	true		
9	a = a * j + i	33	20	1
10	i += 10	33	30	1
11	i < 30	false		
12	j += 0.5	33	30	1.5
13	j <= 2	true		
14	i = 10	33	10	1.5
15	i < 30	true		
16	a = a + j + i	44.5	10	1.5
17	i += 10	44.5	20	1.5
18	i < 30	true		
19	a = a * j + i	66	20	1.5
20	i += 10	66	30	1.5
21	i < 30	false		
22	j += 0.5	66	30	2
23	j <= 2	true		
24	i = 10	66	10	2
25	i < 30	true		
26	a = a + j + i	78	10	2
27	i += 10	78	20	2
28	i < 30	true		
29	a = a * j + i	100	20	2
30	i += 10	100	30	2
31	i < 30	false		
32	j += 0.5	100	30	2.5
33	j <= 2	false		
34	cout << a << endl	It displays: 100		

3. Solution

Step	Statement	s	i	j
1	s = 0	0	?	?
2	i = 1	0	1	?
3	i <= 4	true		
4	j = 3	0	1	3
5	j >= i	true		

6	<code>s = s + i * j</code>	3	1	3
7	<code>j--</code>	3	1	2
8	<code>j >= i</code>	true		
9	<code>s = s + i * j</code>	5	1	2
10	<code>j--</code>	5	1	1
11	<code>j >= i</code>	true		
12	<code>s = s + i * j</code>	6	1	1
13	<code>j--</code>	6	1	0
14	<code>j >= i</code>	false		
15	<code>i++</code>	6	2	0
16	<code>i <= 4</code>	true		
17	<code>j = 3</code>	6	2	3
18	<code>j >= i</code>	true		
19	<code>s = s + i * j</code>	12	2	3
20	<code>j--</code>	12	2	2
21	<code>j >= i</code>	true		
22	<code>s = s + i * j</code>	16	2	2
23	<code>j--</code>	16	2	1
24	<code>j >= i</code>	false		
25	<code>i++</code>	16	3	1
26	<code>i <= 4</code>	true		
27	<code>j = 3</code>	16	3	3
28	<code>j >= i</code>	true		
29	<code>s = s + i * j</code>	25	3	3
30	<code>j--</code>	25	3	2
31	<code>j >= i</code>	false		
32	<code>i++</code>	25	4	2
33	<code>i <= 4</code>	true		
34	<code>j = 3</code>	25	4	3
35	<code>j >= i</code>	false		
36	<code>i++</code>	25	5	3
37	<code>i <= 4</code>	false		
38	<code>cout << s << endl</code>	It displays: 25		

The statement `s = s + i * j` is executed 6 times

4. Solution

For input value of "NO"

Step	Statement	s	y	i	ans
1	s = 1	1	?	?	?
2	y = 25	1	25	?	?
3	i = 1	1	25	1	?
4	i <= 3		true		
5	s = s + y	26	25	1	?
6	y -= 5	26	20	1	?
7	i++	26	20	2	?
8	i <= 3		true		
9	s = s + y	46	20	2	?
10	y -= 5	46	15	2	?
11	i++	46	15	3	?
12	i <= 3		true		
13	s = s + y	61	15	3	?
14	y -= 5	61	10	3	?
15	i++	61	10	4	?
16	i <= 3		false		
17	cin >> ans	61	10	4	"NO"
18	while (ans == "YES")		false		
19	cout << s << endl	It displays: 61			

For input values of "YES", "NO"

Step	Statement	s	y	i	ans
1	s = 1	1	?	?	?
2	y = 25	1	25	?	?
3	i = 1	1	25	1	?
4	i <= 3		true		
5	s = s + y	26	25	1	?
6	y -= 5	26	20	1	?
7	i++	26	20	2	?
8	i <= 3		true		
9	s = s + y	46	20	2	?
10	y -= 5	46	15	2	?
11	i++	46	15	3	?
12	i <= 3		true		
13	s = s + y	61	15	3	?
14	y -= 5	61	10	3	?
15	i++	61	10	4	?

16	i <= 3	false			
17	cin >> ans	61	10	4	"YES"
18	while (ans == "YES")	true			
19	i = 1	61	10	1	"YES"
20	i <= 3	true			
21	s = s + y	71	10	1	"YES"
22	y -= 5	71	5	1	"YES"
23	i++	71	5	2	"YES"
24	i <= 3	true			
25	s = s + y	76	5	2	"YES"
26	y -= 5	76	0	2	"YES"
27	i++	76	0	3	"YES"
28	i <= 3	true			
29	s = s + y	76	0	3	"YES"
30	y -= 5	76	-5	3	"YES"
31	i++	76	-5	4	"YES"
32	i <= 3	false			
33	cin >> ans	76	-5	4	"NO"
34	while (ans == "YES")	false			
35	cout << s << endl	It displays: 76			

For input values of "YES", "YES", "NO"

Step	Statement	s	y	i	ans
1	s = 1	1	?	?	?
2	y = 25	1	25	?	?
3	i = 1	1	25	1	?
4	i <= 3	true			
5	s = s + y	26	25	1	?
6	y -= 5	26	20	1	?
7	i++	26	20	2	?
8	i <= 3	true			
9	s = s + y	46	20	2	?
10	y -= 5	46	15	2	?
11	i++	46	15	3	?
12	i <= 3	true			
13	s = s + y	61	15	3	?
14	y -= 5	61	10	3	?
15	i++	61	10	4	?

16	i <= 3	false			
17	cin >> ans	61	10	4	"YES"
18	while (ans == "YES")	true			
19	i = 1	61	10	1	"YES"
20	i <= 3	true			
21	s = s + y	71	10	1	"YES"
22	y -= 5	71	5	1	"YES"
23	i++	71	5	2	"YES"
24	i <= 3	true			
25	s = s + y	76	5	2	"YES"
26	y -= 5	76	0	2	"YES"
27	i++	76	0	3	"YES"
28	i <= 3	true			
29	s = s + y	76	0	3	"YES"
30	y -= 5	76	-5	3	"YES"
31	i++	76	-5	4	"YES"
32	i <= 3	false			
33	cin >> ans	76	-5	4	"YES"
34	while (ans == "YES")	true			
35	i = 1	76	-5	1	"YES"
36	i <= 3	true			
37	s = s + y	71	-5	1	"YES"
38	y -= 5	71	-10	1	"YES"
39	i++	71	-10	2	"YES"
40	i <= 3	true			
41	s = s + y	61	-10	2	"YES"
42	y -= 5	61	-15	2	"YES"
43	i++	61	-15	3	"YES"
44	i <= 3	true			
45	s = s + y	46	-15	3	"YES"
46	y -= 5	46	-20	3	"YES"
47	i++	46	-20	4	"YES"
48	i <= 3	false			
49	cin >> ans	46	-20	4	"NO"
50	while (ans == "YES")	false			
51	cout << s << endl	It displays: 46			

5. Solution

```
#include <iostream>
using namespace std;
int main() {
    int hour, minutes;

    for (hour = 0; hour <= 23; hour++) {
        for (minutes = 0; minutes <= 59; minutes++) {
            cout << hour << "\t" << minutes << endl;
        }
    }
    return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, j;

    for (i = 5; i >= 1; i--) {
        for (j = 1; j <= i; j++) {
            cout << i << " ";
        }
        cout << endl;
    }
    return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, j;

    for (i = 0; i <= 5; i++) {
        for (j = 0; j <= i; j++) {
            cout << j << " ";
        }
        cout << endl;
    }
    return 0;
}
```

8. Solution

```
#include <iostream>
using namespace std;
int main() {
```

```
int i, j;

for (i = 1; i <= 4; i++) {
    for (j = 1; j <= 10; j++) {
        cout << "* ";
    }
    cout << endl;
}
return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, j, y;

    cout << "Enter an integer between 3 and 20: ";
    cin >> y;

    for (i = 1; i <= y; i++) {
        for (j = 1; j <= y; j++) {
            cout << "* ";
        }
        cout << endl;
    }
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, j, y;

    cout << "Enter an integer between 3 and 20: ";
    cin >> y;

    for (j = 1; j <= y; j++) {
        cout << "* ";
    }
    cout << endl;

    for (i = 1; i <= y - 2; i++) {
        cout << "* ";
        for (j = 1; j <= y - 2; j++) {
            cout << "  ";
        }
        cout << "* " << endl;
    }
}
```

```
    for (j = 1; j <= y; j++) {  
        cout << "* ";  
    }  
    return 0;  
}
```

11. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    int i, j;  
  
    for (i = 1; i <= 5; i++) {  
        for (j = 1; j <= i; j++) {  
            cout << "* ";  
        }  
        cout << endl;  
    }  
  
    for (i = 4; i >= 1; i--) {  
        for (j = 1; j <= i; j++) {  
            cout << "* ";  
        }  
        cout << endl;  
    }  
    return 0;  
}
```

Chapter 28

28.8 Review Questions: True/False

- | | |
|----------|-----------|
| 1. false | 8. false |
| 2. false | 9. true |
| 3. false | 10. true |
| 4. true | 11. false |
| 5. true | 12. false |
| 6. false | 13. false |
| 7. false | 14. true |

28.9 Review Questions: Multiple Choice

- | | |
|------|------|
| 1. c | 5. a |
| 2. d | 6. c |
| 3. b | 7. c |
| 4. a | |

28.10 Review Exercises

1. Solution

```
count_names = 0;
count_not_johns = 0;
name = "";
cout << "Enter a name: ";
cin >> name;
while (name != "STOP") {
    cout << "Enter a name: ";
    cin >> name;
    count_names++;
    if (name != "John") {
        count_not_johns++;
    }
    cout << "Enter a name: ";
    cin >> name;
}
cout << count_names << " names entered" << endl;
cout << "Names other than John entered " << count_not_johns << " times" << endl;
```

2. Solution

First Approach

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    string text, character;
    bool found;
    int i;

    cout << "Enter a text: ";
```

```
getline(cin, text);

found = false;
for (i = 0; i <= text.length() - 1; i++) {
    character = text[i];
    if (character == " ") {
        found = true;
        break;
    }
}

if (!found) {
    cout << "One Single Word" << endl;
}
else {
    cout << "Complete Sentence" << endl;
}
return 0;
}
```

Second Approach

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    string text;

    cout << "Enter a text: ";
    getline(cin, text);

    if (text.find(" ") == -1) {
        cout << "One Single Word" << endl;
    }
    else {
        cout << "Complete Sentence" << endl;
    }
    return 0;
}
```

3. Solution

First Approach

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    string sentence, character;
    bool found;
    int i;
    string digits = "0123456789";

    cout << "Enter a text: ";
    getline(cin, sentence);
```

```
found = false;
for (i = 0; i <= sentence.length() - 1; i++) {
    character = sentence[i];
    if (digits.find(character) != -1) {
        found = true;
        break;
    }
}

if (found) {
    cout << "The sentence contains a number" << endl;
}
return 0;
}
```

Second Approach

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    string sentence;
    bool found;
    int i;
    string digit;

    cout << "Enter a text: ";
    getline(cin, sentence);

    found = false;
    for (i = 0; i <= 9; i++) {
        digit = to_string(i);
        if (sentence.find(digit) != -1) {
            found = true;
            break;
        }
    }

    if (found) {
        cout << "The sentence contains a number" << endl;
    }
    return 0;
}
```

4. Solution

```
cout << "Printing all integers from 1 to 100" << endl;
i = 1;
while (i < 101) {
    cout << i << endl;
    i++;
}
```

5. Solution

```
cout << "Printing odd integers from 1 to 99" << endl;
i = 1;
while (i < 100) {
    cout << i << endl;
    i += 2;
}
```

6. Solution

```
s = 0;
for (i = 1; i <= 100; i++) {
    cin >> number;
    s = s + number;
}
average = s / 100.0;
cout << average << endl;
```

7. Solution

```
int i, denom;
double s;

s = 0;

denom = 1;
for (i = 1; i <= 100; i++) {
    denom *= i;
}

for (i = 1; i <= 100; i++) {
    s += i / (double)denom;
}
cout << s << endl;
```

8. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, j;

    for (i = 1; i <= 4; i++) {
        for (j = 1; j <= 4; j++) {
            cout << i << " x " << j << " = " << i * j << endl;
        }
    }
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, j;

    cout << "\t|\t";
    for (i = 1; i <= 12; i++) {
        cout << i << "\t";
    }
    cout << endl;

    for (i = 1; i <= 12; i++) {
        cout << "-----";
    }
    cout << endl;

    for (i = 1; i <= 12; i++) {
        cout << i << "\t|\t";
        for (j = 1; j <= 12; j++) {
            cout << i * j << "\t";
        }
        cout << endl;
    }
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, j, n;

    cout << "Enter an integer: ";
    cin >> n;

    cout << "\t|\t";
    for (i = 1; i <= n; i++) {
        cout << i << "\t";
    }
    cout << endl;

    for (i = 1; i <= n; i++) {
        cout << "-----";
    }
    cout << endl;

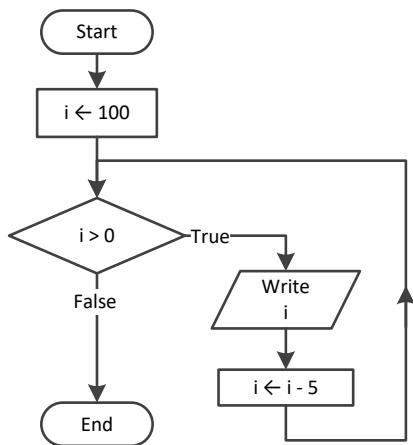
    for (i = 1; i <= n; i++) {
        cout << i << "\t|\t";
        for (j = 1; j <= n; j++) {
            cout << i * j << "\t";
        }
        cout << endl;
    }
}
```

```
    }
    cout << endl;
}
return 0;
}
```

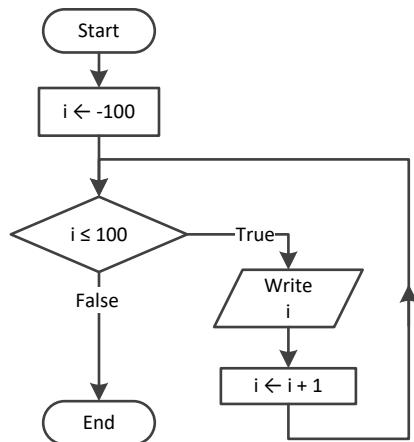
Chapter 29

29.4 Review Exercises

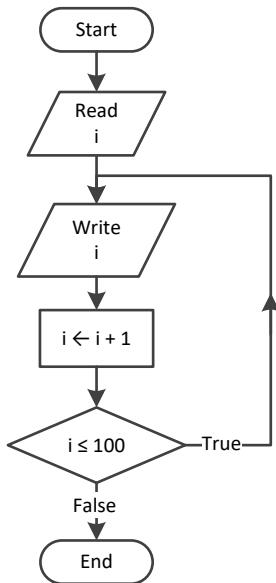
1. Solution



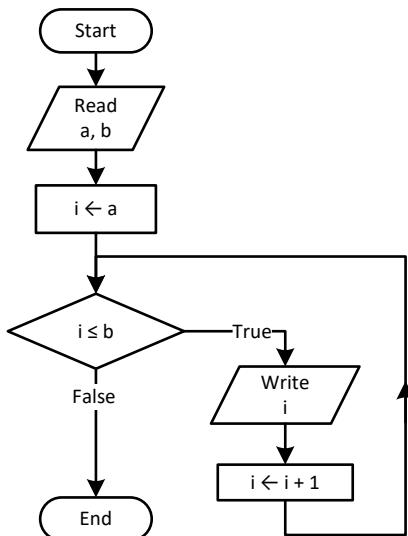
2. Solution



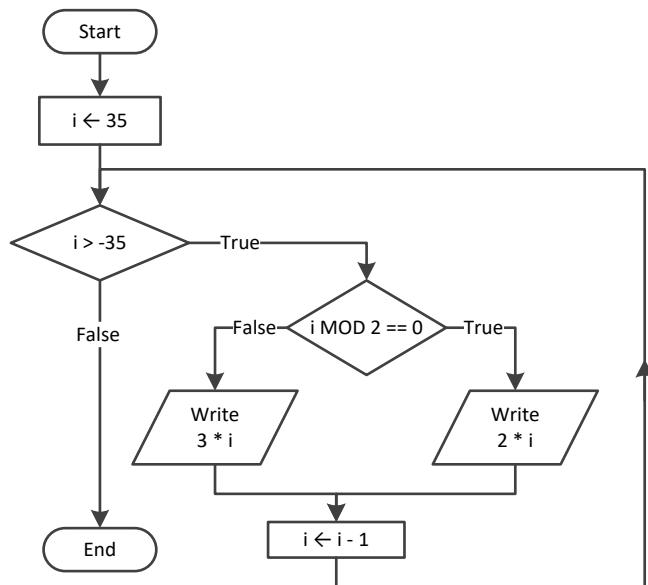
3. Solution



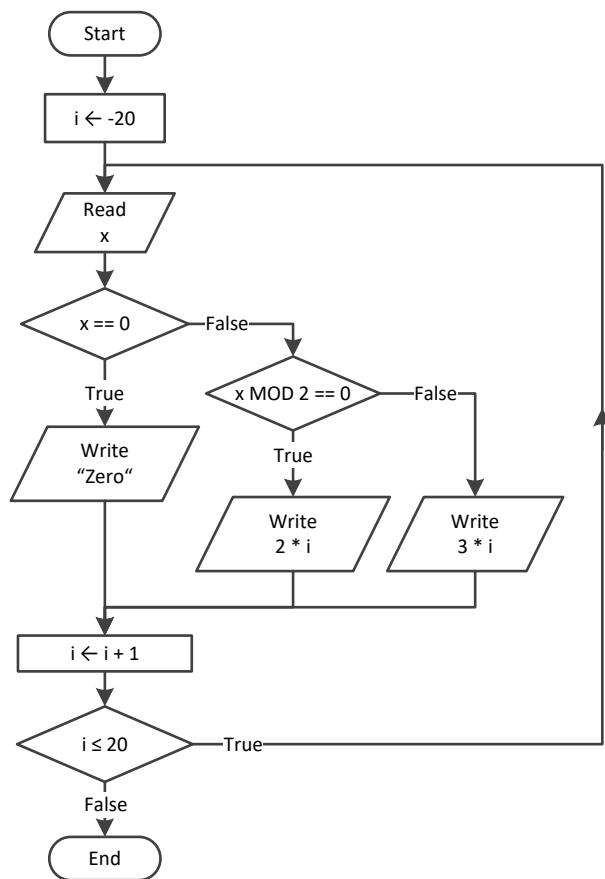
4. Solution



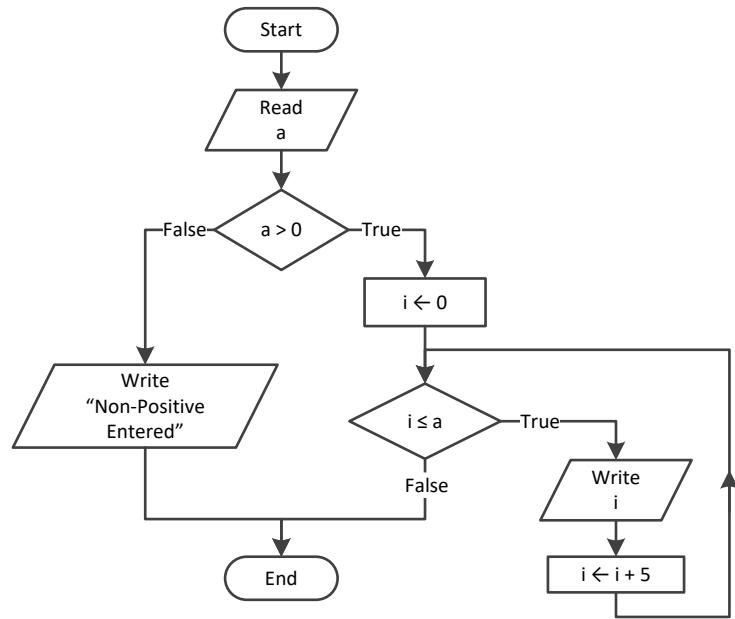
5. Solution



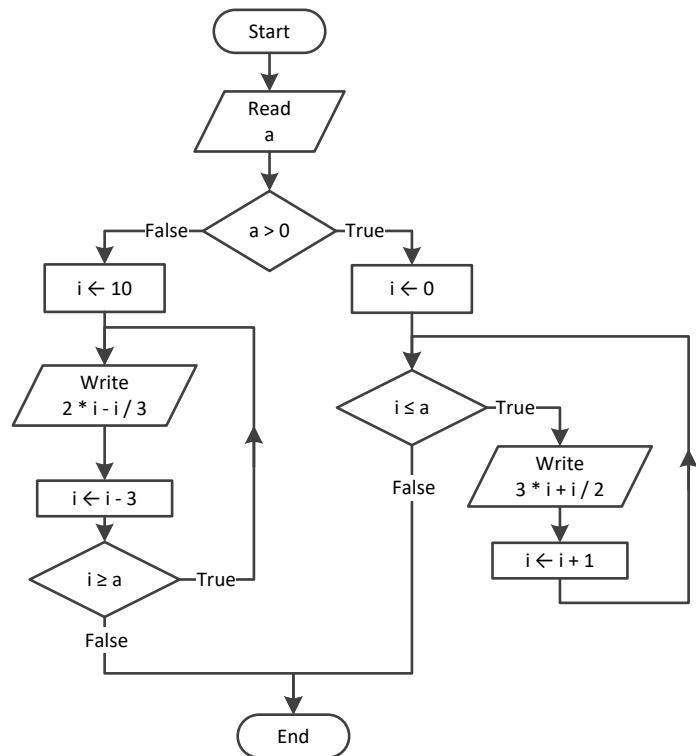
6. Solution

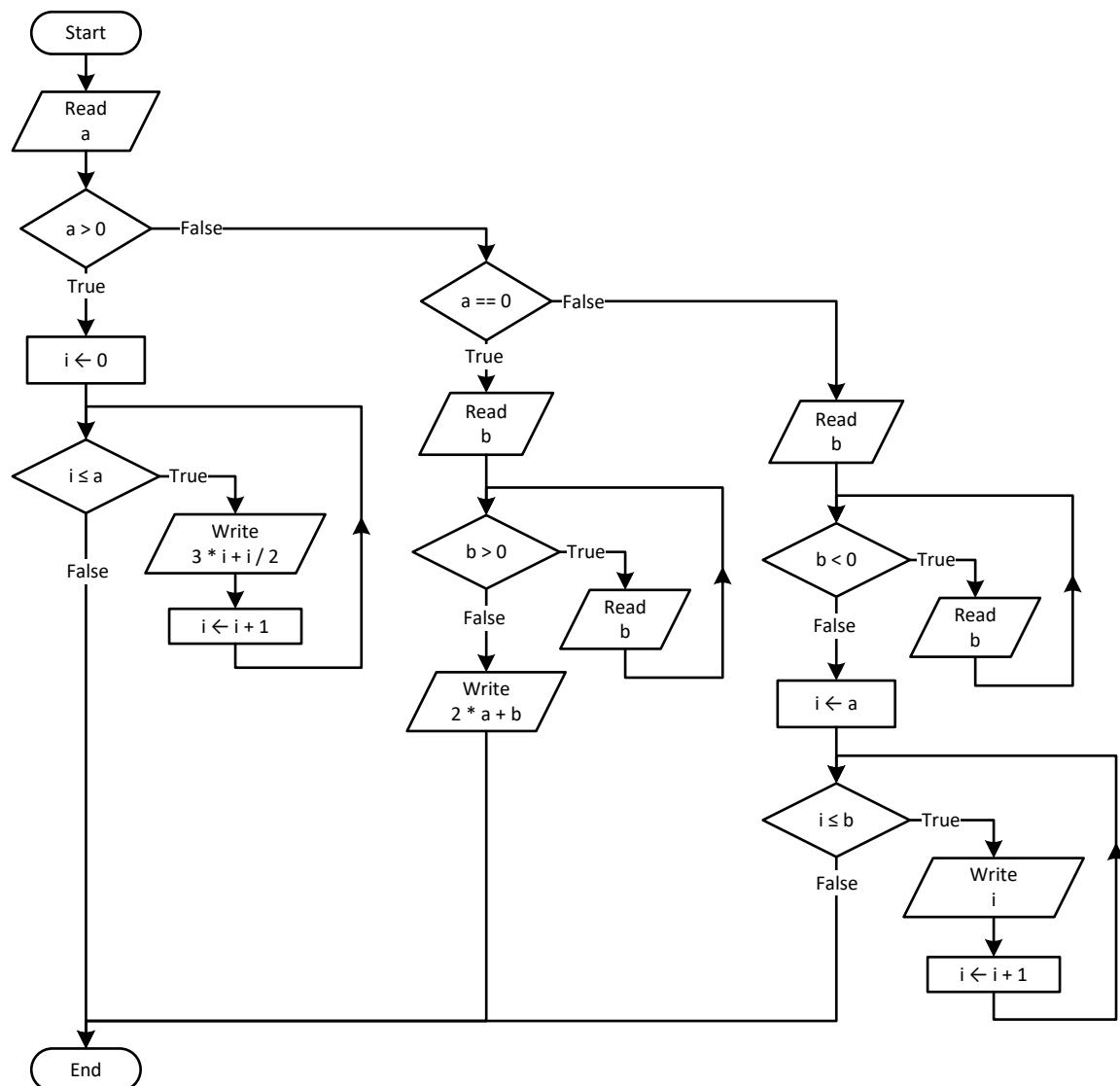


7. Solution

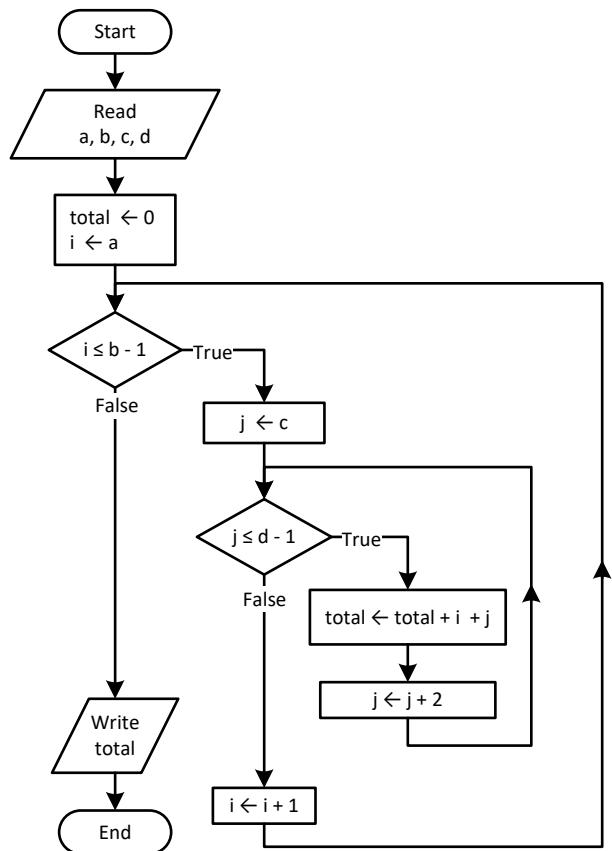


8. Solution

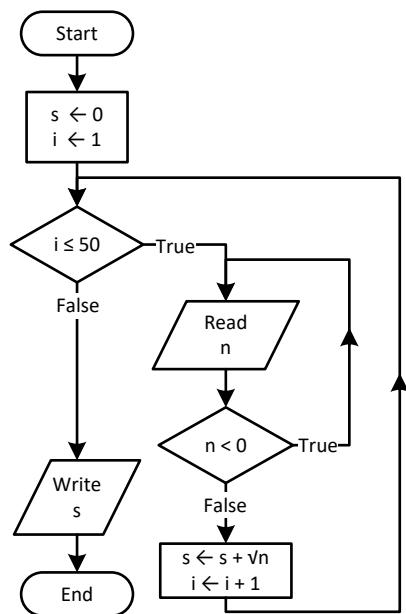


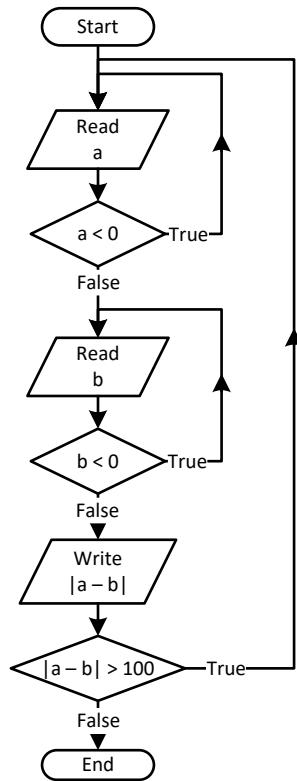
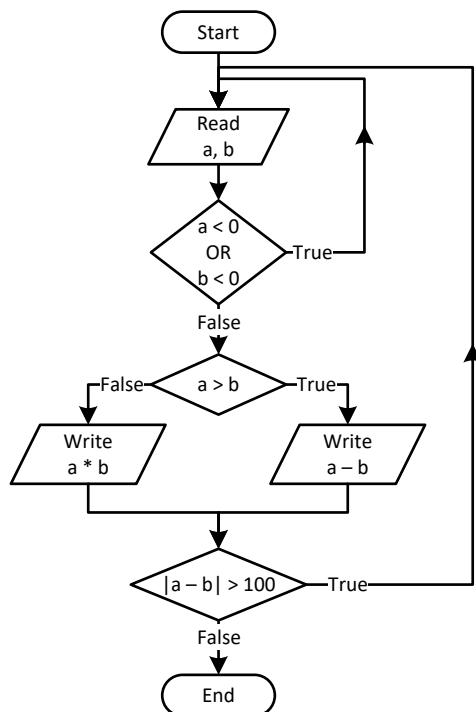
9. Solution

10. Solution



11. Solution



12. Solution**13. Solution**

14. Solution

```
i = 1;
do {
    cout << i << endl;
    i += 5;
} while (i <= 500);
cout << "The End" << endl;
```

15. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, i;

    i = 0;
    cin >> a;
    do {
        if (i % 2 != 0) {
            cout << i << endl;
        }
        i += 5;
    } while (i < a);
    return 0;
}
```

16. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, b, i;

    cin >> a;
    while (a != -1) {
        do {
            cin >> b;
        } while (b <= a);
        for (i = a; i <= b; i++) {
            cout << i << endl;
        }
        cin >> a;
    }
    return 0;
}
```

17. Solution

```
#include <iostream>
using namespace std;
int main() {
```

```
int i;
double P, S, a;

i = 1;
S = 0;
P = 1;
a = 0;

while (true) {
    if (i < 45) {
        S += a;
    }
    else {
        P *= a;
    }
    i++;
    if (i >= 90) break;
    cin >> a;
}

cout << S << " " << P << endl;
return 0;
}
```

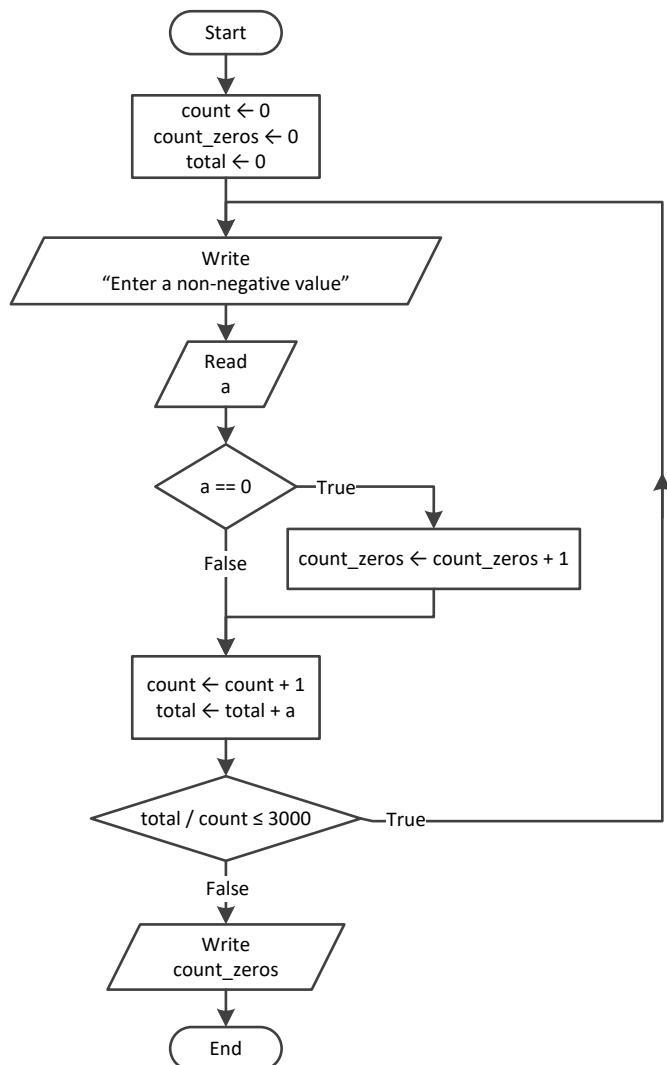
Chapter 30

30.7 Review Questions: True/False

- | | |
|----------|----------|
| 1. true | 6. false |
| 2. false | 7. false |
| 3. false | 8. false |
| 4. true | 9. true |
| 5. false | |

30.8 Review Exercises

1. Solution



```
#include <iostream>
using namespace std;
int main() {
    int count, count_zeros;
    double a, total;
```

```

count = 0;
count_zeros = 0;
total = 0;
do {
    cout << "Enter a non-negative value: ";
    cin >> a;
    if (a == 0) {
        count_zeros++;
    }
    count++;
    total += a;
} while (total / count <= 3000);
cout << count_zeros << endl;
return 0;
}

```

2. Solution

First Approach

```

#include <iostream>
using namespace std;
int main() {
    int a, d1, d2, d3, d4, i, r;

    cout << "Enter an integer between 1 and 20: ";
    cin >> a;
    for (i = 1000; i <= 9999; i++) {
        d4 = i % 10;
        r = (int)(i / 10);
        d3 = r % 10;
        r = (int)(r / 10);
        d2 = r % 10;
        d1 = (int)(r / 10);
        if (d1 + d2 + d3 + d4 < a) {
            cout << i << endl;
        }
    }
    return 0;
}

```

Second Approach

```

#include <iostream>
using namespace std;
int main() {
    int a, d1, d2, d3, d4;

    cout << "Enter an integer between 1 and 20: ";
    cin >> a;
    for (d1 = 1; d1 <= 9; d1++) {
        for (d2 = 0; d2 <= 9; d2++) {
            for (d3 = 0; d3 <= 9; d3++) {
                for (d4 = 0; d4 <= 9; d4++) {

```

```

        if (d1 + d2 + d3 + d4 < a) {
            cout << d1 * 1000 + d2 * 100 + d3 * 10 + d4 << endl;
        }
    }
}
}

return 0;
}

```

3. Solution

First Approach

```

#include <iostream>
using namespace std;
int main() {
    int d1, d2, d3, d4, i, r;

    for (i = 1000; i <= 9999; i++) {
        d4 = i % 10;
        r = (int)(i / 10);
        d3 = r % 10;
        r = (int)(r / 10);
        d2 = r % 10;
        d1 = (int)(r / 10);
        if (d1 > d2 && d2 == d3 && d3 < d4) {
            cout << i << endl;
        }
    }
    return 0;
}

```

Second Approach

```

#include <iostream>
using namespace std;
int main() {
    int d1, d2, d3, d4;

    for (d1 = 1; d1 <= 9; d1++) {
        for (d2 = 0; d2 <= 9; d2++) {
            for (d3 = 0; d3 <= 9; d3++) {
                for (d4 = 0; d4 <= 9; d4++) {
                    if (d1 > d2 && d2 == d3 && d3 < d4) {
                        cout << d1 * 1000 + d2 * 100 + d3 * 10 + d4 << endl;
                    }
                }
            }
        }
    }
    return 0;
}

```

4. Solution

First approach

```
#include <iostream>
using namespace std;
int main() {
    int x, count;

    cout << "Enter a number: ";
    cin >> x;

    count = 0;

    while (x != 0) {
        count++;
        x = (int)(x / 10);
    }

    cout << count << endl;
    return 0;
}
```

Second approach

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    int x, count;

    cout << "Enter a number: ";
    cin >> x;

    //Convert the absolute value of x to string and get its length
    count = to_string(abs(x)).length();

    cout << count << endl;
    return 0;
}
```

5. Solution

```
cin >> x;
while (x != 1 && x != 0) {
    cout << "Error" << endl;
    cin >> x;
}
```

6. Solution

```
do {
    cin >> gender;
    gender = to_upper_copy(gender);
} while (gender != "M" && gender != "F");
```

7. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int count, x;
    double y;

    cout << "Enter a non-negative number: ";
    cin >> x;
    count = 0;
    while (x < 0) {
        count++;
        if (count == 2) break;

        cout << "Error: Invalid number!" << endl;
        cout << "Enter a non-negative number: ";
        cin >> x;
    }

    if (count < 2) {
        y = sqrt(x);
        cout << y << endl;
    }
    else {
        cout << "Dude, you are dumb!" << endl;
    }
    return 0;
}
```

8. Solution

```
#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string answer;
    double area, r;

    do {
        cout << "Enter the length of a radius: ";
        cin >> r;
        while (r <= 0) {
            cout << "Invalid radius. Enter the length of a radius: ";
            cin >> r;
        }
        area = M_PI * pow(r, 2);
    }
```

```

    cout << "The area is: " << area << endl;

    cout << "Would you like to repeat? ";
    cin >> answer;
} while (to_upper_copy(answer) == "YES");

return 0;
}

```

9. Solution

```

#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int x, y;

    for (x = -100; x <= 100; x++) {
        for (y = -100; y <= 100; y++) {
            if (5 * x + 3 * pow(y, 2) == 0) {
                cout << x << ", " << y << endl;
            }
        }
    }
    return 0;
}

```

10. Solution

```

#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int x, y, z;

    for (x = -10; x <= 10; x++) {
        for (y = -10; y <= 10; y++) {
            for (z = -10; z <= 10; z++) {
                if ((x + y) / 2.0 + 3.0 * pow(z, 2) / (x + 3 * y + 45) == x / 3.0) {
                    cout << x << ", " << y << ", " << z << endl;
                }
            }
        }
    }
    return 0;
}

```

11. Solution

```

#include <iostream>
using namespace std;
int main() {
    int m1, m2, m3, s;

```

```
cin >> m1 >> m2 >> m3;

s = 0;
while (m2 != 0) {
    if (m2 % 2 != 0) {
        s += m1;
    }
    m1 *= 2;
    m2 = (int)(m2 / 2);
}

m1 = s;
m2 = m3;

s = 0;
while (m2 != 0) {
    if (m2 % 2 != 0) {
        s += m1;
    }
    m1 *= 2;
    m2 = (int)(m2 / 2);
}

cout << s << endl;
return 0;
}
```

12. Solution

```
#include <iostream>
using namespace std;
int main() {
    int x, number_of_divisors, i;

    cin >> x;
    while (x <= 0) {
        cout << "Error! You must enter a positive integer" << endl;
        cin >> x;
    }

    number_of_divisors = 2;
    for (i = 2; i <= (int)(x / 2); i++) {
        if (x % i == 0) {
            number_of_divisors++;
        }
    }
    cout << number_of_divisors << endl;
    return 0;
}
```

13. Solution

```
#include <iostream>
using namespace std;
int main() {
    int x, number_of_divisors, i;

    cout << "Enter an integer greater than 1: ";
    cin >> x;
    while (x <= 1) {
        cout << "Error!" << endl;
        cin >> x;
    }

    number_of_divisors = 2;
    for (i = 2; i <= (int)(x / 2); i++) {
        if (x % i == 0) {
            number_of_divisors++;
            break;
        }
    }

    if (number_of_divisors == 2) {
        cout << "Number " << x << " is prime" << endl;
    }
    return 0;
}
```

14. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, b, c, i, number_of_divisors, x;

    cout << "Enter an integer greater than 1: ";
    cin >> a;
    while (a < 2) {
        cout << "Wrong number. Please enter an integer greater than 1: ";
        cin >> a;
    }

    cout << "Enter a second integer greater than 1: ";
    cin >> b;
    while (b < 2) {
        cout << "Wrong number. Please enter a second integer greater than 1: ";
        cin >> b;
    }

    if (a > b) {
        c = a;
        a = b;
        b = c;
    }

    for (i = 2; i <= (int)(a / 2); i++) {
        if (a % i == 0) {
            number_of_divisors++;
            break;
        }
    }

    if (number_of_divisors == 1) {
        cout << a << " and " << b << " are coprime" << endl;
    } else {
        cout << a << " and " << b << " are not coprime" << endl;
    }
}
```

```
b = c;
}

for (x = a; x <= b; x++) {
    number_of_divisors = 2;
    i = 2;
    while (i <= (int) (x / 2) && number_of_divisors == 2) {
        if (x % i == 0) {
            number_of_divisors++;
        }
        i++;
    }
    if (number_of_divisors == 2) {
        cout << "Number " << x << " is prime" << endl;
    }
}
return 0;
}
```

15. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, b, c, d1, d2, d3, d4, r, x;

    cout << "Enter a positive four-digit integer: ";
    cin >> a;
    while (a < 1000 || a > 9999) {
        cout << "Wrong number. Please enter a positive four-digit integer: ";
        cin >> a;
    }

    cout << "Enter a second positive four-digit integer: ";
    cin >> b;
    while (b < 1000 || b > 9999) {
        cout << "Wrong number. Please enter a second positive four-digit integer: ";
        cin >> b;
    }

    if (a > b) {
        c = a;
        a = b;
        b = c;
    }

    for (x = a; x <= b; x++) {
        d4 = x % 10;
        r = (int) (x / 10);
        d3 = r % 10;
        r = (int) (r / 10);
        d2 = r % 10;
```

```
d1 = (int)(r / 10);

if (d1 == d4 && d2 == d3) {
    cout << x << endl;
}
}

return 0;
}
```

16. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int i;

    for (i = 0; i <= 30; i++) {
        cout << pow(2, i) << endl;
    }
    return 0;
}
```

17. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, offset;

    offset = 10;
    i = 1;
    while (i <= 401) {
        cout << i << endl;
        i += offset;
        offset += 2;
    }
    return 0;
}
```

18. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i;

    for (i = 1; i <= 100; i++) {
        cout << -i << "\n" << i << endl;
    }
    return 0;
}
```

19. Solution

First Approach

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int i, offset, value;

    value = 0;
    for (i = 1; i <= 8; i++) {
        offset = pow(10, i - 1);
        value += offset;
        cout << value << endl;
    }
    return 0;
}
```

Second Approach

```
#include <iostream>
using namespace std;
int main() {
    int i;
    string value;

    value = "1";
    for (i = 1; i <= 8; i++) {
        cout << value << endl;
        value += "1";
    }
    return 0;
}
```

20. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, fib, fib_prev, fib_prev_prev, i;

    cin >> a;

    fib_prev_prev = 0;
    fib_prev = 1;
    fib = 1;
    for (i = 1; i <= a; i++) {
        cout << fib << endl;
        fib = fib_prev + fib_prev_prev;
        fib_prev_prev = fib_prev;
        fib_prev = fib;
    }
    return 0;
}
```

```
}
```

21. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, fib, fib_prev, fib_prev_prev;

    cin >> a;

    fib_prev_prev = 0;
    fib_prev = 1;
    fib = 1;
    while (fib < a) {
        cout << fib << endl;
        fib = fib_prev + fib_prev_prev;
        fib_prev_prev = fib_prev;
        fib_prev = fib;
    }
    return 0;
}
```

22. Solution

```
#include <iostream>
using namespace std;
int main() {
    int denominator, i, n, nominator;
    double y;

    cout << "Enter a positive integer: ";
    cin >> n;
    while (n <= 0) {
        cout << "Wrong number. Please enter a positive integer: ";
        cin >> n;
    }

    nominator = 0;
    for (i = 2; i <= 2 * n; i += 2) {
        nominator += i;
    }

    denominator = 1;
    for (i = 1; i <= n; i++) {
        denominator *= i;
    }

    y = nominator / (double)denominator;
    cout << y << endl;
    return 0;
}
```

23. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, n, nominator, sign;
    double y;

    cout << "Enter a positive integer: ";
    cin >> n;
    while (n <= 0) {
        cout << "Wrong number. Please enter a positive integer: ";
        cin >> n;
    }

    nominator = 0;
    sign = 1;
    for (i = 1; i <= 2 * n + 1; i += 2) {
        nominator += sign * i;
        sign = -sign;
    }

    y = nominator / (double)n;
    cout << y << endl;
    return 0;
}
```

24. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, n, sign;
    double y;

    cout << "Enter a positive integer: ";
    cin >> n;
    while (n <= 0) {
        cout << "Wrong number. Please enter a positive integer: ";
        cin >> n;
    }

    y = 0.5; //This is equal to the first two terms: 1 - 1 / 2
    sign = 1;
    for (i = 3; i <= n; i += 2) {
        y += sign / (double)i;
        sign = -sign;
    }

    cout << y << endl;
    return 0;
}
```

25. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int i, n;
    double y;

    cout << "Enter a positive integer: ";
    cin >> n;
    while (n <= 0) {
        cout << "Wrong number. Please enter a positive integer: ";
        cin >> n;
    }

    y = 0;
    for (i = 1; i <= n; i++) {
        y += 1 / pow(i, n - i + 1);
    }

    cout << y << endl;
    return 0;
}
```

26. Solution

```
#include <iostream>
using namespace std;
int main() {
    int factorial, i, n;

    cout << "Enter a non-negative integer: ";
    cin >> n;

    factorial = 1;
    for (i = 1; i <= n; i++) {
        factorial *= i;
    }

    cout << factorial << endl;
    return 0;
}
```



Please note that this code operates properly for all non-negative integers, including zero.

27. Solution

First Approach

```
#include <iostream>
#include <cmath>
using namespace std;
const double ACCURACY = 0.00001;
```

```
int main() {
    int i, j;
    double factorial, exponential_previous, exponential, x;

    cin >> x;

    exponential = 0;
    i = 0;
    do {
        exponential_previous = exponential;

        factorial = 1;
        for (j = 1; j <= i; j++) {
            factorial *= j;
        }

        exponential += pow(x, i) / factorial;

        i++;
    } while (abs(exponential - exponential_previous) > ACCURACY);

    cout << "e(" << x << ") ~=" << exponential << endl;
    return 0;
}
```

Second Approach

```
#include <iostream>
#include <cmath>
using namespace std;
const double ACCURACY = 0.00001;

int main() {
    int i;
    double factorial, exponential_previous, exponential, x;

    cin >> x;

    exponential = 1;
    i = 1;
    factorial = 1;
    do {
        exponential_previous = exponential;

        factorial *= i;

        exponential += pow(x, i) / factorial;

        i++;
    } while (abs(exponential - exponential_previous) > ACCURACY);

    cout << "e(" << x << ") ~=" << exponential << endl;
    return 0;
}
```

28. Solution

First Approach

```
#include <iostream>
#include <cmath>
using namespace std;
const double ACCURACY = 0.00001;

int main() {
    int i, j, sign;
    double factorial;
    double sinus, sinus_previous, x;

    cin >> x;

    sign = 1;
    sinus = 0;
    i = 1;
    do {
        sinus_previous = sinus;

        factorial = 1;
        for (j = 1; j <= i; j++) {
            factorial *= j;
        }

        sinus += sign * pow(x, i) / factorial;

        sign = -sign;
        i += 2;
    } while (abs(sinus - sinus_previous) > ACCURACY);

    cout << "sin(" << x << ") ~=" << sinus << endl;
    return 0;
}
```

Second Approach

```
#include <iostream>
#include <cmath>
using namespace std;
const double ACCURACY = 0.00001;

int main() {
    int i, sign;
    double factorial;
    double sinus, sinus_previous, x;

    cin >> x;

    sign = -1;
    sinus = x;
    i = 3;
    factorial = 1;
    do {
```

```
sinus_previous = sinus;
factorial *= i * (i - 1);
sinus += sign * pow(x, i) / factorial;
sign = -sign;
i += 2;
} while (abs(sinus - sinus_previous) > ACCURACY);

cout << "sin(" << x << ") ~= " << sinus << endl;
return 0;
}
```

29. Solution

First Approach

```
#include <iostream>
#include <cmath>
using namespace std;
const double ACCURACY = 0.00001;

int main() {
    int i, j, sign;
    double factorial;
    double cosinus, cosinus_previous, x;

    cin >> x;

    sign = 1;
    cosinus = 0;
    i = 0;
    do {
        cosinus_previous = cosinus;

        factorial = 1;
        for (j = 1; j <= i; j++) {
            factorial *= j;
        }

        cosinus += sign * pow(x, i) / factorial;

        sign = -sign;
        i += 2;
    } while (abs(cosinus - cosinus_previous) > ACCURACY);

    cout << "cos(" << x << ") ~= " << cosinus << endl;
    return 0;
}
```

Second Approach

```
#include <iostream>
#include <cmath>
using namespace std;
```

```
const double ACCURACY = 0.00001;

int main() {
    int i, sign;
    double factorial;
    double cosinus, cosinus_previous, x;

    cin >> x;

    sign = -1;
    cosinus = 1;
    i = 2;
    factorial = 1;
    do {
        cosinus_previous = cosinus;

        factorial *= i * (i - 1);

        cosinus += sign * pow(x, i) / factorial;

        sign = -sign;
        i += 2;
    } while (abs(cosinus - cosinus_previous) > ACCURACY);

    cout << "cos(" << x << ") ~= " << cosinus << endl;
    return 0;
}
```

30. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i;
    double maximum, total, t;

    maximum = -460;
    total = 0;
    for (i = 1; i <= 31; i++) {
        cout << "Enter temperature for day " << i << ": ";
        cin >> t;
        while (t < -459.67) {
            cout << "Error! Wrong temperature." << endl;
            cout << "Enter temperature for day " << i << ": ";
            cin >> t;
        }
        total += t;
        if (t > maximum) {
            maximum = t;
        }
    }
    cout << total / 31 << " " << maximum << endl;
```

```
    return 0;  
}
```

31. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    int hour, max_hour, max_minutes, min_hour, min_minutes, minutes;  
    double level, maximum, minimum;  
  
    cin >> level;  
    if (level != 9999) {  
        cin >> hour;  
        cin >> minutes;  
  
        maximum = level;  
        max_hour = hour;  
        max_minutes = minutes;  
  
        minimum = level;  
        min_hour = hour;  
        min_minutes = minutes;  
  
        cin >> level;  
        while (level != 9999) {  
            cin >> hour;  
            cin >> minutes;  
  
            if (level > maximum) {  
                maximum = level;  
                max_hour = hour;  
                max_minutes = minutes;  
            }  
  
            if (level < minimum) {  
                minimum = level;  
                min_hour = hour;  
                min_minutes = minutes;  
            }  
  
            cin >> level;  
        }  
  
        cout << maximum << ", " << max_hour << ", " << max_minutes << endl;  
        cout << minimum << ", " << min_hour << ", " << min_minutes << endl;  
    }  
    return 0;  
}
```

32. Solution

```
#include <iostream>
```

```
using namespace std;
int main() {
    int a, b, c, i;
    bool failure;

    string alphabet = "abcdefghijklmnopqrstuvwxyz";

    do {
        cout << "Enter an integer between 1 and 26: ";
        cin >> a;

        failure = false;
        if (a < 1) {
            cout << "Please enter positive integers!" << endl;
            failure = true;
        }
        else if (a > 26) {
            cout << "Please enter a value less than or equal to 26!" << endl;
            failure = true;
        }
    } while (failure);

    do {
        cout << "Enter an integer between 1 and 26: ";
        cin >> b;

        failure = false;
        if (b < 1) {
            cout << "Please enter positive integers!" << endl;
            failure = true;
        }
        else if (b > 26) {
            cout << "Please enter a value less than or equal to 26!" << endl;
            failure = true;
        }
    } while (failure);

    if (a > b) {
        c = a;
        a = b;
        b = c;
    }

    for (i = a; i <= b; i++) {
        cout << alphabet[i - 1];
    }
    return 0;
}
```

33. Solution

```
#include <iostream>
#include <ctime>
```

```
#include <cstdlib>
using namespace std;
int main() {
    int attempts, guess, secret_number;

    srand(time(NULL));

    secret_number = 1 + rand() % 100;

    attempts = 1;
    cout << "Enter a guess: ";
    cin >> guess;
    while (guess != secret_number) {
        if (guess > secret_number) {
            cout << "Your guess is bigger than my secret number. Try again." << endl;
        }
        else {
            cout << "Your guess is smaller than my secret number. Try again." << endl;
        }
        attempts++;
        cout << "Enter a guess: ";
        cin >> guess;
    }
    cout << "You found it!" << endl;
    cout << "Attempts: " << attempts << endl;
    return 0;
}
```

34. Solution

```
#include <iostream>
#include <ctime>
#include <cstdlib>
using namespace std;
int main() {
    int attempts = 0, first_player_attempts = 0, guess, i, secret_number;

    srand(time(NULL));

    for (i = 1; i <= 2; i++) {
        secret_number = 1 + rand() % 100;

        attempts = 1;
        cout << "Enter a guess: ";
        cin >> guess;
        while (guess != secret_number) {
            if (guess > secret_number) {
                cout << "Your guess is bigger than my secret number. Try again." << endl;
            }
            else {
                cout << "Your guess is smaller than my secret number. Try again." << endl;
            }
            attempts++;
        }
        first_player_attempts += attempts;
    }
    cout << "First player's total attempts: " << first_player_attempts << endl;
}
```

```
    cout << "Enter a guess: ";
    cin >> guess;
}
cout << "You found it!" << endl;
cout << "Attempts: " << attempts << endl;

if (i == 1) {
    first_player_attempts = attempts;
}
}

if (first_player_attempts < attempts) {
    cout << "First player wins!" << endl;
}
else if (first_player_attempts > attempts) {
    cout << "Second player wins!" << endl;
}
else {
    cout << "It's a draw" << endl;
}
return 0;
}
```

35. Solution

```
#include <iostream>
using namespace std;
int main() {
    int choice, diagonal;

    while (true) {
        cout << "1. 4/3 TV Screen" << endl;
        cout << "2. 16/9 TV Screen" << endl;
        cout << "3. Exit" << endl;
        cout << "Enter a choice: ";
        cin >> choice;

        if (choice == 3) {
            break;
        }
        else if (choice == 1) {
            cout << "Enter diagonal: " << endl;
            cin >> diagonal;
            cout << "Width: " << diagonal * 0.8 << endl;
            cout << "Height: " << diagonal * 0.6 << endl;
        }
        else if (choice == 2) {
            cout << "Enter diagonal: " << endl;
            cin >> diagonal;
            cout << "Width: " << diagonal * 0.87 << endl;
            cout << "Height: " << diagonal * 0.49 << endl;
        }
    }
}
```

```
    }
    return 0;
}
```

36. Solution

```
#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    int count_a, count_a_boys, count_b, count_cdef_girls, grade;
    int i, maximum, minimum, n, total, total_a, total_a_boys, total_b;
    string gender;

    cout << "Enter total number of students: ";
    cin >> n;
    while (n <= 0) {
        cout << "Wrong number. Please enter total number of students: ";
        cin >> n;
    }

    total = 0;
    total_a = 0;
    count_a = 0;
    total_b = 0;
    count_b = 0;
    total_a_boys = 0;
    count_a_boys = 0;
    count_cdef_girls = 0;

    maximum = -1;
    minimum = 101;

    for (i = 1; i <= n; i++) {
        cout << "Enter grade for student No " << i << ": ";
        cin >> grade;
        while (grade < 0 || grade > 100) {
            cout << "Wrong grade. Please enter grade for student No " << i << ": ";
            cin >> grade;
        }

        cout << "Enter gender for student No " << i << ": ";
        cin >> gender;
        gender = to_upper_copy(gender);
        while (gender != "M" && gender != "F") {
            cout << "Wrong gender. Please enter gender for student No " << i << ": ";
            cin >> gender;
            gender = to_upper_copy(gender);
        }

        if (grade >= 90 && grade <= 100) {
            total_a += grade;
        }
    }
}
```

```
count_a++;
if (gender == "M") {
    total_a_boys += grade;
    count_a_boys++;
}
}
else if (grade >= 80 && grade <= 89) {
    total_b += grade;
    count_b++;
}
else {
    if (gender == "F") {
        count_cdef_girls++;
    }
}

if (grade > maximum) {
    maximum = grade;
}

if (grade < minimum) {
    minimum = grade;
}

total += grade;
}

if (count_a > 0) {
    cout << "The average value of those who got an 'A' is: ";
    cout << total_a / (double)count_a << endl;
}
if (count_b > 0) {
    cout << "The average value of those who got a 'B' is: ";
    cout << total_b / (double)count_b << endl;
}
if (count_a_boys > 0) {
    cout << "The average value of boys who got an 'A' is: ";
    cout << total_a_boys / (double)count_a_boys << endl;
}

cout << "The total number of girls that got less than 'B' is: " << count_cdef_girls << endl;
cout << "The highest grade is: " << maximum << endl;
cout << "The lowest grade is: " << minimum << endl;
cout << "The average grade of the whole class is: " << total / (double)n << endl;
return 0;
}
```

37. Solution

```
#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
```

```
int main() {
    double amount, discount;
    string answer;

    do {
        cout << "Enter amount: ";
        cin >> amount;
        while (amount <= 0) {
            cout << "Wrong amount. Please enter amount: " << endl;
            cin >> amount;
        }

        if (amount < 20) {
            discount = 0;
        }
        else if (amount < 50) {
            discount = 3;
        }
        else if (amount < 100) {
            discount = 5;
        }
        else {
            discount = 10;
        }

        cout << "Discount: " << discount << "%" << endl;
        cout << "Amount to pay (discount included): " << amount - amount * discount / 100 << endl;

        cout << "Would you like to repeat? ";
        cin >> answer;
        answer = to_upper_copy(answer);
    } while (answer == "YES");
    return 0;
}
```

38. Solution

```
#include <iostream>
using namespace std;
const double TAX_RATE = 0.25;

int main() {
    int kwh;
    double t;

    cout << "Enter number of Kilowatt-hours consumed: ";
    cin >> kwh;
    while (kwh < 0 && kwh != -1) {
        cout << "Wrong value. Please enter number of Kilowatt-hours consumed: ";
        cin >> kwh;
    }

    while (kwh != -1) {
```

```
if (kwh <= 400) {
    t = kwh * 0.11;
}
else if (kwh <= 1500) {
    t = 400 * 0.11 + (kwh - 400) * 0.22;
}
else if (kwh <= 3500) {
    t = 400 * 0.11 + 1100 * 0.22 + (kwh - 1500) * 0.25;
}
else {
    t = 400 * 0.11 + 1100 * 0.22 + 2000 * 0.25 + (kwh - 3500) * 0.50;
}

t += t * TAX_RATE;
cout << "Total amount to pay (taxes included): " << t << endl;

cout << "Enter number of Kilowatt-hours consumed: ";
cin >> kwh;
while (kwh < 0 && kwh != -1) {
    cout << "Wrong value. Please enter number of Kilowatt-hours consumed: ";
    cin >> kwh;
}
return 0;
}
```

Review in "Loop Control Structures"

Review Crossword Puzzle

1.



Chapter 31

31.13 Review Questions: True/False

- | | |
|-----------|-----------|
| 1. true | 21. true |
| 2. true | 22. false |
| 3. false | 23. true |
| 4. false | 24. false |
| 5. false | 25. true |
| 6. true | 26. false |
| 7. false | 27. false |
| 8. true | 28. true |
| 9. false | 29. false |
| 10. true | 30. true |
| 11. true | 31. true |
| 12. true | 32. false |
| 13. false | 33. false |
| 14. false | 34. true |
| 15. false | 35. true |
| 16. true | 36. true |
| 17. false | 37. false |
| 18. true | 38. false |
| 19. true | 39. true |
| 20. false | |

31.14 Review Questions: Multiple Choice

- | | |
|------|-------|
| 1. b | 8. d |
| 2. a | 9. c |
| 3. c | 10. a |
| 4. b | 11. b |
| 5. d | 12. a |
| 6. b | 13. b |
| 7. d | 14. b |

31.15 Review Exercises

1. Solution

Weights =

170	0	}	People
190	1		
193	2		
165	3		
200	4		

2. Solution

Names =

John Thompson
Chloe Brown
Ryan Miller
Antony Harris
Alexander Lewis
Samantha Clark
Ava Parker

Weights =

170
190
193
165
200
170
172

0 1 2 3 4 5 6

People

3. Solution

Names =

Toba
Issyk Kul
Baikal
Crater
Karakul

Areas =

440	438	437
2408	2405	2402
12248	12247	12240
21	20	18
150	145	142

0 1 2

0 1 2 3 4

Months

Lakes

June July August

4. Solution

Dimensions

0	1	2
10	31	15
15	12	17
22	10	18
22	20	12
26	25	14
66	26	21
54	34	24
64	28	22
34	12	18
33	10	10

0 1 2

0 1 2 3 4 5 6 7 8 9

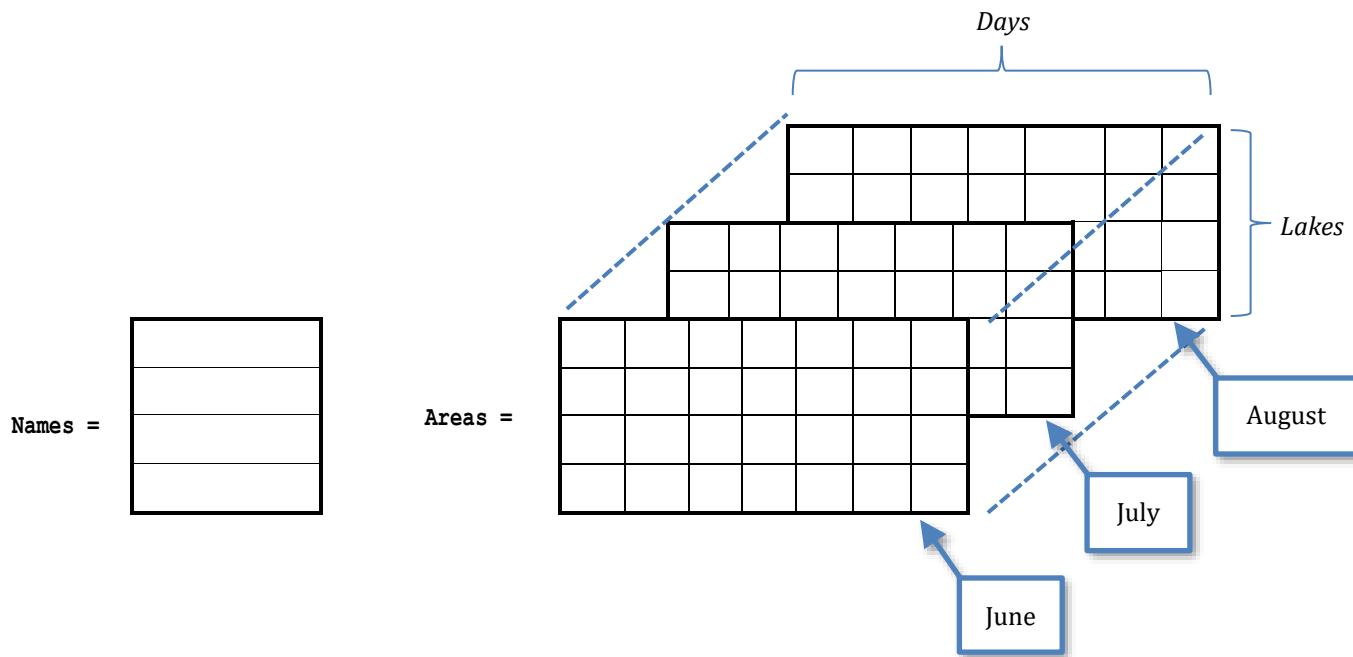
Boxes =

Width Height Length

5. Solution

Names =	Areas =	Depths =	Lakes
Toba	440	1660	0
Issyk Kul	2408	2192	1
Baikal	12248	5380	2
Crater	21	1950	3
Karakul	150	750	4
Quesnel	103	2000	5
Urmia	2317	52	6
Albert	2045	190	7

6. Solution



7. Solution

Step	Statement	x	a[0]	a[1]	a[2]
1	int a[3]	?	?	?	?
2	a[2] = 1	?	?	?	1
3	x = 0	0	?	?	1
4	a[x + a[2]] = 4	0	?	4	1
5	a[x] = a[x + 1] * 4	0	16	4	1

8. Solution

Step	Statement	x	a[0]	a[1]	a[2]	a[3]	a[4]
1	int a[5]	?	?	?	?	?	?
2	a[1] = 5	?	?	5	?	?	?
3	x = 0	0	?	5	?	?	?
4	a[x] = 4	0	4	5	?	?	?
5	a[a[0]] = a[x + 1] % 3	0	4	5	?	?	2
6	a[a[0] / 2] = 10	0	4	5	10	?	2
7	x += 2	2	4	5	10	?	2
8	a[x + 1] = a[x] + 9	2	4	5	10	19	2

9. Solution

For input value of 3

Step	Statement	x	a[0]	a[1]	a[2]	a[3]
1	int a[4]	?	?	?	?	?
2	cin >> a[1]	?	?	3	?	?
3	x = 0	0	?	3	?	?
4	a[x] = 3	0	3	3	?	?
5	a[a[0]] = a[x + 1] % 2	0	3	3	?	1
6	a[a[0] % 2] = 10	0	3	10	?	1
7	x++	1	3	10	?	1
8	a[x + 1] = a[x] + 9	1	3	10	19	1

For input value of 4

Step	Statement	x	a[0]	a[1]	a[2]	a[3]
1	int a[4]	?	?	?	?	?
2	cin >> a[1]	?	?	4	?	?
3	x = 0	0	?	4	?	?
4	a[x] = 3	0	3	4	?	?
5	a[a[0]] = a[x + 1] % 2	0	3	4	?	0
6	a[a[0] % 2] = 10	0	3	10	?	0
7	x++	1	3	10	?	0
8	a[x + 1] = a[x] + 9	1	3	10	19	0

For input value of 1

Step	Statement	x	a[0]	a[1]	a[2]	a[3]
1	int a[4]	?	?	?	?	?
2	cin >> a[1]	?	?	1	?	?
3	x = 0	0	?	1	?	?

4	$a[x] = 3$	0	3	1	?	?
5	$a[a[0]] = a[x + 1] \% 2$	0	3	1	?	3
6	$a[a[0] \% 2] = 10$	0	3	10	?	3
7	$x++$	1	3	10	?	3
8	$a[x + 1] = a[x] + 9$	1	3	10	19	3

10. Solution

For input value of 100

Step	Statement	x	a[0]	a[1]	a[2]	a[3]
1	int a[4]	?	?	?	?	?
2	cin >> a[1]	?	?	100	?	?
3	$x = 0$	0	?	100	?	?
4	$a[x] = 3$	0	3	100	?	?
5	$a[a[0]] = a[x + 1] \% 10$	0	3	100	?	0
6	if ($a[3] > 5$)	false				
7	$a[2] = 3$	0	3	100	3	0

For input value of 108

Step	Statement	x	a[0]	a[1]	a[2]	a[3]
1	int a[4]	?	?	?	?	?
2	cin >> a[1]	?	?	108	?	?
3	$x = 0$	0	?	108	?	?
4	$a[x] = 3$	0	3	108	?	?
5	$a[a[0]] = a[x + 1] \% 10$	0	3	108	?	8
6	if ($a[3] > 5$)	true				
7	$a[a[0] \% 2] = 9$	0	3	9	?	8
8	$x += 1$	1	3	9	?	8
9	$a[x + 1] = a[x] + 9$	1	3	9	18	8

For input value of 1

Step	Statement	x	a[0]	a[1]	a[2]	a[3]
1	int a[4]	?	?	?	?	?
2	cin >> a[1]	?	?	1	?	?
3	$x = 0$	0	?	1	?	?
4	$a[x] = 3$	0	3	1	?	?
5	$a[a[0]] = a[x + 1] \% 10$	0	3	1	?	1
6	if ($a[3] > 5$)	false				
7	$a[2] = 3$	0	3	1	3	1

11. Solution

Step	Statement	x	y	a[0]	a[1]	a[2]
1	int a[3]	?	?	?	?	?
2	x = 4	4	?	?	?	?
3	y = x - 1	4	3	?	?	?
4, 5	if (x > y) a[0] = 1; else a[0] = y;	4	3	1	?	?
6	a[1] = x + 3	4	3	1	7	?
7	y = y - 1	4	2	1	7	?
8	a[y] = (x + 5) % 2	4	2	1	7	1

12. Solution

Step	Statement	i	a[0]	a[1]	a[2]	a[3]	a[4]	a[5]
1	int a[] = {17, 12, 45, 12, 12, 49}	?	17	12	45	12	12	49
2	i = 0	0	17	12	45	12	12	49
3	i <= 5				true			
4	if (a[i] == 12)				false			
5	a[i]++	0	18	12	45	12	12	49
6	i++	1	18	12	45	12	12	49
7	i <= 5				true			
8	if (a[i] == 12)				true			
9	a[i]--	1	18	11	45	12	12	49
10	i++	2	18	11	45	12	12	49
11	i <= 5				true			
12	if (a[i] == 12)				false			
13	a[i]++	2	18	11	46	12	12	49
14	i++	3	18	11	46	12	12	49
15	i <= 5				true			
16	if (a[i] == 12)				true			
17	a[i]--	3	18	11	46	11	12	49
18	i++	4	18	11	46	11	12	49
19	i <= 5				true			
20	if (a[i] == 12)				true			
21	a[i]--	4	18	11	46	11	11	49
22	i++	5	18	11	46	11	11	49
23	i <= 5				true			

24	if (a[i] == 12)	false						
25	a[i]++	5	18	11	46	11	11	50
26	i++	6	18	11	46	11	11	50
27	i <= 5	false						

13. Solution

Step	Statement	i	a[0]	a[1]	a[2]	a[3]	a[4]	a[5]
1	int a[] = {10, 15, 12, 23, 22, 19}	?	10	15	12	23	22	19
2	i = 1	1	10	15	12	23	22	19
3	i <= 4	true						
4	a[i] = a[i + 1] + a[i - 1]	1	10	22	12	23	22	19
5	i++	2	10	22	12	23	22	19
6	i <= 4	true						
7	a[i] = a[i + 1] + a[i - 1]	2	10	22	45	23	22	19
8	i++	3	10	22	45	23	22	19
9	i <= 4	true						
10	a[i] = a[i + 1] + a[i - 1]	3	10	22	45	67	22	19
11	i++	4	10	22	45	67	22	19
12	i <= 4	true						
13	a[i] = a[i + 1] + a[i - 1]	4	10	22	45	67	86	19
14	i++	5	10	22	45	67	86	19
15	i <= 4	false						

14. Solution

It displays:

Navajo

Cherokee

Sioux

15. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
const int ELEMENTS = 100;

int main() {
    int i;

    double a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }
}
```

```
    for (i = 0; i <= ELEMENTS - 1; i++) {  
        cout << pow(a[i], 3) << endl;  
    }  
    return 0;  
}
```

16. Solution

```
#include <iostream>  
#include <cmath>  
using namespace std;  
const int ELEMENTS = 80;  
  
int main() {  
    int i;  
  
    double a[ELEMENTS];  
    for (i = 0; i <= ELEMENTS - 1; i++) {  
        cin >> a[i];  
    }  
  
    for (i = 0; i <= ELEMENTS - 1; i++) {  
        a[i] = pow(a[i], 2);  
    }  
  
    for (i = ELEMENTS - 1; i >= 0; i--) {  
        cout << a[i] << endl;  
    }  
    return 0;  
}
```

17. Solution

```
#include <iostream>  
using namespace std;  
const int ELEMENTS = 90;  
  
int main() {  
    int i;  
  
    int a[ELEMENTS];  
    for (i = 0; i <= ELEMENTS - 1; i++) {  
        cin >> a[i];  
    }  
  
    for (i = ELEMENTS - 1; i >= 0; i--) {  
        if (a[i] % 5 == 0) {  
            cout << a[i] << endl;  
        }  
    }  
    return 0;  
}
```

18. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 50;

int main() {
    int i;

    int a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }

    for (i = 0; i <= ELEMENTS - 1; i++) {
        if (a[i] % 2 == 0 || a[i] > 10) {
            cout << a[i] << endl;
        }
    }
    return 0;
}
```

19. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 30;

int main() {
    int i;
    double total;

    double a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }

    total = 0;
    for (i = 0; i <= ELEMENTS - 1; i++) {
        if (a[i] > 0) {
            total += a[i];
        }
    }
    cout << total << endl;
    return 0;
}
```

20. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 50;

int main() {
```

```
int i, total;

int a[ELEMENTS];
for (i = 0; i <= ELEMENTS - 1; i++) {
    cin >> a[i];
}

total = 0;
for (i = 0; i <= ELEMENTS - 1; i++) {
    if (a[i] >= 10 && a[i] <= 99) {
        total += a[i];
    }
}
cout << total << endl;
return 0;
}
```

21. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 40;

int main() {
    int i;
    double sum_neg, sum_pos;

    double a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }

    sum_pos = sum_neg = 0;
    for (i = 0; i <= ELEMENTS - 1; i++) {
        if (a[i] > 0) {
            sum_pos += a[i];
        }
        else if (a[i] < 0) {
            sum_neg += a[i];
        }
    }
    cout << sum_pos << ", " << sum_neg << endl;
    return 0;
}
```

22. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 20;

int main() {
    int i;
    double total;
```

```
double a[ELEMENTS];
for (i = 0; i <= ELEMENTS - 1; i++) {
    cin >> a[i];
}

total = 0;
for (i = 0; i <= ELEMENTS - 1; i++) {
    total += a[i];
}
cout << total / ELEMENTS << endl;
return 0;
}
```

23. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 50;

int main() {
    int i;

    int a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cout << "Enter an integer: ";
        cin >> a[i];
    }

    for (i = 0; i <= ELEMENTS - 1; i++) {
        if (a[i] < 20) {
            cout << a[i] << endl;
        }
    }
    return 0;
}
```

24. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 60;

int main() {
    int i;

    double a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cout << "Enter a number: ";
        cin >> a[i];
    }

    for (i = 0; i <= ELEMENTS - 1; i += 2) {
        cout << a[i] << endl;
    }
    return 0;
}
```

```
}
```

25. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 20;

int main() {
    int i;
    double total;

    double a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cout << "Enter a number: ";
        cin >> a[i];
    }

    total = 0;
    for (i = 0; i <= ELEMENTS - 1; i += 2) {
        total += a[i];
    }
    cout << total << endl;
    return 0;
}
```

26. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 100;

int main() {
    int i;
    int a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        a[i] = i + 1;
    }
    ...
}
```

27. Solution

First Approach

```
#include <iostream>
using namespace std;
const int ELEMENTS = 100;

int main() {
    int i, k;
    int a[ELEMENTS];
    k = 2;
    for (i = 0; i <= ELEMENTS - 1; i++) {
        a[i] = k;
        k += 2;
    }
    ...
}
```

Second Approach

```
#include <iostream>
using namespace std;
const int ELEMENTS = 100;

int main() {
    int i;
    int a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        a[i] = (i + 1) * 2;
    }
    ...
}
```

28. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int i, n;
    int a[100];

    cout << "Enter N: ";
    cin >> n;
    while (n < 1 || n > 100) {
        cout << "Error! Value must be between 1 and 100" << endl;
        cout << "Enter N: ";
        cin >> n;
    }

    for (i = 1; i <= n; i++) {
        a[i - 1] = pow(i, 2);
    }

    for (i = 0; i <= n - 1; i++) {
        cout << a[i] << endl;
    }
    return 0;
}
```

29. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 10;

int main() {
    int i;

    double a[ELEMENTS];
    for (i = 1; i <= ELEMENTS - 1; i++) {
        cout << "Enter a number: ";
        cin >> a[i];
    }

    for (i = 0; i <= ELEMENTS - 1; i++) {
```

```
    if (a[i] == (int)a[i]) {
        cout << i << endl;
    }
}
return 0;
}
```

30. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 50;

int main() {
    int i, count;

    double a[ELEMENTS];
    for (i = 1; i <= ELEMENTS - 1; i++) {
        cout << "Enter a number: ";
        cin >> a[i];
    }

    count = 0;
    for (i = 0; i <= ELEMENTS - 1; i++) {
        if (a[i] < 0) {
            count++;
        }
    }
    cout << count << endl;
    return 0;
}
```

31. Solution

```
#include <iostream>
using namespace std;
const int WORDS = 50;

int main() {
    int i;

    string a[WORDS];
    for (i = 0; i <= WORDS - 1; i++) {
        cin >> a[i];
    }

    for (i = 0; i <= WORDS - 1; i++) {
        if (a[i].length() >= 10 ) {
            cout << a[i] << endl;
        }
    }
    return 0;
}
```

32. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 30;

int main() {
    int i, k;

    string words[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> words[i];
    }

    int length_limits[] = {0, 5, 10, 20};

    for (k = 1; k <= 3; k++) {
        for (i = 0; i <= ELEMENTS - 1; i++) {
            if (words[i].length() >= length_limits[k - 1] && words[i].length() < length_limits[k]) {
                cout << words[i] << endl;
            }
        }
    }
    return 0;
}
```

33. Solution

```
#include <iostream>
using namespace std;
const int WORDS = 40;

int main() {
    int count, i, j;

    string a[WORDS];
    for (i = 0; i <= WORDS - 1; i++) {
        cout << "Enter a word: ";
        cin >> a[i];
    }

    for (i = 0; i <= WORDS - 1; i++) {
        count = 0;
        for (j = 0; j <= a[i].length() - 1; j++) {
            if (a[i][j] == 'w') { //Alternatively use: if (a[i].substr(j, 1) == "w")
                count++;
            }
        }
        if (count >= 2) {
            cout << a[i] << endl;
        }
    }
    return 0;
}
```

Chapter 32

32.7 Review Questions: True/False

- | | |
|-----------|-----------|
| 1. false | 15. true |
| 2. true | 16. true |
| 3. false | 17. true |
| 4. false | 18. true |
| 5. false | 19. false |
| 6. true | 20. true |
| 7. false | 21. true |
| 8. true | 22. true |
| 9. true | 23. false |
| 10. true | 24. true |
| 11. true | 25. true |
| 12. true | 26. true |
| 13. false | 27. false |
| 14. true | |

32.8 Review Questions: Multiple Choice

1. b
2. b
3. c
4. a
5. d
6. a
7. d
8. c
9. c
10. c
11. b

32.9 Review Exercises

1. Solution

Step	Statement	x	a						
1	int a[2][3]	?	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?
?	?	?							
?	?	?							
2	a[0][2] = 1	?	<table border="1"> <tr><td>?</td><td>?</td><td>1</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	1	?	?	?
?	?	1							
?	?	?							
3	x = 0	0	<table border="1"> <tr><td>?</td><td>?</td><td>1</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	1	?	?	?
?	?	1							
?	?	?							
4	a[0][x] = 9	0	<table border="1"> <tr><td>9</td><td>?</td><td>1</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	9	?	1	?	?	?
9	?	1							
?	?	?							
5	a[0][x + a[0][2]] = 4	0	<table border="1"> <tr><td>9</td><td>4</td><td>1</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	9	4	1	?	?	?
9	4	1							
?	?	?							
6	a[a[0][2]][2] = 19	0	<table border="1"> <tr><td>9</td><td>4</td><td>1</td></tr> <tr><td>?</td><td>?</td><td>19</td></tr> </table>	9	4	1	?	?	19
9	4	1							
?	?	19							
7	a[a[0][2]][x + 1] = 13	0	<table border="1"> <tr><td>9</td><td>4</td><td>1</td></tr> <tr><td>?</td><td>13</td><td>19</td></tr> </table>	9	4	1	?	13	19
9	4	1							
?	13	19							
8	a[a[0][2]][x] = 15	0	<table border="1"> <tr><td>9</td><td>4</td><td>1</td></tr> <tr><td>15</td><td>13</td><td>19</td></tr> </table>	9	4	1	15	13	19
9	4	1							
15	13	19							

2. Solution

Step	Statement	i	j	a						
1	int a[2][3]	?	?	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?
?	?	?								
?	?	?								
2	i = 0	0	?	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?
?	?	?								
?	?	?								
3	i <= 1			true						
4	j = 0	0	0	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?
?	?	?								
?	?	?								

5	$j \leq 2$	true								
6	$a[i][j] = (i + 1) * 5 + j$	0	0	<table border="1"> <tr><td>5</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	?	?	?	?	?
5	?	?								
?	?	?								
7	$j++$	0	1	<table border="1"> <tr><td>5</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	?	?	?	?	?
5	?	?								
?	?	?								
8	$j \leq 2$	true								
9	$a[i][j] = (i + 1) * 5 + j$	0	1	<table border="1"> <tr><td>5</td><td>6</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	6	?	?	?	?
5	6	?								
?	?	?								
10	$j++$	0	2	<table border="1"> <tr><td>5</td><td>6</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	6	?	?	?	?
5	6	?								
?	?	?								
11	$j \leq 2$	true								
12	$a[i][j] = (i + 1) * 5 + j$	0	2	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	6	7	?	?	?
5	6	7								
?	?	?								
13	$j++$	0	3	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	6	7	?	?	?
5	6	7								
?	?	?								
14	$j \leq 2$	false								
15	$i++$	1	3	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	6	7	?	?	?
5	6	7								
?	?	?								
16	$i \leq 1$	true								
17	$j = 0$	1	0	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	6	7	?	?	?
5	6	7								
?	?	?								
18	$j \leq 2$	true								
19	$a[i][j] = (i + 1) * 5 + j$	1	0	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>?</td><td>?</td></tr> </table>	5	6	7	10	?	?
5	6	7								
10	?	?								
20	$j++$	1	1	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>?</td><td>?</td></tr> </table>	5	6	7	10	?	?
5	6	7								
10	?	?								
21	$j \leq 2$	true								
22	$a[i][j] = (i + 1) * 5 + j$	1	1	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>11</td><td>?</td></tr> </table>	5	6	7	10	11	?
5	6	7								
10	11	?								
23	$j++$	1	2	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>11</td><td>?</td></tr> </table>	5	6	7	10	11	?
5	6	7								
10	11	?								

24	$j \leq 2$	true								
25	$a[i][j] = (i + 1) * 5 + j$	1	2	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>11</td><td>12</td></tr> </table>	5	6	7	10	11	12
5	6	7								
10	11	12								
26	$j++$	1	3	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>11</td><td>12</td></tr> </table>	5	6	7	10	11	12
5	6	7								
10	11	12								
27	$j \leq 2$	false								
28	$i++$	2	3	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>11</td><td>12</td></tr> </table>	5	6	7	10	11	12
5	6	7								
10	11	12								
29	$i \leq 1$	false								

3. Solution

Step	Statement	i	j	a									
1	<code>int a[3][3]</code>	?	?	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?	?	?	?
?	?	?											
?	?	?											
?	?	?											
2	$j = 0$?	0	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?	?	?	?
?	?	?											
?	?	?											
?	?	?											
3	$j \leq 2$	true											
4	$i = 0$	0	0	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?	?	?	?
?	?	?											
?	?	?											
?	?	?											
5	$i \leq 2$	true											
6	$a[i][j] = (i + 1) * 2 + j * 4$	0	0	<table border="1"> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	2	?	?	?	?	?	?	?	?
2	?	?											
?	?	?											
?	?	?											
7	$i++$	1	0	<table border="1"> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	2	?	?	?	?	?	?	?	?
2	?	?											
?	?	?											
?	?	?											
8	$i \leq 2$	true											
9	$a[i][j] = (i + 1) * 2 + j * 4$	1	0	<table border="1"> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	2	?	?	4	?	?	?	?	?
2	?	?											
4	?	?											
?	?	?											

10	i++	2	0	<table border="1"> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	2	?	?	4	?	?	?	?	?
2	?	?											
4	?	?											
?	?	?											
11	i <= 2			true									
12	a[i][j] = (i + 1) * 2 + j * 4	2	0	<table border="1"> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </table>	2	?	?	4	?	?	6	?	?
2	?	?											
4	?	?											
6	?	?											
13	i++	3	0	<table border="1"> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </table>	2	?	?	4	?	?	6	?	?
2	?	?											
4	?	?											
6	?	?											
14	i <= 2			false									
15	j++	3	1	<table border="1"> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </table>	2	?	?	4	?	?	6	?	?
2	?	?											
4	?	?											
6	?	?											
16	j <= 2			true									
17	i = 0	0	1	<table border="1"> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </table>	2	?	?	4	?	?	6	?	?
2	?	?											
4	?	?											
6	?	?											
18	i <= 2			true									
19	a[i][j] = (i + 1) * 2 + j * 4	0	1	<table border="1"> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </table>	2	6	?	4	?	?	6	?	?
2	6	?											
4	?	?											
6	?	?											
20	i++	1	1	<table border="1"> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </table>	2	6	?	4	?	?	6	?	?
2	6	?											
4	?	?											
6	?	?											
21	i <= 2			true									
22	a[i][j] = (i + 1) * 2 + j * 4	1	1	<table border="1"> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </table>	2	6	?	4	8	?	6	?	?
2	6	?											
4	8	?											
6	?	?											
23	i++	2	1	<table border="1"> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </table>	2	6	?	4	8	?	6	?	?
2	6	?											
4	8	?											
6	?	?											
24	i <= 2			true									
25	a[i][j] = (i + 1) * 2 + j * 4	2	1	<table border="1"> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </table>	2	6	?	4	8	?	6	10	?
2	6	?											
4	8	?											
6	10	?											

26	i++	3	1	<table border="1"> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </table>	2	6	?	4	8	?	6	10	?
2	6	?											
4	8	?											
6	10	?											
27	i <= 2			false									
28	j++	3	2	<table border="1"> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </table>	2	6	?	4	8	?	6	10	?
2	6	?											
4	8	?											
6	10	?											
29	j <= 2			true									
30	i = 0	0	2	<table border="1"> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </table>	2	6	?	4	8	?	6	10	?
2	6	?											
4	8	?											
6	10	?											
31	i <= 2			true									
32	a[i][j] = (i + 1) * 2 + j * 4	0	2	<table border="1"> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </table>	2	6	10	4	8	?	6	10	?
2	6	10											
4	8	?											
6	10	?											
33	i++	1	2	<table border="1"> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </table>	2	6	10	4	8	?	6	10	?
2	6	10											
4	8	?											
6	10	?											
34	i <= 2			true									
35	a[i][j] = (i + 1) * 2 + j * 4	1	2	<table border="1"> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>12</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </table>	2	6	10	4	8	12	6	10	?
2	6	10											
4	8	12											
6	10	?											
36	i++	2	2	<table border="1"> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>12</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </table>	2	6	10	4	8	12	6	10	?
2	6	10											
4	8	12											
6	10	?											
37	i <= 2			true									
38	a[i][j] = (i + 1) * 2 + j * 4	2	2	<table border="1"> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>12</td></tr> <tr><td>6</td><td>10</td><td>14</td></tr> </table>	2	6	10	4	8	12	6	10	14
2	6	10											
4	8	12											
6	10	14											
39	i++	3	2	<table border="1"> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>12</td></tr> <tr><td>6</td><td>10</td><td>14</td></tr> </table>	2	6	10	4	8	12	6	10	14
2	6	10											
4	8	12											
6	10	14											
40	i <= 2			false									
41	j++	3	3	<table border="1"> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>12</td></tr> <tr><td>6</td><td>10</td><td>14</td></tr> </table>	2	6	10	4	8	12	6	10	14
2	6	10											
4	8	12											
6	10	14											

42	j <= 2	false
----	--------	-------

4. Solution

For input value of 5

0	5	10
0	6	12

For input value of 9

0	9	18
0	10	20

For input value of 3

0	3	6
0	4	8

5. Solution

For input value of 13

0	3	3
0	17	18

For input value of 10

0	10	3
0	11	15

For input value of 8

3	3	3
11	12	13

6. Solution

19	5	31
28	6	20

7. Solution

26	29
37	34
59	49

8. Solution

- i. -1 15 22 25 12 16 7 9 1
- ii. 7 9 1 25 12 16 -1 15 22
- iii. 22 15 -1 16 12 25 1 9 7
- iv. 1 9 7 16 12 25 22 15 -1

v. -1 25 7 15 12 9 22 16 1
vi. 7 25 -1 9 12 15 1 16 22
vii. 22 16 1 15 12 9 -1 25 7
viii. 1 16 22 9 12 15 7 25 -1

9. Solution

```
#include <iostream>
using namespace std;
const int ROWS = 10;
const int COLUMNS = 15;

int main() {
    int i, j;

    int a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            if (a[i][j] % 2 != 0) {
                cout << i << ", " << j << endl;
            }
        }
    }
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
const int ROWS = 10;
const int COLUMNS = 6;

int main() {
    int i, j;

    double a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j += 2) {
            cout << a[i][j] << endl;
        }
    }
}
```

```
    }
    return 0;
}
```

11. Solution

```
#include <iostream>
using namespace std;
const int ROWS = 12;
const int COLUMNS = 8;

int main() {
    int i, j;
    double total;

    double a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    total = 0;
    for (i = 1; i <= ROWS - 1; i += 2) {
        for (j = 0; j <= COLUMNS - 1; j += 2) {
            total += a[i][j];
        }
    }
    cout << total << endl;
    return 0;
}
```

12. Solution

```
#include <iostream>
using namespace std;
const int N = 8 ;

int main() {
    int i, j, k;
    double sum_antidiagonal, sum_diagonal;

    double a[N][N];
    for (i = 0; i <= N - 1; i++) {
        for (j = 0; j <= N - 1; j++) {
            cin >> a[i][j];
        }
    }

    sum_diagonal = 0;
    sum_antidiagonal = 0;
    for (k = 0; k <= N - 1; k++) {
        sum_diagonal += a[k][k];
    }
}
```

```
    sum_antidiagonal += a[k][N - k - 1];
}
cout << sum_diagonal / N << ", " << sum_antidiagonal / N << endl;
return 0;
}
```

13. Solution

```
#include <iostream>
using namespace std;
const int N = 5;

int main() {
    int i, j;

    int a[N][N];
    for (i = 0; i <= N - 1; i++) {
        for (j = 0; j <= N - 1; j++) {
            if (i == N - j - 1) {
                a[i][j] = 5;
            }
            else if (i > N - j - 1) {
                a[i][j] = 88;
            }
            else {
                a[i][j] = 11;
            }
        }
    }

    for (i = 0; i <= N - 1; i++) {
        for (j = 0; j <= N - 1; j++) {
            cout << a[i][j] << "\t";
        }
        cout << endl;
    }
    return 0;
}
```

14. Solution

```
#include <iostream>
using namespace std;
const int N = 5;

int main() {
    int i, j;

    int a[N][N];
    for (i = 0; i <= N - 1; i++) {
        for (j = 0; j <= N - 1; j++) {
            if (i == N - j - 1) {
                a[i][j] = 5;
```

```
        }
    else if (i > N - j - 1) {
        a[i][j] = 88;
    }
    else {
        a[i][j] = 11;
    }
    if (i == j) {
        a[i][j] = 0;
    }
}
}

for (i = 0; i <= N - 1; i++) {
    for (j = 0; j <= N - 1; j++) {
        cout << a[i][j] << "\t";
    }
    cout << endl;
}
return 0;
}
```

15. Solution

```
#include <iostream>
using namespace std;
const int ROWS = 5;
const int COLUMNS = 4;

int main() {
    int i, j;

    double a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            if (a[i][j] == (int)(a[i][j])) {
                cout << i << ", " << j << endl;
            }
        }
    }
    return 0;
}
```

16. Solution

```
#include <iostream>
```

```
using namespace std;
const int ROWS = 10;
const int COLUMNS = 4;

int main() {
    int count, i, j;

    double a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    count = 0;
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            if (a[i][j] < 0) {
                count++;
            }
        }
    }
    cout << count << endl;
    return 0;
}
```

17. Solution

```
#include <iostream>
using namespace std;
const int ROWS = 3;
const int COLUMNS = 4;

int main() {
    int i, j;

    string a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cout << a[i][j] << " ";
        }
    }
    return 0;
}
```

18. Solution

```
#include <iostream>
using namespace std;
const int ROWS = 20;
const int COLUMNS = 14;

int main() {
    int i, j;

    string a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            if (a[i][j].length() < 5) {
                cout << a[i][j] << endl;
            }
        }
    }
    return 0;
}
```

19. Solution

First Approach

```
#include <iostream>
using namespace std;
const int ROWS = 20;
const int COLUMNS = 14;

int main() {
    int i, j, k;

    string a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    int length_limits[] = {5, 10, 20};

    for (k = 0; k <= 2; k++) {
        for (i = 0; i <= ROWS - 1; i++) {
            for (j = 0; j <= COLUMNS - 1; j++) {
                if (a[i][j].length() < length_limits[k]) {
                    cout << a[i][j] << endl;
                }
            }
        }
    }
}
```

```
        }
    }
}
return 0;
}
```

Second Approach

```
#include <iostream>
#include <cmath>
using namespace std;
const int ROWS = 20;
const int COLUMNS = 14;

int main() {
    int i, j, k;

    string a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    for (k = 0; k <= 2; k++) {
        for (i = 0; i <= ROWS - 1; i++) {
            for (j = 0; j <= COLUMNS - 1; j++) {
                if (a[i][j].length() < 5 * pow(2, k)) {
                    cout << a[i][j] << endl;
                }
            }
        }
    }
}
```

Chapter 33

33.7 Review Questions: True/False

- | | |
|----------|-----------|
| 1. true | 7. true |
| 2. false | 8. true |
| 3. true | 9. false |
| 4. false | 10. false |
| 5. false | 11. true |
| 6. false | |

33.8 Review Questions: Multiple Choice

- | | |
|------|------|
| 1. a | 4. b |
| 2. b | 5. a |
| 3. c | 6. a |

33.9 Review Exercises

1. Solution

```
#include <iostream>
using namespace std;
const int STUDENTS = 15;
const int TESTS = 5;

int main() {
    int i, j;

    int grades[STUDENTS][TESTS];
    for (i = 0; i <= STUDENTS - 1; i++) {
        for (j = 0; j <= TESTS - 1; j++) {
            cin >> grades[i][j];
        }
    }

    double average[STUDENTS];
    for (i = 0; i <= STUDENTS - 1; i++) {
        average[i] = 0;
        for (j = 0; j <= TESTS - 1; j++) {
            average[i] += grades[i][j];
        }
        average[i] /= TESTS;
    }

    for (i = 0; i <= STUDENTS - 1; i++) {
        cout << "Student No " << i + 1 << ": ";

        if (average[i] < 60) {
            cout << "E/F" << endl;
        }
        else if (average[i] < 70) {
            cout << "D" << endl;
        }
        else if (average[i] < 80) {
            cout << "C" << endl;
        }
        else if (average[i] < 90) {
            cout << "B" << endl;
        }
        else {
            cout << "A" << endl;
        }
    }
}
```

```
    }
    else if (average[i] < 80) {
        cout << "C" << endl;
    }
    else if (average[i] < 90) {
        cout << "B" << endl;
    }
    else {
        cout << "A" << endl;
    }
}
return 0;
}
```

2. Solution

```
#include <iostream>
using namespace std;
const int OBJECTS = 5;
const int FALLS = 10;

int main() {
    int i, j, total;

    int g[OBJECTS][FALLS];
    for (i = 0; i <= OBJECTS - 1; i++) {
        for (j = 0; j <= FALLS - 1; j++) {
            cin >> g[i][j];
        }
    }

    for (i = 0; i <= OBJECTS - 1; i++) {
        total = 0;
        for (j = 0; j <= FALLS - 1; j++) {
            total += g[i][j];
        }
        cout << "Average g for object No " << i + 1 << ":" << total / (double)FALLS << endl;
    }

    for (j = 0; j <= FALLS - 1; j++) {
        total = 0;
        for (i = 0; i <= OBJECTS - 1; i++) {
            total += g[i][j];
        }
        cout << "Average g for fall No " << j + 1 << ":" << total / (double)OBJECTS << endl;
    }

    total = 0;
    for (i = 0; i <= OBJECTS - 1; i++) {
        for (j = 0; j <= FALLS - 1; j++) {
            total += g[i][j];
        }
    }
}
```

```
    }
    cout << "Overall average g: " << total / (double)(OBJECTS * FALLS) << endl;
    return 0;
}
```

3. Solution

```
#include <iostream>
using namespace std;
const int PLAYERS = 15;
const int MATCHES = 12;

int main() {
    int i, j, total;

    int points[PLAYERS][MATCHES];
    for (i = 0; i <= PLAYERS - 1; i++) {
        for (j = 0; j <= MATCHES - 1; j++) {
            cin >> points[i][j];
        }
    }

    for (i = 0; i <= PLAYERS - 1; i++) {
        total = 0;
        for (j = 0; j <= MATCHES - 1; j++) {
            total += points[i][j];
        }
        cout << "Total number of points for player No " << i + 1 << ":" << total << endl;
    }

    for (j = 0; j <= MATCHES - 1; j++) {
        total = 0;
        for (i = 0; i <= PLAYERS - 1; i++) {
            total += points[i][j];
        }
        cout << "Total number of points for match No " << j + 1 << ":" << total << endl;
    }
    return 0;
}
```

4. Solution

```
#include <iostream>
using namespace std;
const int CITIES = 20;
const int HOURS = 24;

int main() {
    int i, j;
    double total;

    double temperatures[CITIES][HOURS];
    for (i = 0; i <= CITIES - 1; i++) {
```

```
    for (j = 0; j <= HOURS - 1; j++) {
        cin >> temperatures[i][j];
    }
}

for (j = 0; j <= HOURS - 1; j++) {
    total = 0;
    for (i = 0; i <= CITIES - 1; i++) {
        total += temperatures[i][j];
    }
    if (total / CITIES < 10) {
        cout << "Hour: " << j + 1 << endl;
    }
}
return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
const int PLAYERS = 24;
const int MATCHES = 10;

int main() {
    int i, j, total;

    string names[PLAYERS];
    int goals[PLAYERS] [MATCHES];
    for (i = 0; i <= PLAYERS - 1; i++) {
        cin >> names[i];
        for (j = 0; j <= MATCHES - 1; j++) {
            cin >> goals[i][j];
        }
    }

    for (i = 0; i <= PLAYERS - 1; i++) {
        total = 0;
        for (j = 0; j <= MATCHES - 1; j++) {
            total += goals[i][j];
        }
        cout << names[i] << ":" << total / (double)MATCHES << endl;
    }

    for (j = 0; j <= MATCHES - 1; j++) {
        total = 0;
        for (i = 0; i <= PLAYERS - 1; i++) {
            total += goals[i][j];
        }
        cout << "Match No " << j + 1 << ":" << total << endl;
    }
}
return 0;
```

```
}
```

6. Solution

```
#include <iostream>
using namespace std;
const int STUDENTS = 12;
const int LESSONS = 6;

int main() {
    int i, j, total;

    string names[STUDENTS];
    int grades[STUDENTS][LESSONS];
    for (i = 0; i <= STUDENTS - 1; i++) {
        cin >> names[i];
        for (j = 0; j <= LESSONS - 1; j++) {
            cin >> grades[i][j];
        }
    }

    double average[STUDENTS];
    for (i = 0; i <= STUDENTS - 1; i++) {
        total = 0;
        for (j = 0; j <= LESSONS - 1; j++) {
            total += grades[i][j];
        }
        average[i] = total / (double)LESSONS;
        cout << names[i] << ":" << average[i] << endl;
    }

    for (j = 0; j <= LESSONS - 1; j++) {
        total = 0;
        for (i = 0; i <= STUDENTS - 1; i++) {
            total += grades[i][j];
        }
        cout << total / (double)STUDENTS << endl;
    }

    for (i = 0; i <= STUDENTS - 1; i++) {
        if (average[i] < 60) {
            cout << names[i] << endl;
        }
    }

    for (i = 0; i <= STUDENTS - 1; i++) {
        if (average[i] > 89) {
            cout << names[i] << " Bravo!" << endl;
        }
    }
    return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
const int ARTISTS = 15;
const int JUDGES = 5;

int main() {
    int i, j, total;

    string judge_names[JUDGES];
    for (j = 0; j <= JUDGES - 1; j++) {
        cout << "Enter name for judge No " << j + 1 << ": ";
        cin >> judge_names[j];
    }

    string artist_names[ARTISTS];
    string song_titles[ARTISTS];
    int score[ARTISTS][JUDGES];
    for (i = 0; i <= ARTISTS - 1; i++) {
        cout << "Enter name for artist No " << i + 1 << ": ";
        cin >> artist_names[i];
        cout << "Enter song title for artist " << artist_names[i] << ": ";
        cin >> song_titles[i];
        for (j = 0; j <= JUDGES - 1; j++) {
            cout << "Enter score for artist: " << artist_names[i];
            cout << " gotten from judge " << judge_names[j] << ": ";
            cin >> score[i][j];
        }
    }

    for (i = 0; i <= ARTISTS - 1; i++) {
        total = 0;
        for (j = 0; j <= JUDGES - 1; j++) {
            total += score[i][j];
        }
        cout << artist_names[i] << ", " << song_titles[i] << ": " << total << endl;
    }

    for (j = 0; j <= JUDGES - 1; j++) {
        total = 0;
        for (i = 0; i <= ARTISTS - 1; i++) {
            total += score[i][j];
        }
        cout << judge_names[j] << ": " << total / (double)ARTISTS << endl;
    }
    return 0;
}
```

8. Solution

```
#include <iostream>
```

```
#include <cmath>
using namespace std;
const int PEOPLE = 30;
const int MONTHS = 12;

int main() {
    int i, j, sum_heights, sum_weights;
    double average_height, average_weight;

    int weights[PEOPLE][MONTHS];
    int heights[PEOPLE][MONTHS];
    for (i = 0; i <= PEOPLE - 1; i++) {
        for (j = 0; j <= MONTHS - 1; j++) {
            cin >> weights[i][j];
            cin >> heights[i][j];
        }
    }

    for (i = 0; i <= PEOPLE - 1; i++) {
        sum_weights = 0;
        sum_heights = 0;
        for (j = 0; j <= MONTHS - 1; j++) {
            sum_weights += weights[i][j];
            sum_heights += heights[i][j];
        }
        average_weight = sum_weights / (double)MONTHS;
        average_height = sum_heights / (double)MONTHS;
        cout << average_weight << ", " << average_height << endl;
        cout << average_weight * 702 / pow(average_height, 2) << endl;
    }

    for (i = 0; i <= PEOPLE - 1; i++) {
        cout << weights[i][4] * 702 / pow(heights[i][4], 2) << endl;
        cout << weights[i][7] * 702 / pow(heights[i][7], 2) << endl;
    }
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
const double VAT = 0.19;
const int CONSUMERS = 1000;

int main() {
    int consumed, i;
    double payment, total;

    int meter_reading[CONSUMERS][2];
    for (i = 0; i <= CONSUMERS - 1; i++) {
        cin >> meter_reading[i][0];
        cin >> meter_reading[i][1];
```

```
    }

    total = 0;
    for (i = 0; i <= CONSUMERS - 1; i++) {
        consumed = meter_reading[i][1] - meter_reading[i][0];
        cout << consumed << endl;
        payment = consumed * 0.07;
        payment += VAT * payment;
        cout << payment << endl;

        total += consumed;
    }

    cout << total << ", " << total * 0.07 + total * 0.07 * VAT << endl;
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
const int CURRENCIES = 4;
const int DAYS = 5;

int main() {
    int i, j;
    double average, total, usd;

    cout << "Enter an amount in US dollars: ";
    cin >> usd;

    string currency[] = {"British Pounds Sterling", "Euros", "Canadian Dollars", "Australian Dollars"};

    double rate[CURRENCIES][DAYS] = {
        {1.320, 1.321, 1.332, 1.331, 1.341},
        {1.143, 1.156, 1.138, 1.122, 1.129},
        {0.757, 0.764, 0.760, 0.750, 0.749},
        {0.720, 0.725, 0.729, 0.736, 0.739}
    };

    for (i = 0; i <= CURRENCIES - 1; i++) {
        total = 0;
        for (j = 0; j <= DAYS - 1; j++) {
            total += rate[i][j];
        }
        average = total / DAYS;
        cout << usd << " US dollars = " << usd / average << " " << currency[i] << endl;
    }
}
```

11. Solution

```
#include <iostream>
```

```
using namespace std;
const int EMPLOYEES = 10;
const int DAYS = 5;

int main() {
    int i, j;
    double gross_pay, pay_rate, total;

    string days[] = {"Monday", "Tuesday", "Wednesday", "Thursday", "Friday"};

    cin >> pay_rate;

    string names[EMPLOYEES];
    int hours_worked_per_day[EMPLOYEES][DAYS];
    for (i = 0; i <= EMPLOYEES - 1; i++) {
        cin >> names[i];
        for (j = 0; j <= DAYS - 1; j++) {
            cin >> hours_worked_per_day[i][j];
        }
    }

    int hours_worked_per_week[EMPLOYEES];
    for (i = 0; i <= EMPLOYEES - 1; i++) {
        hours_worked_per_week[i] = 0;
        for (j = 0; j <= DAYS - 1; j++) {
            hours_worked_per_week[i] += hours_worked_per_day[i][j];
        }
        if (hours_worked_per_week[i] > 40) {
            cout << names[i] << endl;
        }
    }

    for (i = 0; i <= EMPLOYEES - 1; i++) {
        if (hours_worked_per_week[i] <= 40) {
            gross_pay = pay_rate * hours_worked_per_week[i];
        }
        else {
            gross_pay = pay_rate * 40 + 1.5 * pay_rate * (hours_worked_per_week[i] - 40);
        }
        cout << names[i] << ", " << gross_pay << endl;
    }

    for (i = 0; i <= EMPLOYEES - 1; i++) {
        if (hours_worked_per_week[i] > 40) {
            for (j = 0; j <= DAYS - 1; j++) {
                if (hours_worked_per_day[i][j] > 8) {
                    cout << names[i] << ", " << days[j] << " Overtime!" << endl;
                }
            }
        }
    }

    for (j = 0; j <= DAYS - 1; j++) {
```

```
total = 0;
for (i = 0; i <= EMPLOYEES - 1; i++) {
    if (hours_worked_per_day[i][j] <= 8) {
        gross_pay = pay_rate * hours_worked_per_day[i][j];
    }
    else {
        gross_pay = pay_rate * 8 + 1.5 * pay_rate * (hours_worked_per_day[i][j] - 8);
    }
    total += gross_pay;
}
cout << days[j] << ", " << total << endl;
}
return 0;
}
```

12. Solution

```
#include <iostream>
using namespace std;
const int ROWS = 3;
const int COLUMNS = 4;

int main() {
    int i, j, k;

    int a[ROWS][COLUMNS] = {
        {9, 9, 2, 6},
        {4, 1, 10, 11},
        {12, 15, 7, 3}
    };

    int b[ROWS * COLUMNS];
    k = 0;
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            b[k++] = a[i][j];
        }
    }

    for (k = 0; k <= ROWS * COLUMNS - 1; k++) {
        cout << b[k] << " ";
    }
    return 0;
}
```

13. Solution

```
#include <iostream>
using namespace std;
const int ROWS = 3;
const int COLUMNS = 3;

int main() {
```

```
int i, j, k;

int a[] = {16, 12, 3, 5, 6, 9, 18, 19, 20};

int b[ROWS][COLUMNS];
k = 0;
for (i = ROWS - 1; i >= 0; i--) {
    for (j = 0; j <= COLUMNS - 1; j++) {
        b[i][j] = a[k++];
    }
}

for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS - 1; j++) {
        cout << b[i][j] << "\t";
    }
    cout << endl;
}
return 0;
}
```

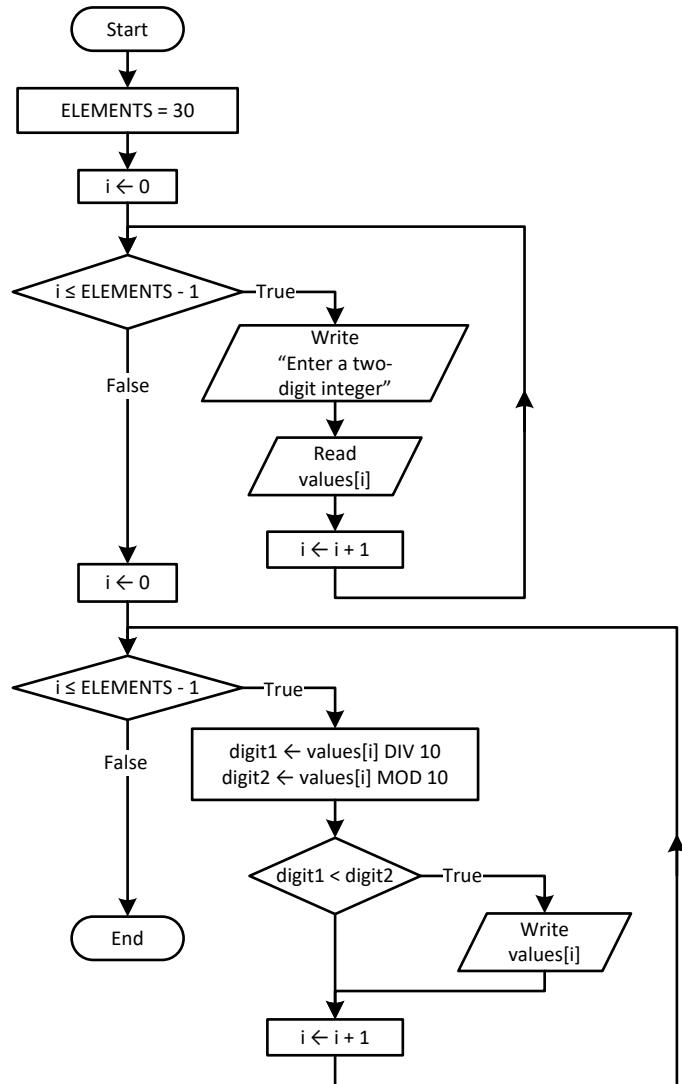
Chapter 34

34.7 Review Questions: True/False

- | | |
|-----------|-----------|
| 1. true | 21. true |
| 2. false | 22. true |
| 3. true | 23. true |
| 4. true | 24. false |
| 5. true | 25. true |
| 6. true | 26. false |
| 7. true | 27. false |
| 8. false | 28. false |
| 9. true | 29. true |
| 10. false | 30. true |
| 11. false | 31. true |
| 12. true | 32. false |
| 13. false | 33. true |
| 14. false | 34. false |
| 15. false | 35. true |
| 16. true | 36. true |
| 17. true | 37. false |
| 18. true | 38. true |
| 19. false | 39. true |
| 20. false | 40. false |

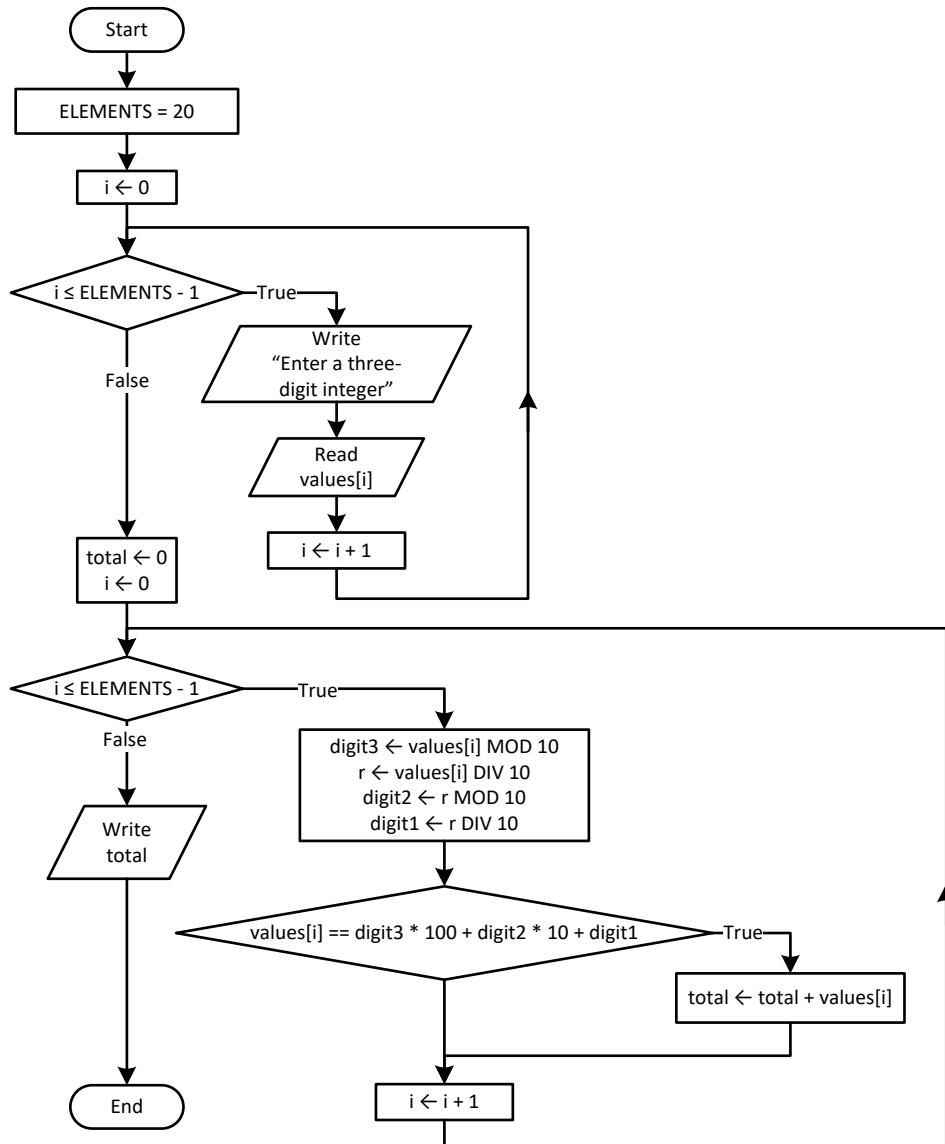
34.8 Review Exercises

1. Solution

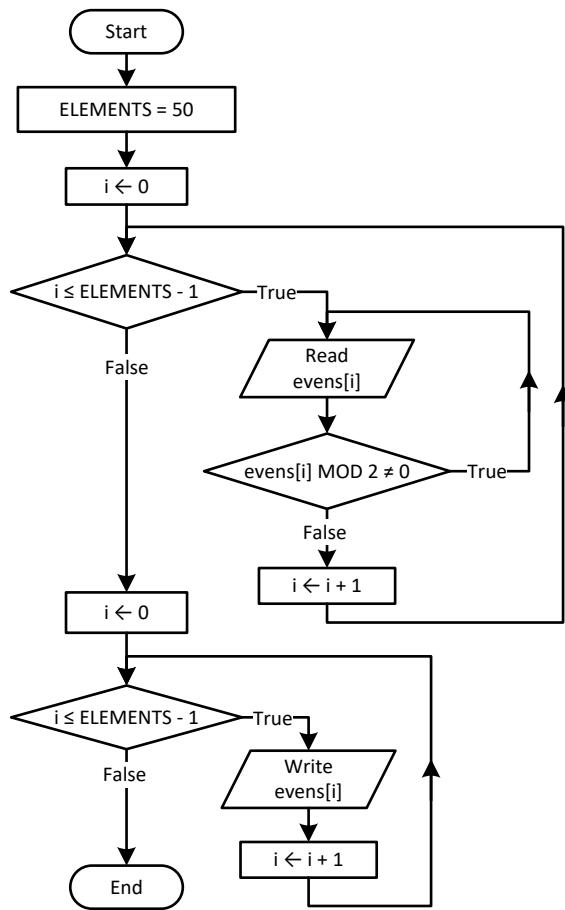


Please note that since flowcharts are a loose method to represent an algorithm, it is not necessary to initialize an array within a flowchart; that is, there is no need to represent the statement `int values[ELEMENTS]`.

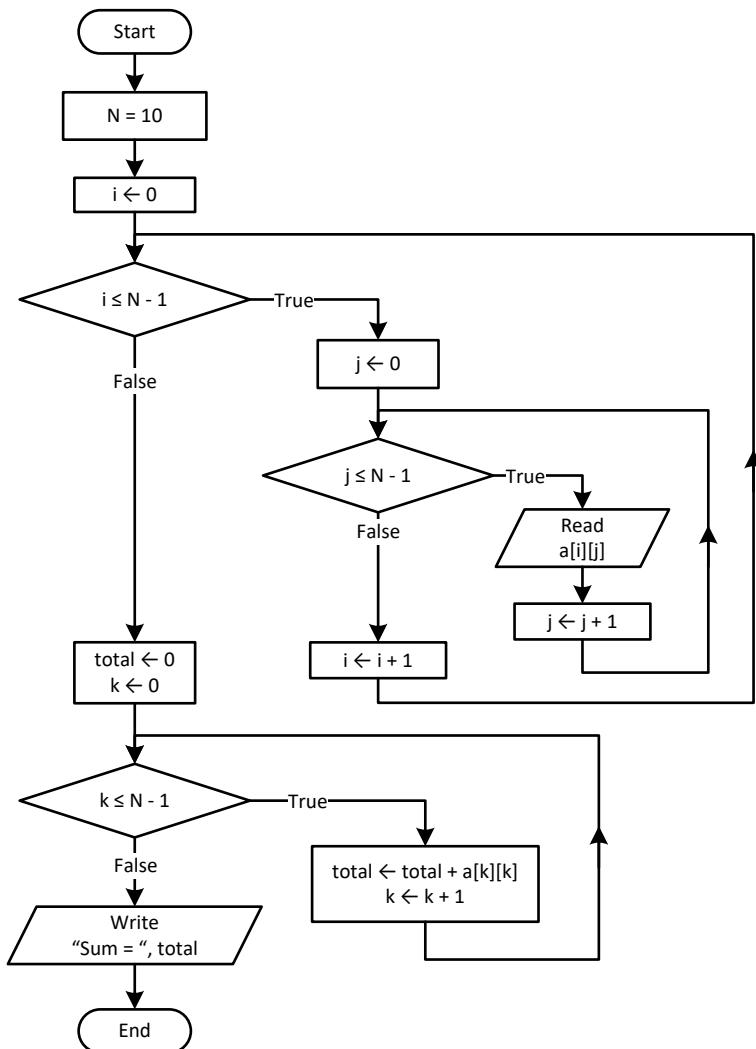
2. Solution



3. Solution



4. Solution



5. Solution

```

for (i = 0; i <= CITIES - 1; i++) {
    do {
        cin >> b[i];
    } while (b[i] >= 0);
}
    
```

6. Solution

```

#include <iostream>
using namespace std;
int main() {
    int i, m, n;
    double b;
    double pos[90];
    double neg[90];
    
```

```
i = 1;
m = 0;
n = 0;
do {
    cin >> b;
    if (b < 0) {
        pos[m] = b;
        m++;
    }
    else {
        neg[n] = b;
        n++;
    }
    i++;
} while (i < 90);
cout << "The End" << endl;
return 0;
}
```

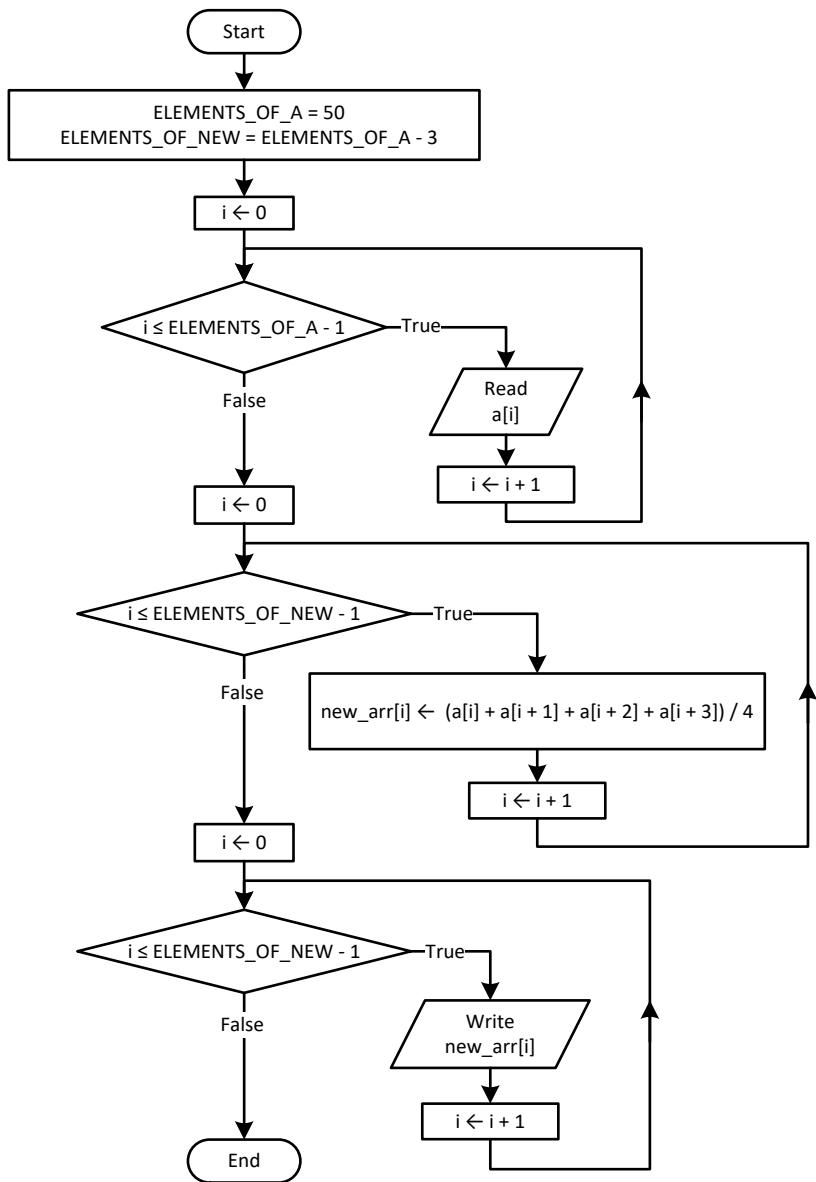
7. Solution

```
max_i = 0;
max_j = 0;
for (i = 0; i <= CITIES - 1; i++) {
    for (j = 0; j <= CITIZENS - 1; j++) {
        if (a[i][j] > a[max_i][max_j]) {
            max_i = i;
            max_j = j;
        }
    }
}
cout << a[max_i][max_j] << endl;
```

8. Solution

```
for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS - 1; j++) {
        cin >> a[i][j];
        while (a[i][j] == 0) {
            cout << "Error" << endl;
            cin >> a[i][j];
        }
    }
}
```

9. Solution



```

#include <iostream>
using namespace std;
const int ELEMENTS_OF_A = 50;
const int ELEMENTS_OF_NEW = ELEMENTS_OF_A - 3;

int main() {
    int i;

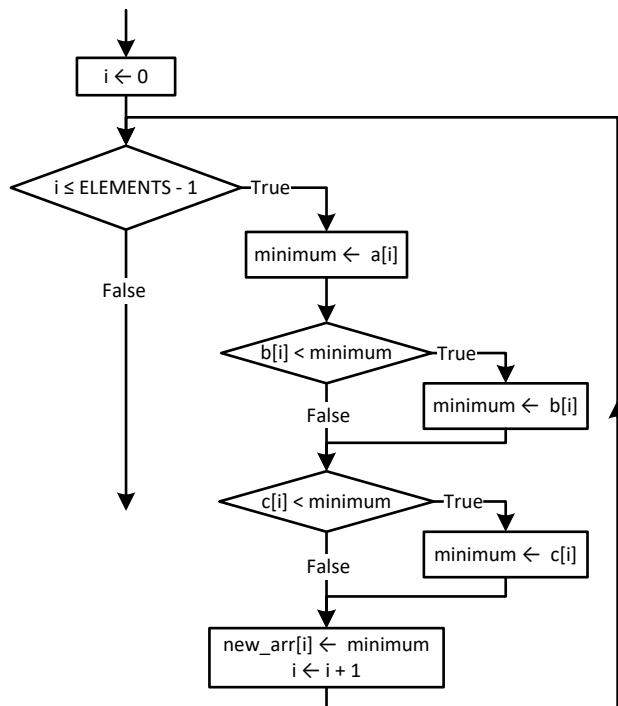
    double a[ELEMENTS_OF_A];
    for (i = 0; i <= ELEMENTS_OF_A - 1; i++) {
        cin >> a[i];
    }

    double new_arr[ELEMENTS_OF_NEW];
    for (i = 0; i <= ELEMENTS_OF_NEW - 1; i++) {
    }
}
  
```

```
    new_arr[i] = (a[i] + a[i + 1] + a[i + 2] + a[i + 3]) / 4;  
}  
  
for (i = 0; i <= ELEMENTS_OF_NEW - 1; i++) {  
    cout << new_arr[i] << "\t" << endl;  
}  
return 0;  
}
```

10. Solution

```
#include <iostream>  
using namespace std;  
const int ELEMENTS = 15;  
  
int main() {  
    int i;  
    double minimum  
  
    double a[ELEMENTS];  
    for (i = 0; i <= ELEMENTS - 1; i++) {  
        cin >> a[i];  
    }  
    double b[ELEMENTS];  
    for (i = 0; i <= ELEMENTS - 1; i++) {  
        cin >> b[i];  
    }  
    double c[ELEMENTS];  
    for (i = 0; i <= ELEMENTS - 1; i++) {  
        cin >> c[i];  
    }  
  
    double new_arr[ELEMENTS];  
    for (i = 0; i <= ELEMENTS - 1; i++) {  
        minimum = a[i];  
        if (b[i] < minimum) {  
            minimum = b[i];  
        }  
        if (c[i] < minimum) {  
            minimum = c[i];  
        }  
        new_arr[i] = minimum;  
    }  
  
for (i = 0; i <= ELEMENTS - 1; i++) {  
    cout << new_arr[i] << endl;  
}  
return 0;  
}
```



11. Solution

```

#include <iostream>
using namespace std;
const int ELEMENTS_OF_A = 10;
const int ELEMENTS_OF_B = 5;
const int ELEMENTS_OF_C = 15;
const int ELEMENTS_OF_NEW = ELEMENTS_OF_A + ELEMENTS_OF_B + ELEMENTS_OF_C;

int main() {
    int i;

    double a[ELEMENTS_OF_A];
    for (i = 0; i <= ELEMENTS_OF_A - 1; i++) {
        cin >> a[i];
    }
    double b[ELEMENTS_OF_B];
    for (i = 0; i <= ELEMENTS_OF_B - 1; i++) {
        cin >> b[i];
    }
    double c[ELEMENTS_OF_C];
    for (i = 0; i <= ELEMENTS_OF_C - 1; i++) {
        cin >> c[i];
    }

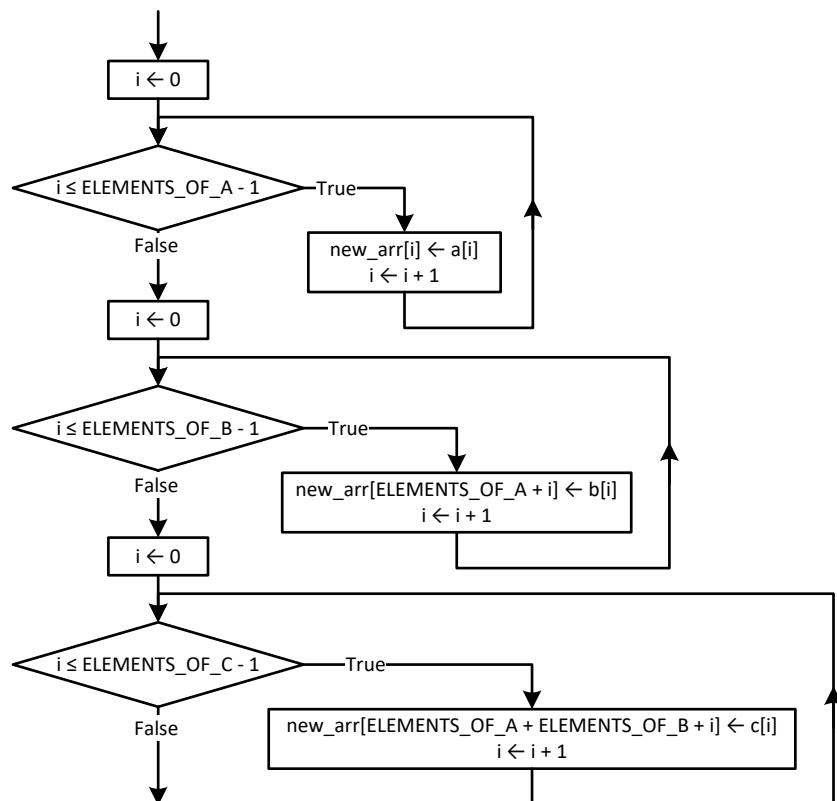
    double new_arr[ELEMENTS_OF_NEW];
    for (i = 0; i <= ELEMENTS_OF_C - 1; i++) {
        new_arr[i] = c[i];
    }
}
  
```

```

for (i = 0; i <= ELEMENTS_OF_B - 1; i++) {
    new_arr[ELEMENTS_OF_C + i] = b[i];
}
for (i = 0; i <= ELEMENTS_OF_A - 1; i++) {
    new_arr[ELEMENTS_OF_B + ELEMENTS_OF_C + i] = a[i];
}

//Display array new
for (i = 0; i <= ELEMENTS_OF_NEW - 1; i++) {
    cout << new_arr[i] << "\t";
}
return 0;
}

```



12. Solution

```

#include <iostream>
using namespace std;
const int COLUMNS_OF_A = 10;
const int COLUMNS_OF_B = 15;
const int COLUMNS_OF_C = 20;
const int ROWS = 5;
const int COLUMNS = COLUMNS_OF_A + COLUMNS_OF_B + COLUMNS_OF_C;

int main() {
    int i, j;

```

```
double a[ROWS][COLUMNS_OF_A];
for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS_OF_A - 1; j++) {
        cin >> a[i][j];
    }
}

double b[ROWS][COLUMNS_OF_B];
for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS_OF_B - 1; j++) {
        cin >> b[i][j];
    }
}

double c[ROWS][COLUMNS_OF_C];
for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS_OF_C - 1; j++) {
        cin >> c[i][j];
    }
}

double new_arr[ROWS][COLUMNS];
for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS_OF_A - 1; j++) {
        new_arr[i][j] = a[i][j];
    }
}
for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS_OF_B - 1; j++) {
        new_arr[i][COLUMNS_OF_A + j] = b[i][j];
    }
}
for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS_OF_C - 1; j++) {
        new_arr[i][COLUMNS_OF_A + COLUMNS_OF_B + j] = c[i][j];
    }
}

for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS - 1; j++) {
        cout << new_arr[i][j] << "\t";
    }
    cout << endl;
}
return 0;
}
```

13. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 50;
```

```

int main() {
    int i, integers_index, reals_index;

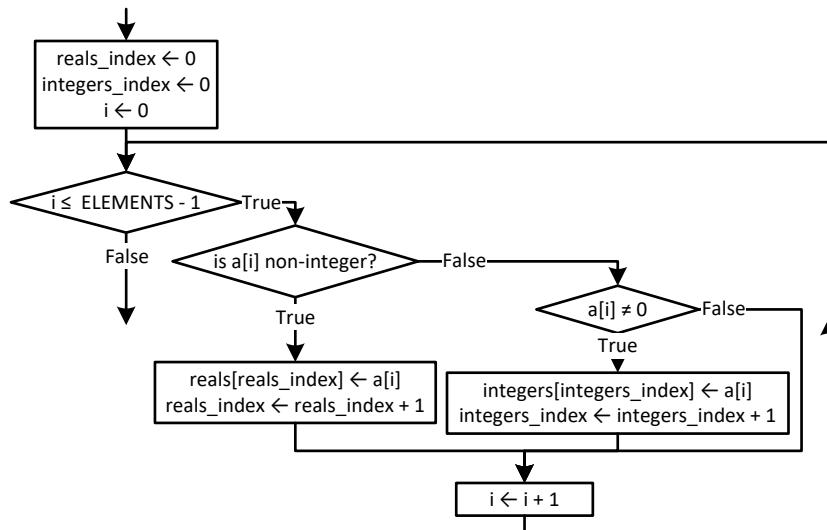
    double a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }

    double reals[ELEMENTS];
    int integers[ELEMENTS];
    reals_index = 0;
    integers_index = 0;
    for (i = 0; i <= ELEMENTS - 1; i++) {
        if (a[i] != (int)(a[i])) {
            reals[reals_index] = a[i];
            reals_index++;
        }
        else if (a[i] != 0) {
            integers[integers_index] = (int)a[i];
            integers_index++;
        }
    }

    for (i = 0; i <= reals_index - 1; i++) {
        cout << reals[i] << "\t";
    }

    cout << endl;
    for (i = 0; i <= integers_index - 1; i++) {
        cout << integers[i] << "\t";
    }
    return 0;
}

```



14. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 50;

int main() {
    int digit1, digit2, digit3, i, k, r;

    int a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }

    int b[ELEMENTS];
    k = 0;
    for (i = 0; i <= ELEMENTS - 1; i++) {
        digit3 = a[i] % 10;
        r = (int)(a[i] / 10);
        digit2 = r % 10;
        digit1 = (int)(r / 10);

        if (digit1 < digit2 && digit2 < digit3) {
            b[k] = a[i];
            k++;
        }
    }

    for (i = 0; i <= k - 1; i++) {
        cout << b[i] << "\t";
    }
    return 0;
}
```

15. Solution

```
#include <iostream>
using namespace std;
const int PRODUCTS = 10;
const int CITIZENS = 200;

int main() {
    int count_B, i, j, maximum;

    string prod_names[PRODUCTS];
    string answers[PRODUCTS][CITIZENS];
    for (i = 0; i <= PRODUCTS - 1; i++) {
        cin >> prod_names[i];
        for (j = 0; j <= CITIZENS - 1; j++) {
            cin >> answers[i][j];
            while (answers[i][j] < "A" || answers[i][j] > "D") {
                cout << "Error! " << endl;
                cin >> answers[i][j];
            }
        }
    }
}
```

```
        }
    }
}

int count_A[PRODUCTS];
for (i = 0; i <= PRODUCTS - 1; i++) {
    count_A[i] = 0;
    for (j = 0; j <= CITIZENS - 1; j++) {
        if (answers[i][j] == "A") {
            count_A[i]++;
        }
    }
    cout << prod_names[i] << ", " << count_A[i] << endl;
}

for (j = 0; j <= CITIZENS - 1; j++) {
    count_B = 0;
    for (i = 0; i <= PRODUCTS - 1; i++) {
        if (answers[i][j] == "B") {
            count_B++;
        }
    }
    cout << count_B << endl;
}

maximum = count_A[0];
for (i = 1; i <= PRODUCTS - 1; i++) {
    if (count_A[i] > maximum) {
        maximum = count_A[i];
    }
}
for (i = 0; i <= PRODUCTS - 1; i++) {
    if (count_A[i] == maximum) {
        cout << prod_names[i] << endl;
    }
}
return 0;
}
```

16. Solution

```
#include <iostream>
using namespace std;
const int US_CITIES = 20;
const int CANADIAN_CITIES = 20;

int main() {
    int i, j, min_j;
    double minimum;

    string us_names[US_CITIES];
    for (i = 0; i <= US_CITIES - 1; i++) {
```

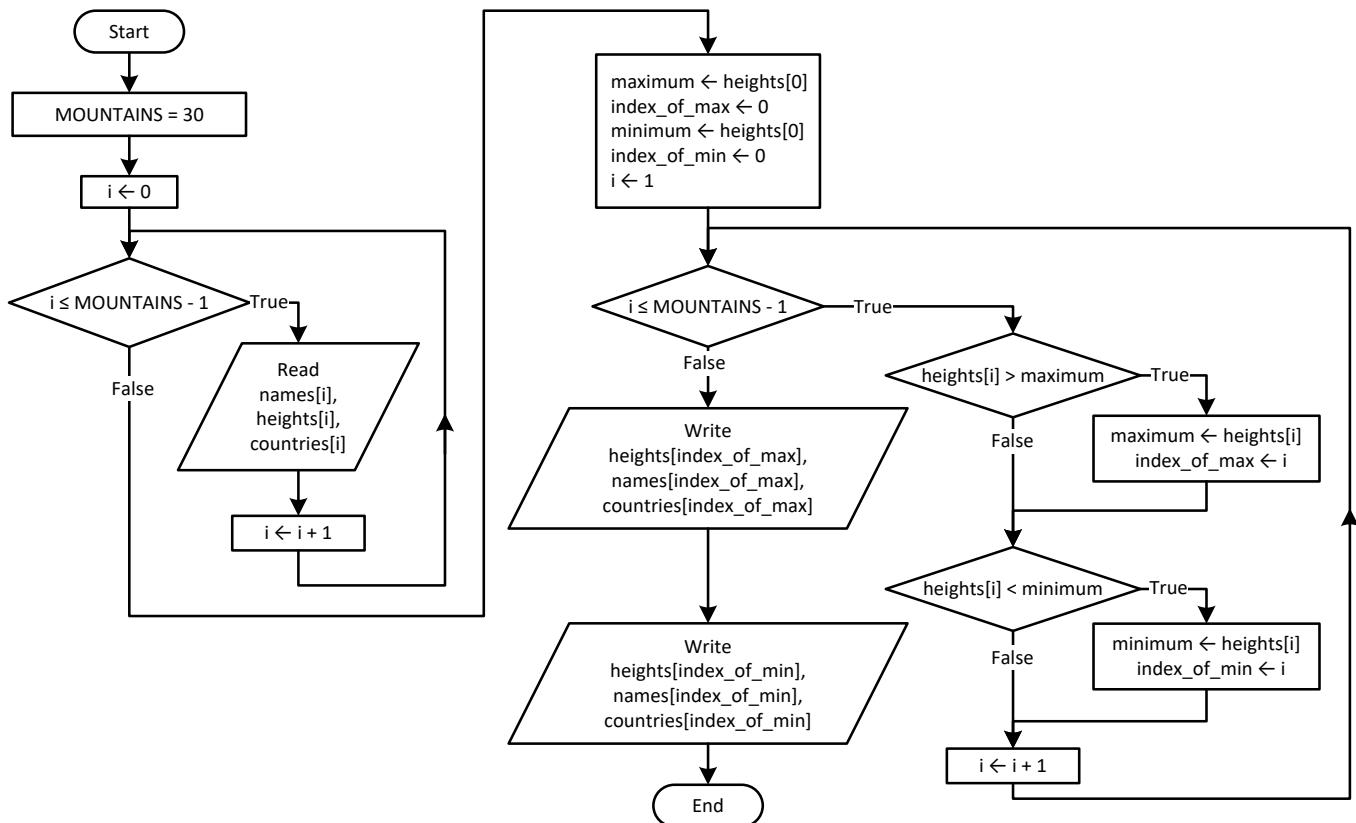
```
cout << "Enter name for US city No " << i + 1 << ":" << endl;
cin >> us_names[i];
}

string canadian_names[CANADIAN_CITIES];
for (j = 0; j <= CANADIAN_CITIES - 1; j++) {
    cout << "Enter name for Canadian city No " << j + 1 << ":" << endl;
    cin >> canadian_names[j];
}

double distances[US_CITIES][CANADIAN_CITIES];
for (i = 0; i <= US_CITIES - 1; i++) {
    for (j = 0; j <= CANADIAN_CITIES - 1; j++) {
        cout << "Enter distance between " << us_names[i] << " and " << canadian_names[j] << ":" << endl;
        cin >> distances[i][j];
    }
}

for (i = 0; i <= US_CITIES - 1; i++) {
    minimum = distances[i][0];
    min_j = 0;
    for (j = 1; j <= CANADIAN_CITIES - 1; j++) {
        if (distances[i][j] < minimum) {
            minimum = distances[i][j];
            min_j = j;
        }
    }
    cout << "Closest Canadian city to " << us_names[i] << " is " << canadian_names[min_j] << endl;
}
return 0;
}
```

17. Solution



```

#include <iostream>
using namespace std;
const int MOUNTAINS = 30;

int main() {
    int i, index_of_max, index_of_min;
    double maximum, minimum;

    string names[MOUNTAINS];
    double heights[MOUNTAINS];
    string countries[MOUNTAINS];
    for (i = 0; i <= MOUNTAINS - 1; i++) {
        cin >> names[i];
        cin >> heights[i];
        cin >> countries[i];
    }

    maximum = heights[0];
    index_of_max = 0;
    minimum = heights[0];
    index_of_min = 0;
    for (i = 1; i <= MOUNTAINS - 1; i++) {
        if (heights[i] > maximum) {
            maximum = heights[i];
            index_of_max = i;
        }
        if (heights[i] < minimum) {
            minimum = heights[i];
            index_of_min = i;
        }
    }
}
  
```

```

    }

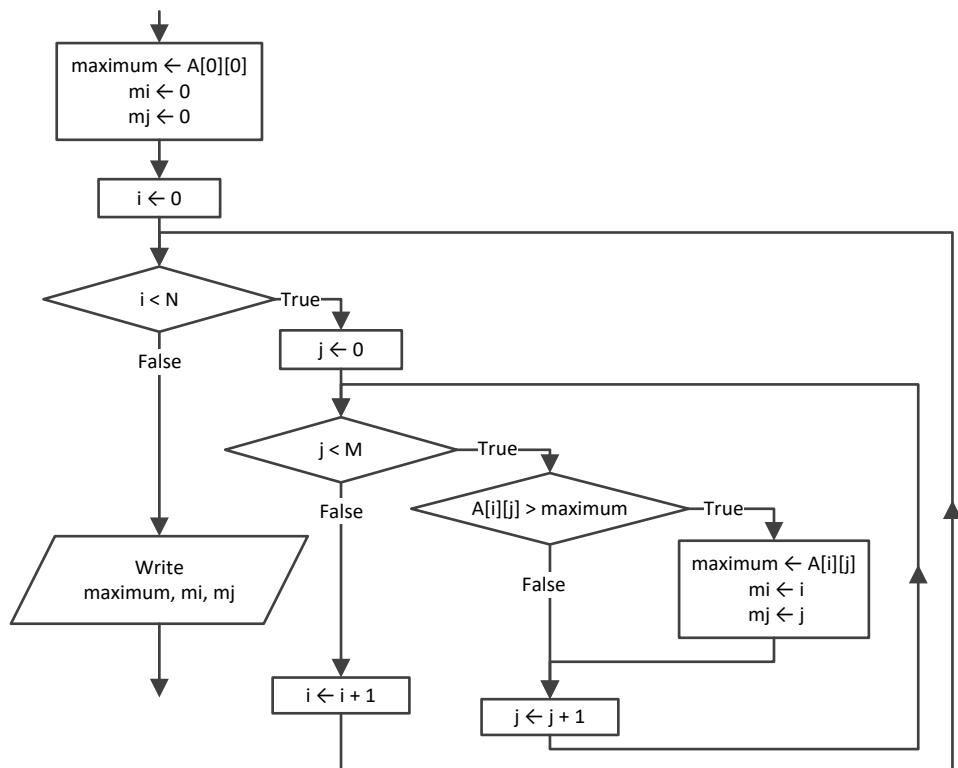
    if (heights[i] < minimum) {
        minimum = heights[i];
        index_of_min = i;
    }
}

cout << heights[index_of_max] << ", " << names[index_of_max] << ", " << countries[index_of_max] << endl;
cout << heights[index_of_min] << ", " << names[index_of_min] << ", " << countries[index_of_min] << endl;

return 0;
}

```

18. Solution



19. Solution

```

#include <iostream>
using namespace std;
const int TEAMS = 26;
const int GAMES = 15;

int main() {
    int i, j, m_i, maximum;

    string names[TEAMS];
    string results[TEAMS][GAMES];
    for (i = 0; i <= TEAMS - 1; i++) {

```

```
    cin >> names[i];
    for (j = 0; j <= GAMES - 1; j++) {
        cin >> results[i][j];
    }
}

int points[TEAMS];
for (i = 0; i <= TEAMS - 1; i++) {
    points[i] = 0;
    for (j = 0; j <= GAMES - 1; j++) {
        if (results[i][j] == "W") {
            points[i] += 3;
        }
        else if (results[i][j] == "T") {
            points[i] += 1;
        }
    }
}

maximum = points[0];
m_i = 0;
for (i = 1; i <= TEAMS - 1; i++) {
    if (points[i] > maximum) {
        maximum = points[i];
        m_i = i;
    }
}

cout << names[m_i] << endl;
return 0;
}
```

20. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
const int OBJECTS = 10;
const int FALLS = 20;

int main() {
    int i, j;
    double maxi, mini;

    double heights[OBJECTS][FALLS];
    double times[OBJECTS][FALLS];
    for (i = 0; i <= OBJECTS - 1; i++) {
        for (j = 0; j <= FALLS - 1; j++) {
            cin >> heights[i][j];
            cin >> times[i][j];
        }
    }
}
```

```

double g[OBJECTS][FALLS];
for (i = 0; i <= OBJECTS - 1; i++) {
    for (j = 0; j <= FALLS - 1; j++) {
        g[i][j] = 2 * heights[i][j] / pow(times[i][j], 2);
    }
}

double minimum[OBJECTS];
double maximum[OBJECTS];
for (i = 0; i <= OBJECTS - 1; i++) {
    minimum[i] = g[i][0];
    maximum[i] = g[i][0];
    for (j = 1; j <= FALLS - 1; j++) {
        if (g[i][j] < minimum[i]) {
            minimum[i] = g[i][j];
        }
        if (g[i][j] > maximum[i]) {
            maximum[i] = g[i][j];
        }
    }
}

for (i = 0; i <= OBJECTS - 1; i++) {
    cout << minimum[i] << ", " << maximum[i] << endl;
}

maxi = maximum[0];
mini = minimum[0];
for (i = 1; i <= OBJECTS - 1; i++) {
    if (maximum[i] > maxi) {
        maxi = maximum[i];
    }
    if (minimum[i] < mini) {
        mini = minimum[i];
    }
}

cout << mini << ", " << maxi << endl;
return 0;
}

```

21. Solution

```

#include <iostream>
using namespace std;
const int STATIONS = 10;
const int DAYS = 365;

int main() {
    int i, j, m_i;
    double minimum;

    string names[STATIONS];

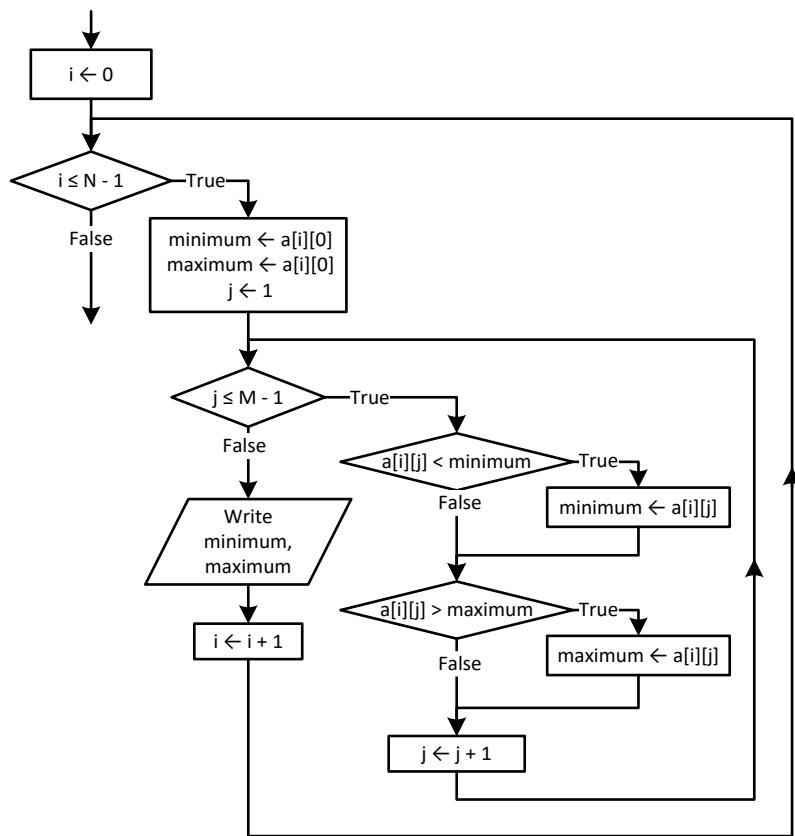
```

```
double co2[STATIONS][DAYS];
for (i = 0; i <= STATIONS - 1; i++) {
    cin >> names[i];
    for (j = 0; j <= DAYS - 1; j++) {
        cin >> co2[i][j];
    }
}

double average[STATIONS];
for (i = 0; i <= STATIONS - 1; i++) {
    average[i] = 0;
    for (j = 0; j <= DAYS - 1; j++) {
        average[i] += co2[i][j];
    }
    average[i] /= DAYS;
}

minimum = average[0];
m_i = 0;
for (i = 1; i <= STATIONS - 1; i++) {
    if (average[i] < minimum) {
        minimum = average[i];
        m_i = i;
    }
}
cout << names[m_i] << endl;
return 0;
}
```

22. Solution



23. Solution

First Approach

```

#include <iostream>
using namespace std;
const int ROWS = 20;
const int COLUMNS = 30;

int main() {
    int i, j;

    double b[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> b[i][j];
        }
    }

    double minimum[COLUMNS];
    double maximum[COLUMNS];
    for (j = 0; j <= COLUMNS - 1; j++) {
        minimum[j] = b[0][j];
        maximum[j] = b[0][j];
    }
  
```

```

for (i = 1; i <= ROWS - 1; i++) {
    if (b[i][j] < minimum[j]) {
        minimum[j] = b[i][j];
    }
    if (b[i][j] > maximum[j]) {
        maximum[j] = b[i][j];
    }
}
}

for (j = 0; j <= COLUMNS - 1; j++) {
    cout << minimum[j] << " " << maximum[j] << endl;
}
return 0;
}

```

Second Approach

```

#include <iostream>
using namespace std;
const int ROWS = 20;
const int COLUMNS = 30;

int main() {
    int i, j;
    double minimum, maximum;

    double b[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> b[i][j];
        }
    }

    for (j = 0; j <= COLUMNS - 1; j++) {
        minimum = b[0][j];
        maximum = b[0][j];
        for (i = 1; i <= ROWS - 1; i++) {
            if (b[i][j] < minimum) {
                minimum = b[i][j];
            }
            if (b[i][j] > maximum) {
                maximum = b[i][j];
            }
        }
        cout << minimum << " " << maximum << endl;
    }
    return 0;
}

```

24. Solution

```

#include <iostream>
using namespace std;

```

```
const int TEAMS = 20;
const int GAMES = 10;

int main() {
    int i, j, m, n, temp;
    bool swaps;
    string temp_str;

    string names[TEAMS];
    string results[TEAMS][GAMES];
    for (i = 0; i <= TEAMS - 1; i++) {
        cout << "Enter team name: ";
        cin >> names[i];
        for (j = 0; j <= GAMES - 1; j++) {
            cout << "Enter result for team " << names[i] << " for game No " << j + 1 << ":" << endl;
            cin >> results[i][j];
            while (results[i][j] != "W" && results[i][j] != "L" && results[i][j] != "T") {
                cout << "Error! Enter only value W, L, or T: ";
                cin >> results[i][j];
            }
        }
    }

    int points[TEAMS];
    for (i = 0; i <= TEAMS - 1; i++) {
        points[i] = 0;
        for (j = 0; j <= GAMES - 1; j++) {
            if (results[i][j] == "W") {
                points[i] += 3;
            }
            else if (results[i][j] == "T") {
                points[i] += 1;
            }
        }
    }

    for (m = 1; m <= TEAMS - 1; m++) {
        swaps = false;
        for (n = TEAMS - 1; n >= m; n--) {
            if (points[n] > points[n - 1]) {
                temp = points[n];
                points[n] = points[n - 1];
                points[n - 1] = temp;

                temp_str = names[n];
                names[n] = names[n - 1];
                names[n - 1] = temp_str;

                swaps = true;
            }
        }
        if (!swaps) break;
    }
}
```

```
    }

    cout << "Gold: " << names[0] << endl;
    cout << "Silver: " << names[1] << endl;
    cout << "Bronze: " << names[2] << endl;
    return 0;
}
```

25. Solution

```
#include <iostream>
using namespace std;
const int PEOPLE = 50;

int main() {
    int i, m, n;
    double temp;
    string temp_str;

    string names[PEOPLE];
    double heights[PEOPLE];
    for (i = 0; i <= PEOPLE - 1; i++) {
        cout << "Enter name for person No. " << i + 1 << ": ";
        cin >> names[i];
        cout << "Enter height for person No. " << i + 1 << ": ";
        cin >> heights[i];
    }

    for (m = 1; m <= PEOPLE - 1; m++) {
        for (n = PEOPLE - 1; n >= m; n--) {
            if (heights[n] > heights[n - 1]) {
                temp = heights[n];
                heights[n] = heights[n - 1];
                heights[n - 1] = temp;

                temp_str = names[n];
                names[n] = names[n - 1];
                names[n - 1] = temp_str;
            }
            else if (heights[n] == heights[n - 1]) {
                if (names[n] < names[n - 1]) {
                    temp_str = names[n];
                    names[n] = names[n - 1];
                    names[n - 1] = temp_str;
                }
            }
        }
    }

    for (i = 0; i <= PEOPLE - 1; i++) {
        cout << heights[i] << "\t" << names[i] << endl;
    }
    return 0;
}
```

```
}
```

26. Solution

```
#include <iostream>
using namespace std;
const int ARTISTS = 12;
const int JUDGES = 10;

int main() {
    int i, j, m, maximum, minimum, n, temp;
    string temp_str;

    string artist_names[ARTISTS];
    int score[ARTISTS][JUDGES];
    for (i = 0; i <= ARTISTS - 1; i++) {
        cout << "Enter name for artist No " << i + 1 << ":" << endl;
        cin >> artist_names[i];
        for (j = 0; j <= JUDGES - 1; j++) {
            cout << "Enter score for artist: " << artist_names[i];
            cout << " gotten from judge No " << j + 1 << ":" << endl;
            cin >> score[i][j];
        }
    }

    int total[ARTISTS];
    for (i = 0; i <= ARTISTS - 1; i++) {
        total[i] = 0;
        for (j = 1; j <= JUDGES - 1; j++) {
            total[i] += score[i][j];
        }
    }

    for (i = 0; i <= ARTISTS - 1; i++) {
        minimum = score[i][0];
        maximum = score[i][0];
        for (j = 1; j <= JUDGES - 1; j++) {
            if (score[i][j] < minimum) {
                minimum = score[i][j];
            }
            if (score[i][j] > maximum) {
                maximum = score[i][j];
            }
        }
        total[i] = total[i] - minimum - maximum;
        cout << total[i] << endl;
    }

    for (m = 1; m <= ARTISTS - 1; m++) {
        for (n = ARTISTS - 1; n >= m; n--) {
            if (total[n] > total[n - 1]) {
                temp = total[n];
```

```

total[n] = total[n - 1];
total[n - 1] = temp;

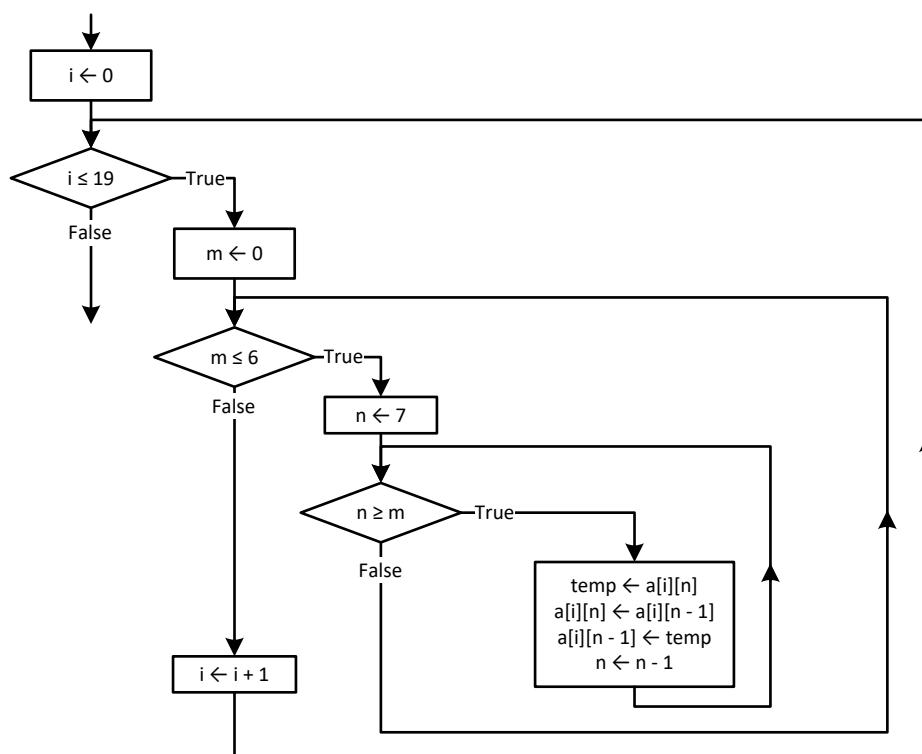
temp_str = artist_names[n];
artist_names[n] = artist_names[n - 1];
artist_names[n - 1] = temp_str;
}

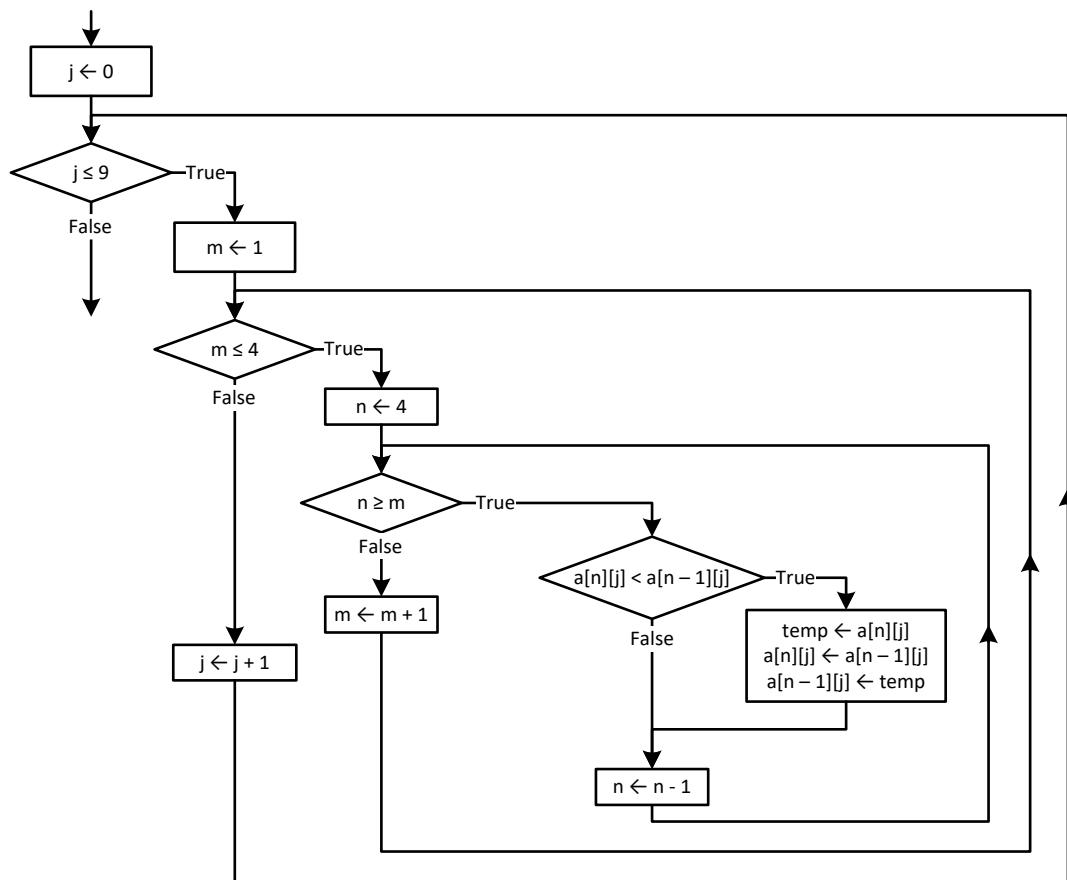
else if (total[n] == total[n - 1]) {
    if (artist_names[n] < artist_names[n - 1]) {
        temp_str = artist_names[n];
        artist_names[n] = artist_names[n - 1];
        artist_names[n - 1] = temp_str;
    }
}
}

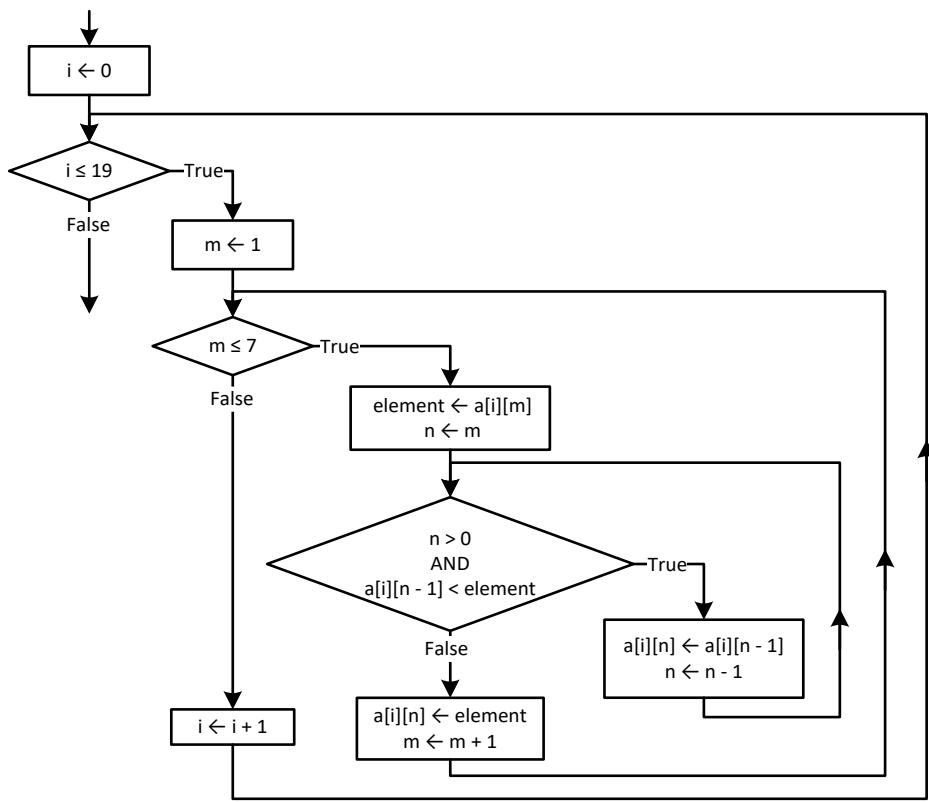
for (i = 0; i <= ARTISTS - 1; i++) {
    cout << artist_names[i] << ", " << total[i] << endl;
}
return 0;
}

```

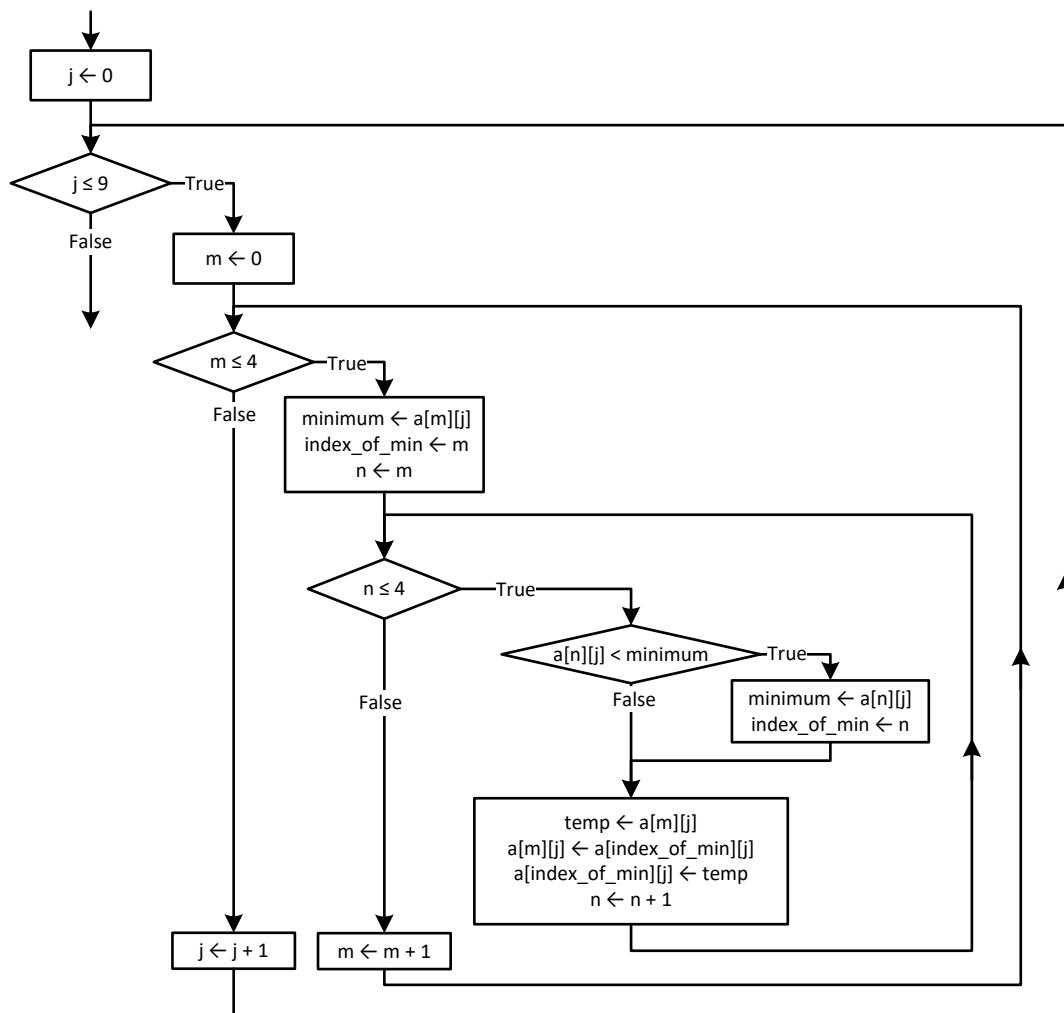
27. Solution



28. Solution

29. Solution

30. Solution



31. Solution

```

#include <iostream>
using namespace std;
const int PEOPLE = 10;
const int PUZZLES = 8;

int main() {
    int i, index_of_min, j, m, n;
    double minimum, temp;
    string temp_str;

    string names[PEOPLE];
    double times[PEOPLE][PUZZLES];
    for (i = 0; i <= PEOPLE - 1; i++) {
        cin >> names[i];
        for (j = 0; j <= PUZZLES - 1; j++) {
            cin >> times[i][j];
        }
    }
}
  
```

```
        }

    }

for (i = 0; i <= PEOPLE - 1; i++) {
    for (m = 0; m <= PUZZLES - 1; m++) {
        minimum = times[i][m];
        index_of_min = m;
        for (n = m; n <= PUZZLES - 1; n++) {
            if (times[i][n] < minimum) {
                minimum = times[i][n];
                index_of_min = n;
            }
        }
        temp = times[i][m];
        times[i][m] = times[i][index_of_min];
        times[i][index_of_min] = temp;
    }
}

for (i = 0; i <= PEOPLE - 1; i++) {
    cout << names[i] << endl;
    for (j = 0; j <= 2; j++) {
        cout << times[i][j] << endl;
    }
}

double average[PEOPLE];
for (i = 0; i <= PEOPLE - 1; i++) {
    average[i] = 0;
    for (j = 0; j <= PUZZLES - 1; j++) {
        average[i] += times[i][j];
    }
    average[i] /= PUZZLES;
}

for (m = 0; m <= PEOPLE - 1; m++) {
    minimum = average[m];
    index_of_min = m;
    for (n = m; n <= PEOPLE - 1; n++) {
        if (average[n] < minimum) {
            minimum = average[n];
            index_of_min = n;
        }
    }
    temp = average[m];
    average[m] = average[index_of_min];
    average[index_of_min] = temp;

    temp_str = names[m];
    names[m] = names[index_of_min];
    names[index_of_min] = temp_str;
}
```

```
    cout << names[0] << ", " << names[1] << ", " << names[2] << endl;
    return 0;
}
```

32. Solution

```
#include <iostream>
using namespace std;
const int AREAS = 5;
const int HOURS = 48;

int main() {
    int i, j, m, m_i, m_j, n;
    double maximum, element_1;
    string element_2;

    string names[AREAS];
    double CO2[AREAS][HOURS];
    for (i = 0; i <= AREAS - 1; i++) {
        cin >> names[i];
        for (j = 0; j <= HOURS - 1; j++) {
            cin >> CO2[i][j];
        }
    }

    double average_per_hour[AREAS];
    for (i = 0; i <= AREAS - 1; i++) {
        average_per_hour[i] = 0;
        for (j = 0; j <= HOURS - 1; j++) {
            average_per_hour[i] += CO2[i][j];
        }
        average_per_hour[i] /= HOURS;
    }

    for (i = 0; i <= AREAS - 1; i++) {
        cout << names[i] << ", " << average_per_hour[i] << endl;
    }

    double average_per_city[HOURS];
    for (j = 0; j <= HOURS - 1; j++) {
        average_per_city[j] = 0;
        for (i = 0; i <= AREAS - 1; i++) {
            average_per_city[j] += CO2[i][j];
        }
        average_per_city[j] /= AREAS;
    }

    for (j = 0; j <= HOURS - 1; j++) {
        cout << average_per_city[j] << endl;
    }

    maximum = average_per_city[0];
```

```
m_j = 0;
for (j = 1; j <= HOURS - 1; j++) {
    if (average_per_city[j] > maximum) {
        maximum = average_per_city[j];
        m_j = j;
    }
}
cout << m_j << endl;

maximum = CO2[0][0];
m_i = 0;
m_j = 0;
for (i = 0; i <= AREAS - 1; i++) {
    for (j = 0; j <= HOURS - 1; j++) {
        if (CO2[i][j] > maximum) {
            maximum = CO2[i][j];
            m_i = i;
            m_j = j;
        }
    }
}
cout << m_j << ", " << names[m_i] << endl;

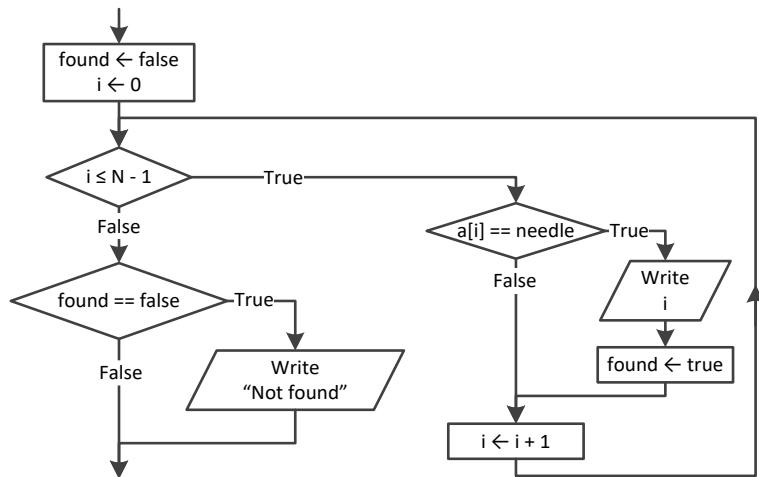
for (m = 1; m <= AREAS - 1; m++) {
    element_1 = average_per_hour[m];
    element_2 = names[m];

    n = m;
    while (n > 0 && average_per_hour[n - 1] < element_1) {
        average_per_hour[n] = average_per_hour[n - 1];
        names[n] = names[n - 1];
        n--;
    }

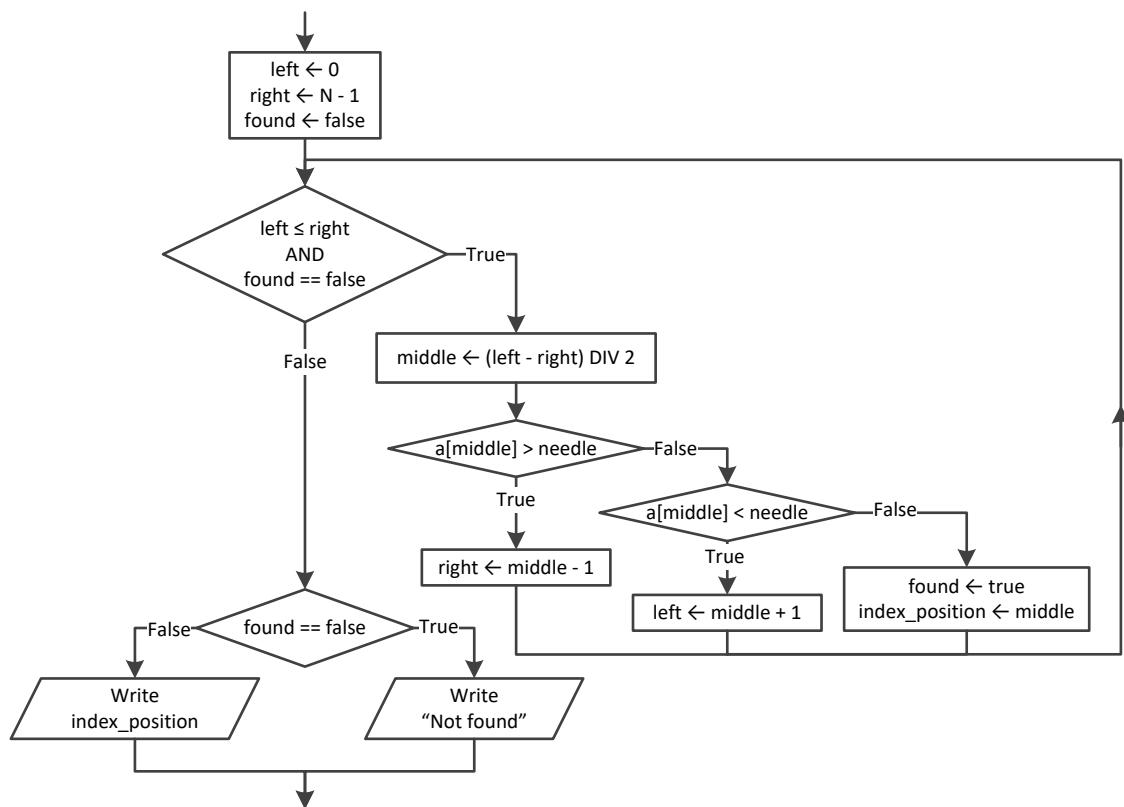
    average_per_hour[n] = element_1;
    names[n] = element_2;
}

cout << names[0] << ", " << names[1] << ", " << names[2] << endl;
return 0;
}
```

33. Solution



34. Solution



35. Solution

```

#include <iostream>
using namespace std;
const int TEAMS = 10;
const int GAMES = 16;
  
```

```
int main() {
    int i, j, total;
    string needle;

    string names[TEAMS];
    int goals_scored[TEAMS][GAMES];
    int goals_let_in[TEAMS][GAMES];
    for (i = 0; i <= TEAMS - 1; i++) {
        cout << "Enter team name: ";
        cin >> names[i];
        for (j = 0; j <= GAMES - 1; j++) {
            cout << "Enter goals scored: ";
            cin >> goals_scored[i][j];
            while (goals_scored[i][j] < 0) {
                cout << "Error! Enter goals scored: ";
                cin >> goals_scored[i][j];
            }
            cout << "Enter goals let in: ";
            cin >> goals_let_in[i][j];
            while (goals_let_in[i][j] < 0) {
                cout << "Error! Enter goals let in: ";
                cin >> goals_let_in[i][j];
            }
        }
    }

    cout << "Enter a team to search: ";
    cin >> needle;

    i = 0;
    while (i < TEAMS - 1 && names[i] != needle) {
        i++;
    }

    if (names[i] != needle) {
        cout << "This team does not exist" << endl;
    }
    else {
        total = 0;
        for (j = 0; j <= GAMES - 1; j++) {
            if (goals_scored[i][j] > goals_let_in[i][j]) {
                total += 3;
            }
            else if (goals_scored[i][j] == goals_let_in[i][j]) {
                total += 1;
            }
        }
        cout << total << endl;
    }
}

return 0;
```

```
}
```

36. Solution

```
#include <iostream>
using namespace std;
const int CLASS1 = 20;
const int CLASS2 = 25;

int main() {
    int i, left, m, middle, n, right;
    string temp, needle;
    bool found;

    cout << "Class 1" << endl;
    string names1[CLASS1];
    for (i = 0; i <= CLASS1 - 1; i++) {
        cout << "Enter name: ";
        cin >> names1[i];
    }
    cout << "Class 2" << endl;
    string names2[CLASS2];
    for (i = 0; i <= CLASS2 - 1; i++) {
        cout << "Enter name: ";
        cin >> names2[i];
    }

    //Bubble sort
    for (m = 1; m <= CLASS1 - 1; m++) {
        for (n = CLASS1 - 1; n >= m; n--) {
            if (names1[n] < names1[n - 1]) {
                temp = names1[n];
                names1[n] = names1[n - 1];
                names1[n] = temp;
            }
        }
    }
    for (m = 1; m <= CLASS2 - 1; m++) {
        for (n = CLASS2 - 1; n >= m; n--) {
            if (names2[n] < names2[n - 1]) {
                temp = names2[n];
                names2[n] = names2[n - 1];
                names2[n] = temp;
            }
        }
    }

    cout << "\nClass 1" << endl;
    for (i = 0; i <= CLASS1 - 1; i++) {
        cout << names1[i] << endl;
    }
    cout << "\nClass 2" << endl;
```

```
for (i = 0; i <= CLASS2 - 1; i++) {
    cout << names2[i] << endl;
}

cout << "Enter a name to search: ";
cin >> needle;

left = 0;
right = CLASS1 - 1;
found = false;
while (left <= right && !found) {
    middle = (int)((left + right) / 2);

    if (names1[middle] > needle) {
        right = middle - 1;
    }
    else if (names1[middle] < needle) {
        left = middle + 1;
    }
    else {
        found = true;
    }
}

if (found) {
    cout << "Student found in Class No 1" << endl;
}
else {
    left = 0;
    right = CLASS2 - 1;
    while (left <= right && !found) {
        middle = (int)((left + right) / 2);

        if (names2[middle] > needle) {
            right = middle - 1;
        }
        else if (names2[middle] < needle) {
            left = middle + 1;
        }
        else {
            found = true;
        }
    }

    if (found) {
        cout << "Student found in Class No 2" << endl;
    }
    else {
        cout << "Student not found in either class" << endl;
    }
}

return 0;
```

```
}
```

37. Solution

```
cout << "Enter username: ";
cin >> usr;
usr = to_upper_copy(usr);
cout << "Enter password: ";
cin >> pwd;
pwd = to_upper_copy(pwd);

i = 0;
while (i < 99 && to_upper_copy(usernames[i]) != usr) {
    i++;
}

if (to_upper_copy(usernames[i]) == usr && to_upper_copy(passwords[i]) == pwd) {
    cout << "Login OK!" << endl;
}
else {
    cout << "Login Failed!" << endl;
}
```

38. Solution

```
cout << "Enter a value to search: ";
cin >> value_str;

found = false;

for (i = 0; i <= 999; i++) {
    if (names[i] == value_str) {
        cout << SSNs[i] << endl;
        found = true;
    }
}

if (!found) {
    value = stoi(value_str);
    i = 0;
    while (i < 999 && SSNs[i] != value) {
        i++;
    }

    if (SSNs[i] == value) {
        found = true;
        cout << names[i] << endl;
    }
}

if (!found) {
    cout << "This value does not exist" << endl;
}
```

39. Solution

```
#include <iostream>
using namespace std;
const int STUDENTS = 12;
const int LESSONS = 6;

int main() {
    int i, j;
    bool found, failure;

    int grades[STUDENTS][LESSONS];
    for (i = 0; i <= STUDENTS - 1; i++) {
        for (j = 0; j <= LESSONS - 1; j++) {
            do {
                cin >> grades[i][j];
                failure = false;
                if (grades[i][j] < 0) {
                    cout << "Error! You entered a negative value" << endl;
                    failure = true;
                }
                else if (grades[i][j] > 100) {
                    cout << "Error! You entered a value grater than 100" << endl;
                    failure = true;
                }
            } while (failure);
        }
    }

    double average[STUDENTS];
    for (i = 0; i <= STUDENTS - 1; i++) {
        average[i] = 0;
        for (j = 0; j <= LESSONS - 1; j++) {
            average[i] += grades[i][j];
        }
        average[i] /= LESSONS;
    }

    found = false;
    for (i = 0; i <= STUDENTS - 1; i++) {
        if (average[i] < 70) {
            found = true;
            break;
        }
    }

    if (found) {
        cout << "There is at least one student that has an average value below 70" << endl;
    }
    return 0;
}
```

40. Solution

```
#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
#include <unordered_map>
using namespace std;
int main() {
    string word, letter;
    int i;

    unordered_map<string, string> morseAlphabet = {
        {"A", ".-"}, {"B", "-..."}, {"C", "-.-."}, {"D", "-.."}, {"E", "."}, {"F", "...-"}, {"G", "--."}, {"H", "...."}, {"I", ".."}, {"J", ".---"}, {"K", "-.-"}, {"L", ".-.."}, {"M", "--"}, {"N", "-."}, {"O", "---"}, {"P", ".--."}, {"Q", "--.-"}, {"R", ".-."}, {"S", "..."}, {"T", "-"}, {"U", "...-"}, {"V", "...-"}, {"W", ".--"}, {"X", "-..-"}, {"Y", "-.--"}, {"Z", "--.."}, {"" ", "/"}
    };

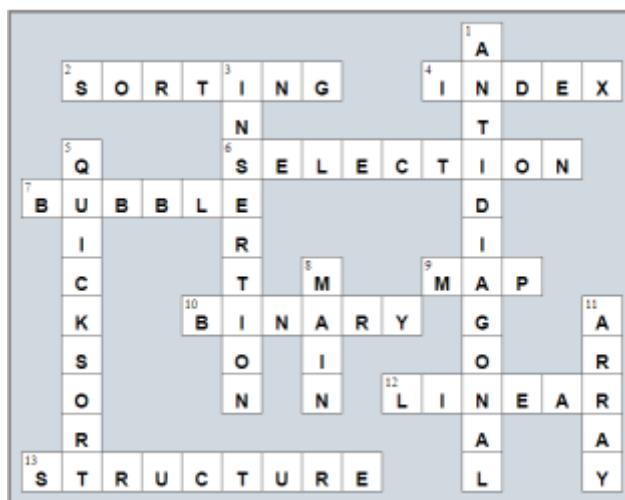
    cout << "Enter a word: ";
    cin >> word;

    for (i = 0; i <= word.length() - 1; i++) {
        letter = word[i];
        cout << morseAlphabet[to_upper_copy(letter)] << " ";
    }
    return 0;
}
```

Review in "Data Structures in C#"

Review Crossword Puzzle

1.



Chapter 35

35.4 Review Questions: True/False

- | | |
|----------|-----------|
| 1. true | 7. true |
| 2. true | 8. false |
| 3. false | 9. true |
| 4. false | 10. true |
| 5. true | 11. false |
| 6. true | 12. true |

Chapter 36

36.8 Review Questions: True/False

- | | |
|-----------|-----------|
| 1. false | 18. true |
| 2. true | 19. false |
| 3. false | 20. true |
| 4. true | 21. true |
| 5. true | 22. true |
| 6. false | 23. true |
| 7. true | 24. false |
| 8. false | 25. true |
| 9. true | 26. false |
| 10. false | 27. true |
| 11. true | 28. false |
| 12. true | 29. true |
| 13. true | 30. true |
| 14. true | 31. true |
| 15. true | 32. true |
| 16. false | 33. false |
| 17. false | |

36.9 Review Exercises

1. Solution

```
int find_max(int a, int b) {
    int maximum;
    if (a > b) {
        maximum = a;
    }
    else {
        maximum = b;
    }
    return maximum;
}
```

2. Solution

Step	Statement	Main Code		Method sum_digits()		
		s	i	a	d1	d2
1	s = 0	0	?			
2	i = 25	0	25			
3	i <= 27	true				
4	s += sum_digits(i)			25	?	?
5	d1 = a % 10			25	5	?

6	d2 = (int)(a / 10)			25	5	2
7	return d1 + d2	7	25			
8	i++	7	26			
9	i <= 27	true				
10	s += sum_digits(i)			26	?	?
11	d1 = a % 10			26	6	?
12	d2 = (int)(a / 10)			26	6	2
13	return d1 + d2	15	26			
14	i++	15	27			
15	i <= 27	true				
16	s += sum_digits(i)			27	?	?
17	d1 = a % 10			27	7	?
18	d2 = (int)(a / 10)			27	7	2
19	return d1 + d2	24	27			
20	i++	24	28			
21	i <= 27	false				
22	cout << s << endl	It displays: 24				

3. Solution

Step	Statement	Main Code		Method sss()		
		s	i	a	total	k
1	i = 1	?	1			
2	s = 0	0	1			
3	while(i < 6)	true				
4	if (i % 2 == 1)	true				
5	s += 1	1	1			
6	i++	1	2			
7	while(i < 6)	true				
8	if (i % 2 == 1)	false				
9	s += sss(i)			2	?	?
10	total = 0			2	0	?
11	k = 1			2	0	1
12	k <= a	true				
13	total += k			2	1	1
14	k++			2	1	2
15	k <= a	true				
16	total += k			2	3	2

17	k++			2	3	3
18	k <= a			false		
19	return total	4	2			
20	i++	4	3			
21	while(i < 6)	true				
22	if (i % 2 == 1)	true				
23	s += 1	5	3			
24	i++	5	4			
25	while(i < 6)	true				
26	if (i % 2 == 1)	false				
27	s += sss(i)			4	?	?
28	total = 0			4	0	?
29	k = 1			4	0	1
30	k <= a			true		
31	total += k			4	1	1
32	k++			4	1	2
33	k <= a			true		
34	total += k			4	3	2
35	k++			4	3	3
36	k <= a			true		
37	total += k			4	6	4
38	k++			4	6	4
39	k <= a			true		
40	total += k			4	10	4
41	k++			4	10	5
42	k <= a			false		
43	return total	15	4			
44	i++	15	5			
45	while(i < 6)	true				
46	if (i % 2 == 1)	true				
47	s += 1	16	5			
48	i++	16	6			
49	while(i < 6)	false				
50	cout << s << endl	It displays: 16				

4. Solution

Step	Statement	Main Code				Method custom_div()	
		k	m	a	x	b	d
1	cin >> k	12	?	?	?		
2	m = 2	12	2	?	?		
3	a = 1	12	2	1	?		
4	while (a < 6)			true			
5	if (k % m != 0)			false			
6	x = a + m + custom_div(m, a)					2	1
7	return (int)((b + d) / 2)	12	2	1	4		
8	cout << m << " " << a << " " << x << endl	It displays: 2 1 4					
9	a += 2	12	2	3	4		
10	m++	12	3	3	4		
11	while (a < 6)			true			
12	if (k % m != 0)			false			
13	x = a + m + custom_div(m, a)					3	3
14	return (int)((b + d) / 2)	12	3	3	9		
15	cout << m << " " << a << " " << x << endl	It displays: 3 3 9					
16	a += 2	12	3	5	9		
17	m++	12	4	5	9		
18	while (a < 6)			true			
19	if (k % m != 0)			false			
20	x = a + m + custom_div(m, a)					4	5
21	return (int)((b + d) / 2)	12	4	5	13		
22	cout << m << " " << a << " " << x << endl	It displays: 4 5 13					
23	a += 2	12	4	7	13		
24	m++	12	5	7	13		
25	while (a < 6)			false			

5. Solution

Step	Statement	Main Code			void Method display()
		i	x	a	
1	i = 1	1	?		
2	i <= 5		true		
3	cin >> x	1	3		

4	display(x)			3
5	if (a % 2 == 0)			false
6	cout << a << " is odd" << endl	It displays: 3 is odd		
7	i++	2	3	
8	i <= 5	true		
9	cin >> x	2	7	
10	display(x)			7
11	if (a % 2 == 0)			false
12	cout << a << " is odd" << endl	It displays: 7 is odd		
13	i++	3	7	
14	i <= 5	true		
15	cin >> x	3	9	
16	display(x)			9
17	if (a % 2 == 0)			false
18	cout << a << " is odd" << endl	It displays: 9 is odd		
19	i++	4	9	
20	i <= 5	true		
21	cin >> x	4	2	
22	display(x)			2
23	if (a % 2 == 0)			true
24	cout << a << " is even" << endl	It displays: 2 is even		
25	i++	5	2	
26	i <= 5	true		
27	cin >> x	5	4	
28	display(x)			4
29	if (a % 2 == 0)			true
30	cout << a << " is even" << endl	It displays: 4 is even		
31	i++	6	4	
32	i <= 5	false		

6. Solution

Step	Statement	Main Code		void Method division()	
		x	y	a	b
1	x = 20	20	?		
2	y = 30	20	30		
3	while (x % y < 30)	true			

4	division(y, x)			30	20
5	b = (int)(b / a)			30	0
6	cout << a * b << endl	It displays: 0			
7	x = 4 * y	120	30		
8	y++	120	31		
9	while (x % y < 30)	true			
10	division(y, x)			31	120
11	b = (int)(b / a)			31	3
12	cout << a * b << endl	It displays: 93			
13	x = 4 * y	124	31		
14	y++	124	32		
15	while (x % y < 30)	true			
16	division(y, x)			32	124
17	b = (int)(b / a)			32	3
18	cout << a * b << endl	It displays: 96			
19	x = 4 * y	128	32		
20	y++	128	33		
21	while (x % y < 30)	true			
22	division(y, x)			33	128
23	b = (int)(b / a)			33	3
24	cout << a * b << endl	It displays: 99			
25	x = 4 * y	132	33		
26	y++	132	34		
27	while (x % y < 30)	false			

7. Solution

Step	Statement	Main Code		void Method calculate()		
		i	m	n	s	j
1	i = 1	1	?			
2	i <= 3	true				
3	cin >> m	1	2			
4	calculate(m)			2	?	?
5	s = 0			2	0	?
6	j = 2			2	0	2
7	j <= 2 * n			true		
8	s = s + pow(j, 2)			2	4	2

9	j += 2			2	4	4
10	j <= 2 * n				true	
11	s = s + pow(j, 2)			2	20	4
12	j += 2			2	20	6
13	j <= 2 * n				false	
14	cout << s << endl	It displays: 20				
15	i++	2	2			
16	i <= 3	true				
17	cin >> m	2	3			
18	calculate(m)			3	?	?
19	s = 0			3	0	?
20	j = 2			3	0	2
21	j <= 2 * n				true	
22	s = s + pow(j, 2)			3	4	2
23	j += 2			3	4	4
24	j <= 2 * n				true	
25	s = s + pow(j, 2)			3	20	4
26	j += 2			3	20	6
27	j <= 2 * n				true	
28	s = s + pow(j, 2)			3	56	6
29	j += 2			3	56	8
30	j <= 2 * n				false	
31	cout << s << endl	It displays: 56				
32	i++	3	3			
33	i <= 3	true				
34	cin >> m	3	4			
35	calculate(m)			4	?	?
36	s = 0			4	0	?
37	j = 2			4	0	2
38	j <= 2 * n				true	
39	s = s + pow(j, 2)			4	4	2
40	j += 2			4	4	4
41	j <= 2 * n				true	
42	s = s + pow(j, 2)			4	20	4
43	j += 2			4	20	6
44	j <= 2 * n				true	
45	s = s + pow(j, 2)			4	56	6

46	j += 2			4	56	8
47	j <= 2 * n				true	
48	s = s + pow(j, 2)			4	120	8
49	j += 2			4	120	10
50	j <= 2 * n				false	
51	cout << s << endl		It displays: 120			
52	i++	4	4			
53	i <= 3		false			

8. Solution

```
double find_sum(double a, double b, double c) {
    return a + b + c;
}
```

9. Solution

```
double find_avg(double a, double b, double c, double d) {
    return (a + b + c + d) / 4;
}
```

10. Solution

```
double maximum(double a, double b, double c) {
    double m;

    m = a;
    if (b > m) {
        m = b;
    }
    if (c > m) {
        m = c;
    }
    return m;
}
```

11. Solution

```
void display_max(double a, double b, double c, double d, double e) {
    double m;

    m = a;
    if (b > m) {
        m = b;
    }
    if (c > m) {
        m = c;
    }
    if (d > m) {
```

```
    m = d;
}
if (e > m) {
    m = e;
}
cout << m << endl;
}
```

12. Solution

```
double my_round(double x) {
    int digit_to_check;
    double return_value;

    digit_to_check = (int)(x * 1000) % 10;
    if (digit_to_check >= 5) {
        return_value = ((int)(x * 100) + 1) / 100.0;
    }
    else {
        return_value = ((int)(x * 100)) / 100.0;
    }

    return return_value;
}
```

13. Solution

```
#include <iostream>
using namespace std;

double find_min(double a, double b) {
    double minimum;

    minimum = a;
    if (b < minimum) {
        minimum = b;
    }
    return minimum;
}

int main() {
    double temp1, temp2, x1, x2, x3, x4;

    cout << "Enter four numbers: ";
    cin >> x1 >> x2 >> x3 >> x4;

    //First approach
    temp1 = find_min(x1, x2);
    temp2 = find_min(x3, x4);
    cout << find_min(temp1, temp2) << endl;

    //Second approach
    cout << find_min(find_min(x1, x2), find_min(x3, x4)) << endl;
}
```

```
    return 0;
}
```

14. Solution

```
#include <iostream>
using namespace std;

double Kelvin_to_Fahrenheit(double kelvin) {
    return 1.8 * kelvin - 459.67;
}

double Kelvin_to_Celsius(double kelvin) {
    return kelvin - 273.15;
}

int main() {
    double k;

    cout << "Enter a temperature in degrees Kelvin: ";
    cin >> k;
    cout << "Fahrenheit: " << Kelvin_to_Fahrenheit(k) << endl;
    cout << "Celsius: " << Kelvin_to_Celsius(k) << endl;
    return 0;
}
```

15. Solution

```
#include <iostream>
#include <cmath>
using namespace std;

string bmi(double w, double h) {
    double b;
    string return_value;

    b = w * 703 / pow(h, 2);
    if (b < 16) {
        return_value = "You must add weight.";
    }
    else if (b < 18.5) {
        return_value = "You should add some weight.";
    }
    else if (b < 25) {
        return_value = "Maintain your weight.";
    }
    else if (b < 30) {
        return_value = "You should lose some weight.";
    }
    else {
        return_value = "You must lose weight.";
    }
}
```

```
    return return_value;
}

int main() {
    double height, weight;
    int age;

    cout << "Enter your weight (in pounds): ";
    cin >> weight;
    while (weight < 0) {
        cout << "Error! Enter your weight (in pounds): ";
        cin >> weight;
    }

    cout << "Enter your age: " << endl;
    cin >> age;
    while (age < 18) {
        cout << "Error! Enter your age: ";
        cin >> age;
    }

    cout << "Enter your height (in inches): " << endl;
    cin >> height;
    while (height < 0) {
        cout << "Error! Enter your height (in inches): " << endl;
        cin >> height;
    }

    cout << bmi(weight, height) << endl;
    return 0;
}
```

16. Solution

```
#include <iostream>
using namespace std;

void num_of_days(int year, int month) {
    int days;

    switch (month) {
        case 4:
        case 6:
        case 9:
        case 11:
            days = 30;
            break;
        case 2:
            if (year % 4 == 0 && year % 100 != 0 || year % 400 == 0) {
                days = 29;
            }
            else {
                days = 28;
            }
    }
}
```

```
        }
        break;
    default:
        days = 31;
    }

    cout << days << endl;
}

int main() {
    int m, y;

    cout << "Enter a year: ";
    cin >> y;
    for (m = 1; m <= 12; m++) {
        num_of_days(y, m);
    }
    return 0;
}
```

17. Solution

```
#include <iostream>
using namespace std;

void display_menu() {
    cout << endl;
    cout << "1. Convert meters to miles" << endl;
    cout << "2. Convert miles to meters" << endl;
    cout << "3. Exit" << endl;
    cout << "Enter a choice: ";
}

void meters_to_miles(double meters) {
    cout << meters << " meters equals " << meters / 1609.344 << " miles" << endl;
}

void miles_to_meters(double miles) {
    cout << miles << " miles equals " << miles * 1609.344 << " meters" << endl;
}

int main() {
    int choice;
    double distance;

    display_menu();
    cin >> choice;
    while (choice != 3) {
        cout << "Enter distance: " << endl;
        cin >> distance;
        if (choice == 1) {
            meters_to_miles(distance);
        }
    }
}
```

```
    else {
        miles_to_meters(distance);
    }

    display_menu();
    cin >> choice;
}
return 0;
}
```

18. Solution

```
#include <iostream>
using namespace std;

void amount_to_pay(int seconds) {
    double extra, tax, total, total_without_tax;

    if (seconds <= 600) {
        extra = 0;
    }
    else if (seconds <= 1200) {
        extra = (seconds - 600) * 0.01;
    }
    else {
        extra = 600 * 0.01 + (seconds - 1200) * 0.02;
    }

    total_without_tax = 10 + extra;
    tax = total_without_tax * 11 / 100;
    total = total_without_tax + tax;

    cout << "Total amount to pay: " << total << endl;
}

int main() {
    int seconds;

    cout << "Enter number of seconds: ";
    cin >> seconds;
    amount_to_pay(seconds);
}
```

Chapter 37

37.10 Review Questions: True/False

- | | |
|-----------|-----------|
| 1. true | 14. false |
| 2. true | 15. true |
| 3. true | 16. true |
| 4. false | 17. false |
| 5. true | 18. false |
| 6. false | 19. false |
| 7. true | 20. false |
| 8. false | 21. true |
| 9. true | 22. true |
| 10. false | 23. false |
| 11. true | 24. true |
| 12. true | 25. true |
| 13. true | 26. true |

37.11 Review Exercises

1. *Solution*

It displays: 5

2. *Solution*

It displays: 14

3. *Solution*

It displays: 14

4. *Solution*

It displays: hellohellohello

5. *Solution*

It displays: 15

6. *Solution*

It displays: 11 4

7. *Solution*

```
#include <iostream>
using namespace std;

const int STUDENTS = 10;
const int LESSONS = 5;
```

```
void part1(string names[], int grades[][LESSONS]) {
    int i, j;

    for (i = 0; i <= STUDENTS - 1; i++) {
        cout << "Enter name for student No. " << i + 1 << ": ";
        cin >> names[i];
        for (j = 0; j <= LESSONS - 1; j++) {
            cout << "Enter grade for lesson No. " << j + 1 << ": ";
            cin >> grades[i][j];
        }
    }
}

double *part2(int grades[][LESSONS]) {
    static double average[STUDENTS];
    int i, j;

    for (i = 0; i <= STUDENTS - 1; i++) {
        average[i] = 0;
        for (j = 0; j <= LESSONS - 1; j++) {
            average[i] += grades[i][j];
        }
        average[i] /= LESSONS;
    }
    return average;
}

void part3(double average[], string names[]) {
    int m, n;
    double temp;
    string temp_str;

    for (m = 1; m <= STUDENTS - 1; m++) {
        for (n = STUDENTS - 1; n >= m; n--) {
            if (average[n] > average[n - 1]) {
                temp = average[n];
                average[n] = average[n - 1];
                average[n - 1] = temp;

                temp_str = names[n];
                names[n] = names[n - 1];
                names[n - 1] = temp_str;
            }
            else if (average[n] == average[n - 1]) {
                if (names[n] < names[n - 1]) {
                    temp_str = names[n];
                    names[n] = names[n - 1];
                    names[n - 1] = temp_str;
                }
            }
        }
    }
}
```

```
}
```

```
int main() {
    int i;
```

```
    string names[STUDENTS];
    int grades[STUDENTS][LESSONS];
```

```
    part1(names, grades);
```

```
    double *average = part2(grades);
```

```
    part3(average, names);
```

```
    for (i = 0; i <= STUDENTS - 1; i++) {
        cout << names[i] << "\t" << average[i] << endl;
    }
    return 0;
}
```

8. Solution

```
#include <iostream>
#include <string>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;

string part1() {
    string message;

    cout << "Enter a message: ";
    getline(cin, message);
    message = to_lower_copy(message);
    return message;
}

string part2(string message) {
    string letter, message_clean;
    int i;

    message_clean = "";
    for (i = 0; i <= message.length() - 1; i++) {
        letter = message[i];
        if (letter != " " && letter != "," && letter != "." && letter != "?") {
            message_clean += letter;
        }
    }
    return message_clean;
}

bool part3(string message_clean) {
    int middle_pos, i, j;
    bool palindrome;
```

```
string left_letter, right_letter;

middle_pos = (int)((message_clean.length() - 1) / 2);
j = message_clean.length() - 1;
palindrome = true;
for (i = 0; i <= middle_pos; i++) {
    left_letter = message_clean[i];
    right_letter = message_clean[j];
    if (left_letter != right_letter) {
        palindrome = false;
        break;
    }
    j--;
}
return palindrome;
}

bool part4(string message) {
    string message_clean;
    bool palindrome;

    message_clean = part2(message);
    palindrome = part3(message_clean);
    return palindrome;
}

int main() {
    string message;
    bool palindrome;

    message = part1();
    palindrome = part4(message);
    if (palindrome) {
        cout << "The message is palindrome" << endl;
    }
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, b, c, maximum;

    cin >> a >> b >> c >> d;

    maximum = a;
    if (b > maximum) {
        maximum = b;
    }
    if (c > maximum) {
        maximum = c;
```

```

    }
    if (d > maximum) {
        maximum = d;
    }

    cout << maximum << endl;
    return 0;
}

```

10. Solution

```

void f1(double a, double b, double c, double returning_array[]) {
    returning_array[0] = a + b + c;
    returning_array[1] = returning_array[0] / 3;
}

```

11. Solution

```

double my_round(double x, int decimal_places = 2) {
    double return_value;

    int digit_to_check = (int)((x * pow(10, decimal_places + 1))) % 10;
    if (digit_to_check >= 5) {
        return_value = ((int)((x * pow(10, decimal_places))) + 1) / pow(10, decimal_places);
    }
    else {
        return_value = ((int)(x * pow(10, decimal_places))) / pow(10, decimal_places);
    }
    return return_value;
}

```

12. Solution

```

#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;

bool get_input() {
    string answer;

    do {
        cout << "Enter Yes or No: ";
        cin >> answer;
        answer = to_upper_copy(answer);
    } while (answer != "YES" && answer != "NO");

    return answer == "YES"; //This returns true or false
}

double find_area(double b, double h) {
    return b * h;
}

```

```
int main() {
    double b, h;

    do {
        cout << "Enter the base of the parallelogram: ";
        cin >> b;
        cout << "Enter the height of the parallelogram: ";
        cin >> h;

        cout << "Area = " << find_area(b, h) << endl;

        cout << "Would you like to repeat? " << endl;
    } while (get_input());
    return 0;
}
```

13. Solution

```
#include <iostream>
using namespace std;

const int STUDENTS = 100;

void get_arrays(string names[],  int grades[]) {
    int i;

    for (i = 0; i <= STUDENTS - 1; i++) {
        cout << "Enter name: ";
        cin >> names[i];
        cout << "Enter grade: ";
        cin >> grades[i];
    }
}

double get_average(int grades[]) {
    int i, total = 0;
    for (i = 0; i <= STUDENTS - 1; i++) {
        total += grades[i];
    }
    return total / (double)STUDENTS;
}

void sort_arrays(int grades[], string names[]) {
    int m, n, element_grds;
    string element_nms;

    for (m = 1; m <= STUDENTS - 1; m++) {
        element_grds = grades[m];
        element_nms = names[m];

        n = m;
        while (n > 0 && grades[n - 1] > element_grds) {
            grades[n] = grades[n - 1];
```

```
    names[n] = names[n - 1];
    n--;
}

grades[n] = element_grds;
names[n] = element_nms;
}
}

int main() {
    int i;
    double average;

    string names[STUDENTS];
    int grades[STUDENTS];

    get_arrays(names, grades);
    average = get_average(grades);
    sort_arrays(grades, names);
    for (i = 0; i <= STUDENTS - 1; i++) {
        if (grades[i] < average) {
            cout << names[i] << endl;
        }
    }
    return 0;
}
```

14. Solution

```
#include <iostream>
using namespace std;

const int JUDGES = 10;

int get_array() {
    int score[JUDGES];
    int i;

    for (i = 0; i <= JUDGES - 1; i++) {
        cout << "Judge No " << i + 1 << ". Enter score: ";
        cin >> score[i];
    }
    return score;
}

void find_min_max(int score[], int &minimum, int &maximum) {
    int i;
    minimum = score[0];
    maximum = score[0];
    for (i = 1; i <= JUDGES - 1; i++) {
        if (score[i] > maximum) {
            maximum = score[i];
        }
    }
}
```

```
    if (score[i] < minimum) {
        minimum = score[i];
    }
}

int main() {
    string name;
    int total, i, points, minimum = 0, maximum = 0;

    cout << "Enter artist's name: ";
    cin >> name;
    int score[] = get_array();
    find_min_max(score, minimum, maximum);

    total = 0;
    for (i = 0; i <= JUDGES - 1; i++) {
        total += score[i];
    }

    points = total - minimum - maximum;
    cout << "Artist " << name << " got " << points << " points" << endl;
    return 0;
}
```

15. Solution

```
#include <iostream>
using namespace std;

double woc(int index) {
    double return_value;

    if (index == 1) {
        return_value = 1;
    }
    else {
        return_value = 2 * woc(index - 1);
    }
    return return_value;
}

int main() {
    double total;
    int i;

    total = 0;
    for (i = 1; i <= 64; i++) {
        total += woc(i);
    }
    cout << total << endl;
    return 0;
}
```

16. Solution

```
#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
using namespace std;

double factorial(int value) {
    double return_value;

    if (value == 1) {
        return_value = 1;
    }
    else {
        return_value = value * factorial(value - 1);
    }

    return return_value;
}

double my_cos(double x, int i = 40) {
    double return_value;

    if (i == 0) {
        return_value = 1;
    }
    else {
        return_value = my_cos(x, i - 4) + pow(x, i) / factorial(i) - pow(x, i - 2) / factorial(i - 2);
    }

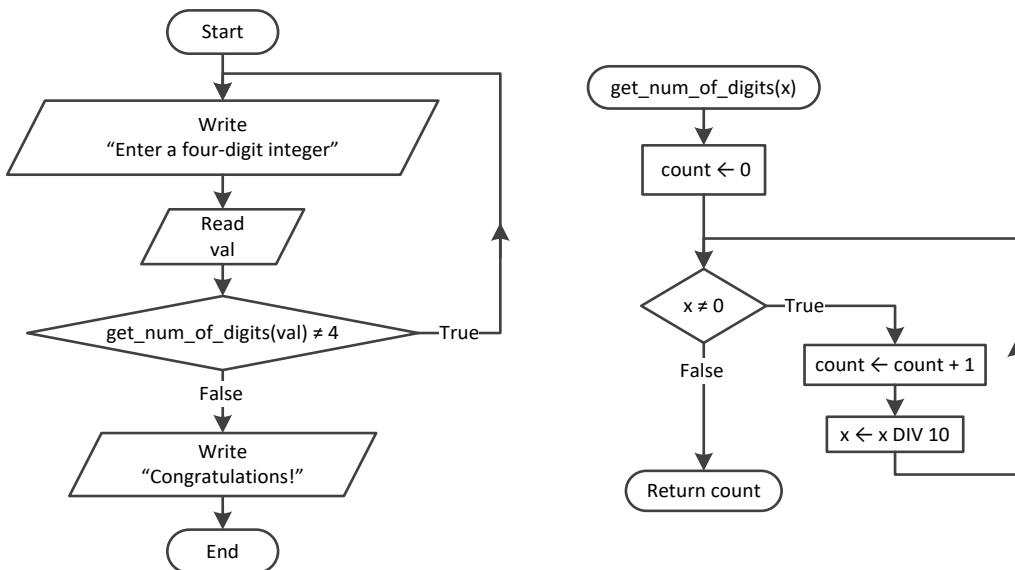
    return return_value;
}

int main() {
    cout << my_cos(M_PI / 4) << endl;
}
```

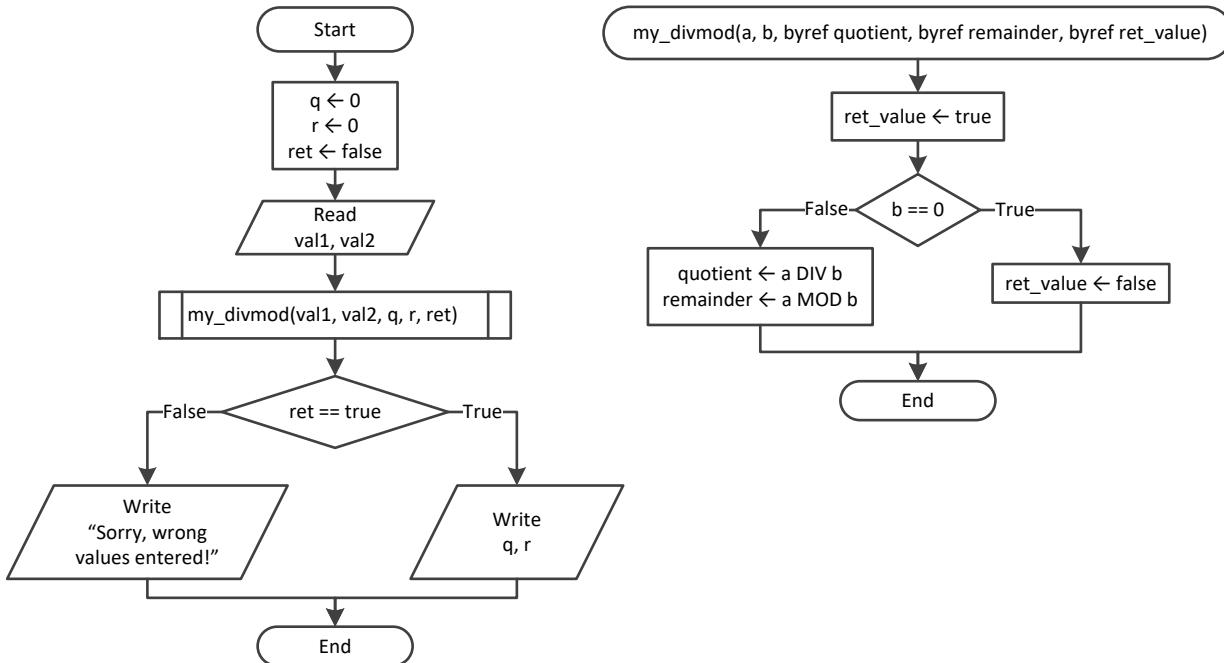
Chapter 38

38.3 Review Exercises

1. Solution



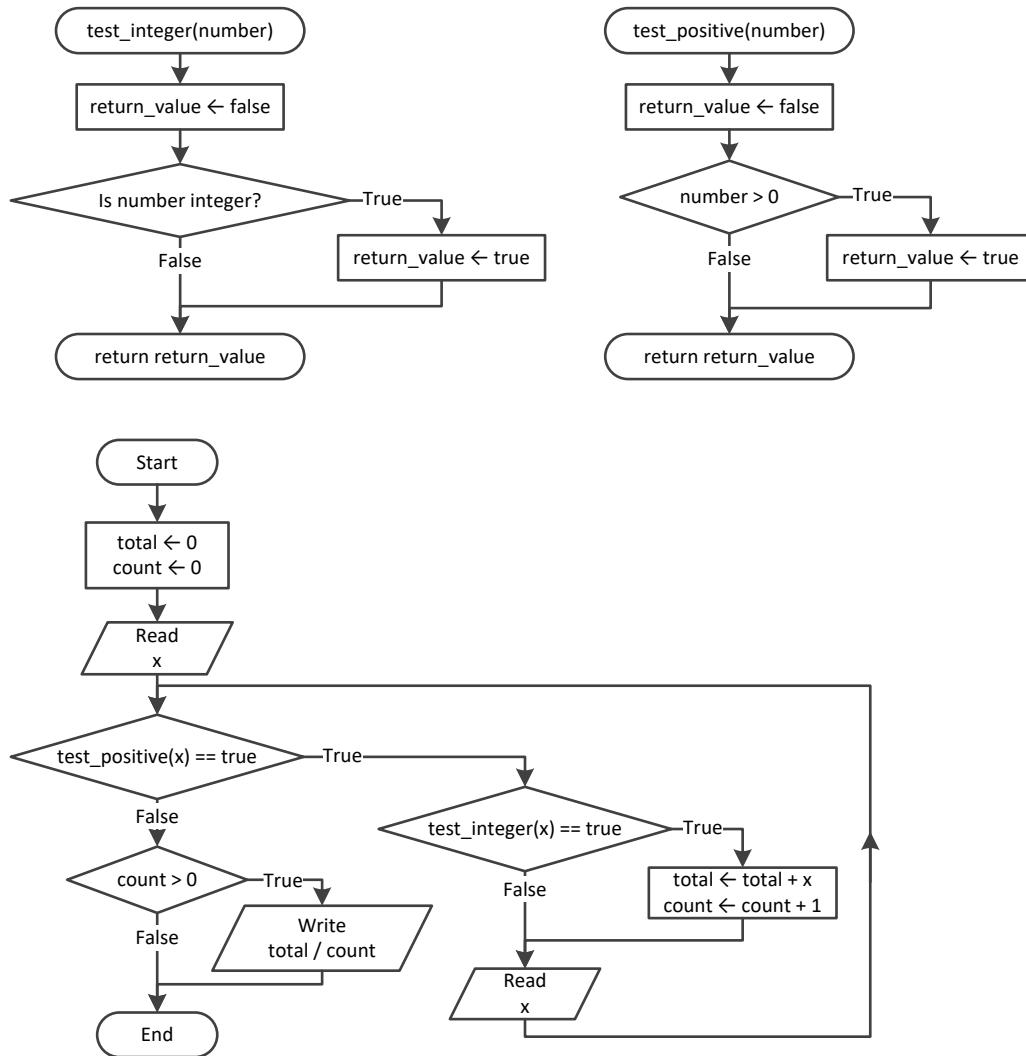
2. Solution



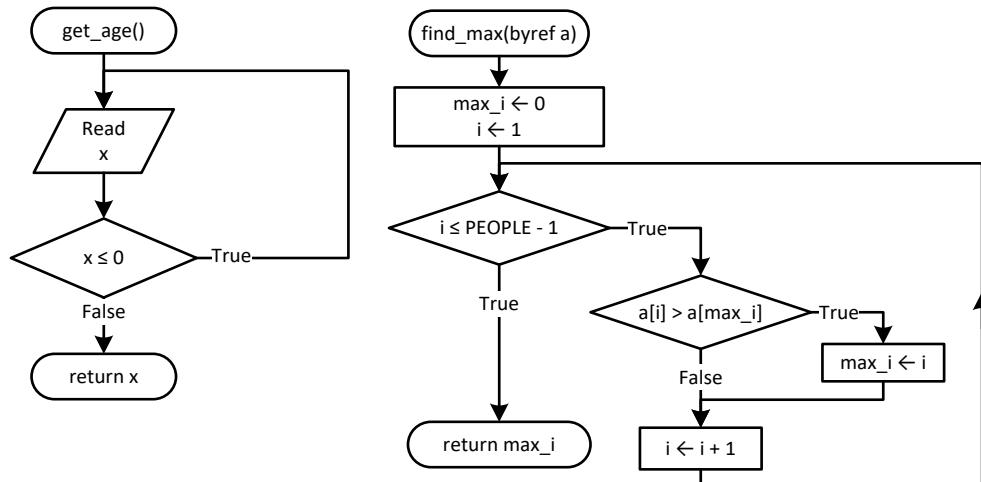
Flowcharts are a loose method of representing an algorithm. Thus, you can represent a pass by reference using the keyword `byref`, which clearly denotes what it actually does.

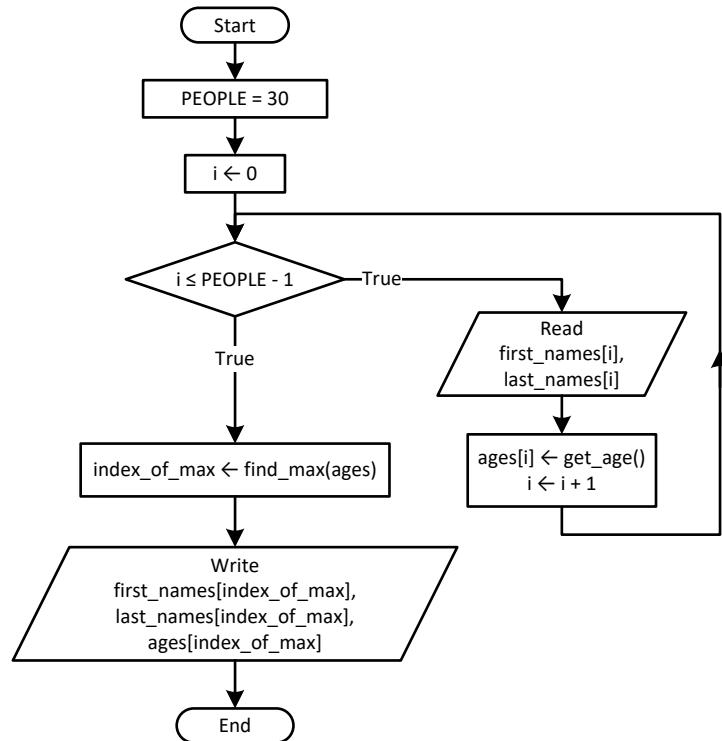
Some programmers, instead of using the keyword `byref`, prefer to write the keyword `inout`, which denotes pretty much the same thing—that the variable is both input (it accepts values) and output (it returns values).

3. Solution

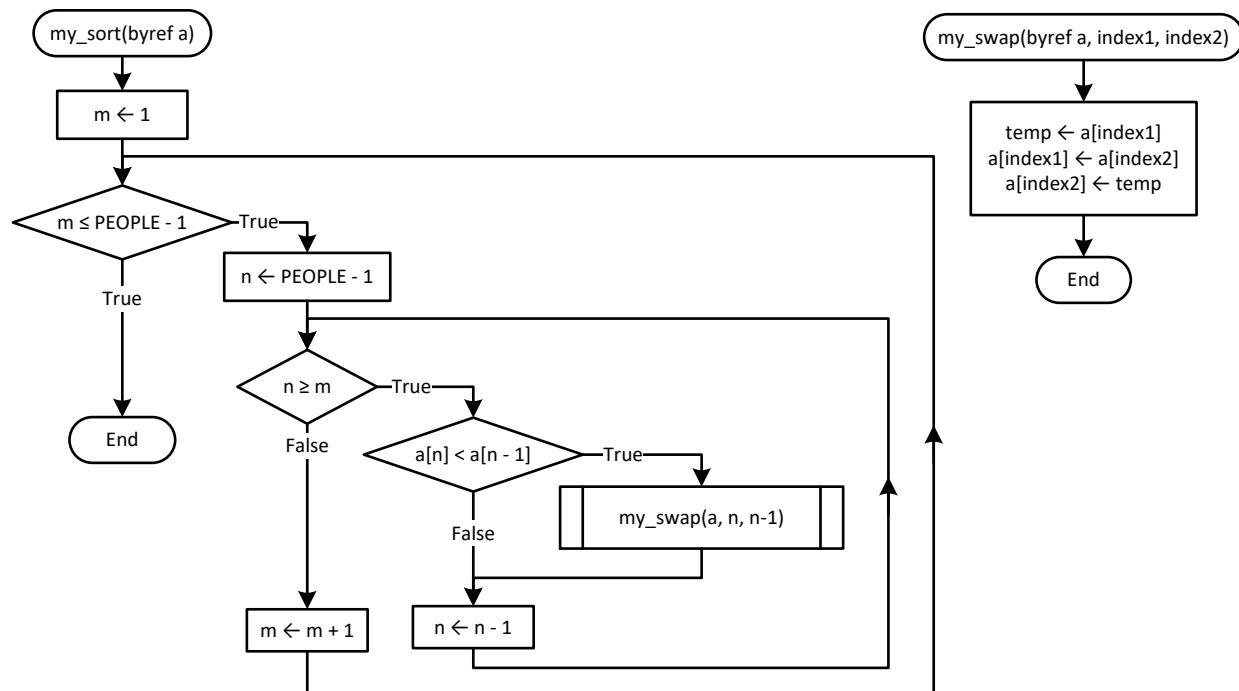


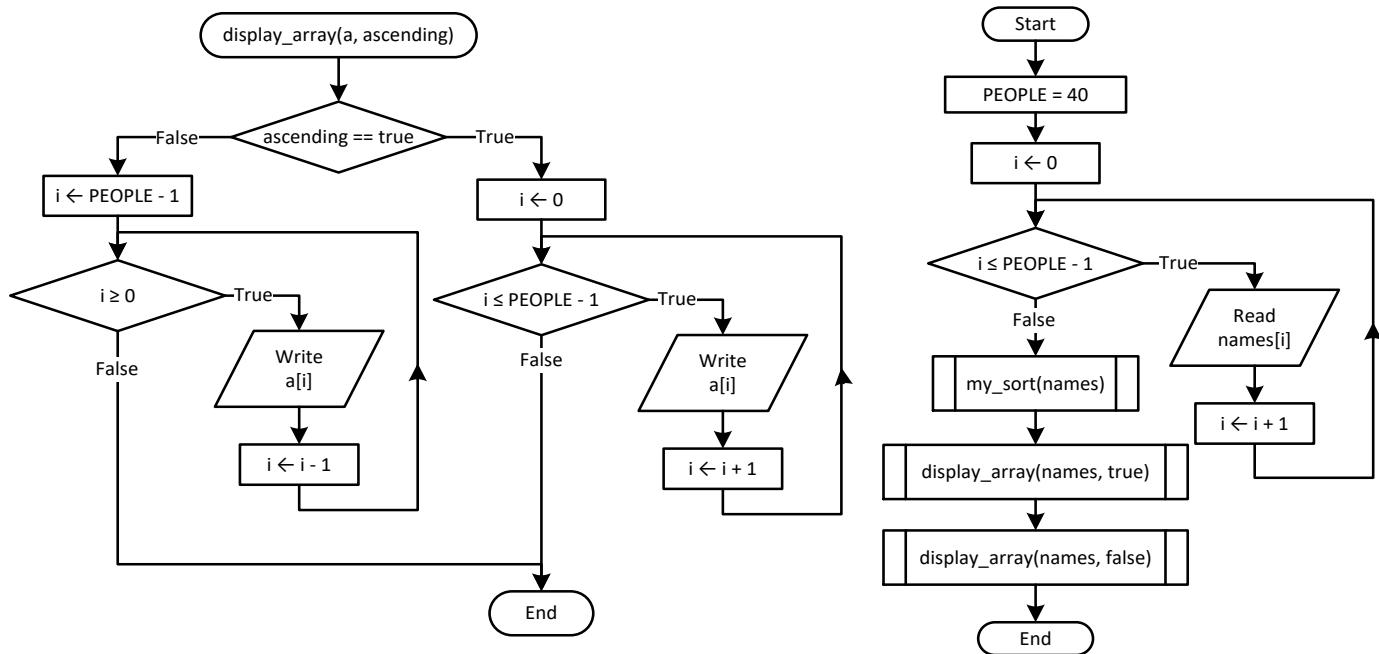
4. Solution





5. Solution





6. Solution

```

#include <iostream>
#include <cmath>
using namespace std;

double f1(int n) {
    double s;
    int i;

    s = 0;
    for (i = 1; i <= n; i++) {
        if (i < n / 2.0) {
            s += pow(n, 2);
        }
        else {
            s += pow(n, 3);
        }
    }
    return s;
}

int main() {
    int val;

    do {
        cout << "Enter a positive integer ";
        cin >> val;
    } while (val < 0);
    cout << f1(val) << endl;
    return 0;
}

```

7. Solution

```
#include <iostream>
using namespace std;

const int ELEMENTS = 100;

double *read_values() {
    static double values[ELEMENTS];
    int i;

    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> values[i];
    }
    return values;
}

void find_min_max(double values[], int &min_i, int &max_i) {
    int i;

    min_i = max_i = 0;
    for (i = 1; i <= ELEMENTS - 1; i++) {
        if (values[i] < values[min_i]) {
            min_i = i;
        }
        if (values[i] > values[max_i]) {
            max_i = i;
        }
    }
}

int main() {
    int min_i = 0, max_i = 0;

    double *v = read_values();
    find_min_max(v, min_i, max_i);
    cout << v[min_i] << ", " << v[max_i] << endl;
    return 0;
}
```

 Please note the way the void method `find_min_max()` finds the index positions of the minimum and the maximum values of the array `values`. This method is not the same as the one you learned in paragraph 34.3; however, it can be used as an alternative.

8. Solution

```
#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
using namespace std;

const double ACCURACY = 0.000000001;
```

```
double factorial(int n) {
    int i;

    double return_value = 1;
    for (i = 1; i <= n; i++) {
        return_value *= i;
    }
    return return_value;
}

double my_sin(double x) {
    int i, sign;
    double sinus, sinus_previous;
    sign = 1;
    sinus = 0;
    i = 1;
    do {
        sinus_previous = sinus;
        sinus += sign * pow(x, i) / factorial(i);

        sign = -sign;
        i += 2;
    } while (abs(sinus - sinus_previous) > ACCURACY);
    return sinus;
}

double degrees_to_rad(double degrees) {
    return 2 * M_PI * degrees / 360;
}

int main() {
    int i;

    for (i = 0; i <= 360; i++) {
        cout << "sin(" << i << ") ~= " << my_sin(degrees_to_rad(i)) << endl;
    }
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;

bool is_leap(int year) {
    bool return_value = false;
    if (year % 4 == 0 && year % 100 != 0 || year % 400 == 0) {
        return_value = true;
    }
    return return_value;
}

int num_of_days(int year, int month) {
```

```
int days;

switch (month) {
    case 4:
    case 6:
    case 9:
    case 11:
        days = 30;
        break;
    case 2:
        if (is_leap(year)) {
            days = 29;
        }
        else {
            days = 28;
        }
        break;
    default:
        days = 31;
}

return days;
}

bool check_date(int day, int month, int year) {
    bool return_value = true;
    if (month < 1 || month > 12) {
        return_value = false;
    }
    else if (day < 1 || day > num_of_days(year, month)) {
        return_value = false;
    }
    return return_value;
}

int main() {
    int day, month, year, total, i;

    cout << "Enter day: ";
    cin >> day;
    cout << "Enter month: ";
    cin >> month;
    cout << "Enter year: ";
    cin >> year;
    while (!check_date(day, month, year)) {
        cout << "Error!" << endl;
        cout << "Enter day: ";
        cin >> day;
        cout << "Enter month: ";
        cin >> month;
        cout << "Enter year: ";
        cin >> year;
    }
}
```

```
    }

    total = 0;
    for (i = 1; i <= month - 1; i++) {
        total += num_of_days(year, i);
    }
    total += day;

    cout << total << endl;
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;

void display_menu() {
    cout << "-----" << endl;
    cout << "1. Convert USD to Euro (EUR)" << endl;
    cout << "2. Convert USD to British Pound Sterling (GBP)" << endl;
    cout << "3. Convert EUR to USD" << endl;
    cout << "4. Convert EUR to GBP" << endl;
    cout << "5. Convert GBP to USD" << endl;
    cout << "6. Convert GBP to EUR" << endl;
    cout << "7. Exit" << endl;
    cout << "-----" << endl;
    cout << "Enter a choice: ";
}

double USD_to_EUR(double value) {
    return value * 0.87;
}

double USD_to_GBP(double value) {
    return value * 0.76;
}

int main() {
    int choice;
    double amount;

    display_menu();
    cin >> choice;
    while (choice != 7) {
        cout << "Enter an amount: ";
        cin >> amount;
        switch (choice) {
            case 1:
                cout << amount << " USD = " << USD_to_EUR(amount) << " Euro" << endl;
                break;
            case 2:
                cout << amount << " USD = " << USD_to_GBP(amount) << " GBP" << endl;
        }
    }
}
```

```
        break;
    case 3:
        cout << amount << " EUR = " << 1 / USD_to_EUR(1 / amount) << " USD" << endl;
        break;
    case 4:
        cout << amount << " EUR = " << USD_to_GBP(1 / USD_to_EUR(1 / amount)) << " GBP" << endl;
        break;
    case 5:
        cout << amount << " GBP = " << 1 / USD_to_GBP(1 / amount) << " USD" << endl;
        break;
    case 6:
        cout << amount << " GBP = " << USD_to_EUR(1 / USD_to_GBP(1 / amount)) << " EUR" << endl;
        break;
    }

    display_menu();
    cin >> choice;
}
return 0;
}
```

11. Solution

```
#include <iostream>
#include <string>
#include <ctime>
#include <cstdlib>
using namespace std;

int dice() {
    return 1 + rand() % 6;
}

int main() {
    int dice1, dice2, i, player, total, total_player1 = 0, total_player2 = 0;
    string key, names[2];

    srand(time(NULL));

    cout << "Player1 - Enter name: ";
    cin >> names[0];
    cout << "Player2 - Enter name: ";
    cin >> names[1];

    for (player = 0; player <= 1; player++) {
        total = 0;
        for (i = 1; i <= 10; i++) {
            cout << names[player] << ", hit enter to roll the dice!" << endl;
            getline(cin, key); //This statement just waits the user to hit the enter key

            dice1 = dice();
            dice2 = dice();
            cout << dice1 << " " << dice2 << endl;
        }
    }
}
```

```
        total += dice1 + dice2;
    }
    if (player == 1) {
        total_player1 = total;
    }
    else {
        total_player2 = total;
    }
}

if (total_player1 == total_player2) {
    cout << "Tie!" << endl;
}
else if (total_player1 > total_player2) {
    cout << names[0] << " wins" << endl;
}
else {
    cout << names[1] << " wins" << endl;
}
return 0;
}
```

12. Solution

```
#include <iostream>
using namespace std;

const int GAS = 1;
const int DIESEL = 2;
const int HYBRID = 3;
const double TAX_RATE = 0.10;
const int CARS = 40;

int get_choice() {
    int choice;
    cout << "1. Gas" << endl;
    cout << "2. Diesel" << endl;
    cout << "3. Hybrid" << endl;
    cout << "Enter type of the car: ";
    cin >> choice;
    return choice;
}

int get_days() {
    int days;
    cout << "Enter total number of rental days: ";
    cin >> days;
    return days;
}

double get_charge(int car_type, int rental_days) {
    double charge;
```

```
if (car_type == GAS) {
    if (rental_days <= 5) {
        charge = rental_days * 24;
    }
    else if (rental_days <= 8) {
        charge = 5 * 24 + (rental_days - 5) * 22;
    }
    else {
        charge = 5 * 24 + 3 * 22 + (rental_days - 8) * 18;
    }
}
else if (car_type == DIESEL) {
    if (rental_days <= 5) {
        charge = rental_days * 28;
    }
    else if (rental_days <= 8) {
        charge = 5 * 28 + (rental_days - 5) * 25;
    }
    else {
        charge = 5 * 28 + 3 * 25 + (rental_days - 8) * 21;
    }
}
else {
    if (rental_days <= 5) {
        charge = rental_days * 30;
    }
    else if (rental_days <= 8) {
        charge = 5 * 30 + (rental_days - 5) * 28;
    }
    else {
        charge = 5 * 30 + 3 * 28 + (rental_days - 8) * 23;
    }
}
charge = charge * (1 + TAX_RATE); //This is equivalent to charge += charge * TAX_RATE;
return charge;
}

int main() {
    int count, i;
    double charge, total;

    int rented_car_types[CARS];
    int rented_days[CARS];

    for (i = 0; i <= CARS - 1; i++) {
        rented_car_types[i] = get_choice();
        rented_days[i] = get_days();
    }

    total = 0;
    for (i = 0; i <= CARS - 1; i++) {
        charge = get_charge(rented_car_types[i], rented_days[i]);
```

```

        cout << "Car No " << i + 1 << ":" << charge << endl;
        total += charge;
    }

    count = 0;
    for (i = 0; i <= CARS - 1; i++) {
        if (rented_car_types[i] == HYBRID) {
            count++;
        }
    }

    cout << "Hybrids rented: " << count << endl;
    cout << "Net profit: " << total / (1 + TAX_RATE) << endl;
    return 0;
}

```

13. Solution

```

#include <iostream>
using namespace std;

const int CHANNELS = 10;
const int DAYS = 7;
const string day_names[] = {"Monday", "Tuesday", "Wednesday",
                           "Thursday", "Friday", "Saturday", "Sunday"};

//Note that in C++, in order to pass multidimensional arrays to functions,
//the arrays in the formal argument list must have bounds for all dimensions except the first
void get_data(string names[],  int viewers[] [DAYS]) {
    int i, j;

    for (i = 0; i <= CHANNELS - 1; i++) {
        cout << "Enter name for channel No. " << i + 1 << ":" << endl;
        cin >> names[i];
        for (j = 0; j <= DAYS - 1; j++) {
            cout << "Enter the number of viewers of the main news program on " << day_names[j];
            cout << " for channel " << names[i] << ":" ;
            cin >> viewers[i][j];
        }
    }
}

double get_average(int a[]) {
    int total ,i;

    total = 0;
    for (i = 0; i <= 4; i++) {
        total += a[i];
    }
    return total / 5.0;
}

int main() {

```

```

int i, j;
double weekend;
bool increasing;

string names[CHANNELS];
int viewers[CHANNELS][DAYS];
get_data(names, viewers);

int temporary_array[5];
for (i = 0; i <= CHANNELS - 1; i++) {
    for (j = 0; j <= 4; j++) {
        temporary_array[j] = viewers[i][j];
    }
    weekend = (viewers[i][DAYS - 2] + viewers[i][DAYS - 1]) / 2;
    if (weekend >= 1.2 * get_average(temporary_array)) {
        cout << names[i] << endl;
    }
}

for (i = 0; i <= CHANNELS - 1; i++) {
    increasing = true;
    for (j = 1; j <= DAYS - 1; j++) {
        if (viewers[i][j] <= viewers[i][j - 1]) {
            increasing = false;
        }
    }
    if (increasing) {
        cout << names[i] << endl;
    }
}
return 0;
}

```

14. Solution

```

#include <iostream>
using namespace std;

const int CITIZENS = 300;

void input_data(long long SSNs[], string answers[]) {
    int i;

    for (i = 0; i <= CITIZENS - 1; i++) {
        cout << "Enter SSN: ";
        cin >> SSNs[i];
        cout << "Enter answer: ";
        cin >> answers[i];
    }
}

void sort_arrays(long long SSNs[], string answers[]) {
    int m, n, index_of_min;

```

```
long long minimum, temp;
string temp_str;

for (m = 0; m <= CITIZENS - 1; m++) {
    minimum = SSNs[m];
    index_of_min = m;
    for (n = m; n <= CITIZENS - 1; n++) {
        if (SSNs[n] < minimum) {
            minimum = SSNs[n];
            index_of_min = n;
        }
    }
    temp = SSNs[m];
    SSNs[m] = SSNs[index_of_min];
    SSNs[index_of_min] = temp;
    temp_str = answers[m];
    answers[m] = answers[index_of_min];
    answers[index_of_min] = temp_str;
}
}

int search_array(long long SSNs[], long long SSN) {
    int left, right, middle, index_position = 0, return_value;
    bool found;

    left = 0;
    right = CITIZENS - 1;
    found = false;
    while (left <= right && !found) {
        middle = (int)((left + right) / 2);

        if (SSNs[middle] > SSN) {
            right = middle - 1;
        }
        else if (SSNs[middle] < SSN) {
            left = middle + 1;
        }
        else {
            found = true;
            index_position = middle;
        }
    }

    if (!found) {
        cout << "SSN not found!" << endl;
        return_value = -1;
    }
    else {
        return_value = index_position;
    }
}
return return_value;
}
```

```
int count_answers(string answers[], string answer) {
    int count, i;

    count = 0;
    for (i = 0; i <= CITIZENS - 1; i++) {
        if (answers[i] == answer) {
            count++;
        }
    }
    return count;
}

int main() {
    long long SSNs[CITIZENS];
    long long SSN;
    string answers[CITIZENS];
    int index, count;
    string answer;

    do {
        input_data(SSNs, answers);
        sort_arrays(SSNs, answers);

        cout << "Enter an SSN to search: ";
        cin >> SSN;

        index = search_array(SSNs, SSN);
        if (index != -1) {
            answer = answers[index];
            cout << answer << endl;

            count = count_answers(answers, answer);
            cout << count * 100 / (double)CITIZENS << endl;
        }
        cout << "Repeat? " << endl;
        cin >> answer;
    } while (answer == "yes");
    return 0;
}
```

15. Solution

```
#include <iostream>
using namespace std;

const int TEAMS = 8;
const int GAMES = 12;

//Note that in C++, in order to pass multidimensional arrays to functions,
//the arrays in the formal argument list must have bounds for all dimensions except the first
void input_data(string names[], string results[][GAMES]) {
    int i, j;
```

```
for (i = 0; i <= TEAMS - 1; i++) {
    cout << "Enter team name: ";
    cin >> names[i];
    for (j = 0; j <= GAMES - 1; j++) {
        cout << "Enter result (W, L, T): ";
        cin >> results[i][j];
    }
}
}

void display_result(string names[], string results[][GAMES]) {
    string result;
    int i, j;
    bool found;

    cout << "Enter a result to search (W, L, T): ";
    cin >> result;
    for (i = 0; i <= TEAMS - 1; i++) {
        cout << "Team: " << names[i] << endl;
        found = false;
        for (j = 0; j <= GAMES - 1; j++) {
            if (results[i][j] == result) {
                cout << "Week: " << j + 1 << endl;
                found = true;
            }
        }
        if (!found) {
            cout << "Nothing found" << endl;
        }
    }
}

int find_team(string names[]) {
    string name;
    int i, return_value;

    cout << "Enter a name to search: ";
    cin >> name;

    i = 0;
    while (i < TEAMS - 1 && names[i] != name) {
        i++;
    }

    if (names[i] != name) {
        return_value = -1;
    }
    else {
        return_value = i;
    }
    return return_value;
}
```

```
int main() {
    string names[TEAMS];
    string results[TEAMS][GAMES];
    int j, index, total;

    input_data(names, results);
    display_result(names, results);

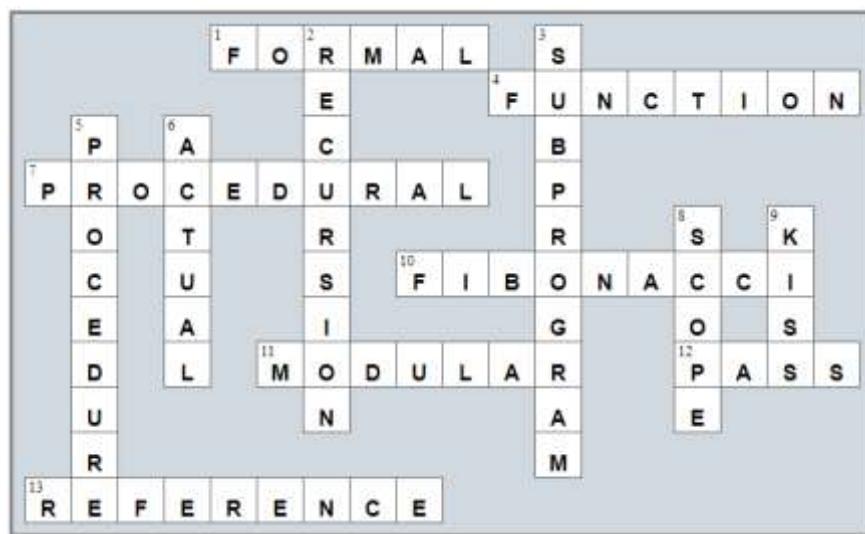
    index = find_team(names);
    while (index != -1) {
        total = 0;
        for (j = 0; j <= GAMES - 1; j++) {
            if (results[index][j] == "W") {
                total += 3;
            }
            else if (results[index][j] == "T") {
                total += 1;
            }
        }
        cout << "Points: " << total << endl;
        index = find_team(names);
    }

    if (index == -1) {
        cout << "Team not found" << endl;
    }
    return 0;
}
```

Review in “Subprograms”

Review Crossword Puzzle

1.



Chapter 39

39.8 Review Questions: True/False

- | | | |
|----------|-----------|-----------|
| 1. false | 7. false | 13. true |
| 2. true | 8. true | 14. false |
| 3. true | 9. true | 15. true |
| 4. false | 10. false | 16. false |
| 5. false | 11. true | 17. false |
| 6. false | 12. true | |

39.9 Review Exercises

1. Solution

```
#include <iostream>
using namespace std;

class Trigonometry {
public:
    double square_area(double side) {
        return side * side;
    }

    double rectangle_area(double b, double h) {
        return b * h;
    }

    double triangle_area(double b, double h) {
        return b * h / 2;
    }
};

int main() {
    double sqr_side, rctngl_base, rctngl_height, trngl_base, trngl_height;
    Trigonometry tr;

    cout << "Enter square side: ";
    cin >> sqr_side;

    cout << "Enter rectangle base: ";
    cin >> rctngl_base;
    cout << "Enter rectangle height: ";
    cin >> rctngl_height;

    cout << "Enter triangle base: ";
    cin >> trngl_base;
    cout << "Enter triangle height: ";
    cin >> trngl_height;

    cout << tr.square_area(sqr_side) << endl;
    cout << tr.rectangle_area(rctngl_base, rctngl_height) << endl;
    cout << tr.triangle_area(trngl_base, trngl_height) << endl;
}
```

```
    return 0;
}
```

2. Solution

```
#include <iostream>
using namespace std;

class Pet {
public:
    string kind;
    int legs_number;

    void start_running() {
        cout << "Pet is running" << endl;
    }

    void stop_running() {
        cout << "Pet stopped" << endl;
    }
};

int main() {
    Pet pet1;
    pet1.kind = "dog";
    pet1.legs_number = 4;

    Pet pet2;
    pet2.kind = "monkey";
    pet2.legs_number = 2;

    pet1.start_running();
    pet2.start_running();
    pet1.stop_running();
    return 0;
}
```

3. Solution

```
#include <iostream>
using namespace std;

class Pet {
private:
    string _kind;
    int _legs_number;

public:
    //Define the constructor
    Pet(string kind, int legs_number) {
        this->setKind(kind);
        this->setLegs_number(legs_number);
    }
};
```

```
//Define the getter
string getKind() {
    return this->_kind;
}

//Define the setter
void setKind(string value) {
    if (value != "") {
        this->_kind = value;
    }
    else {
        throw "Cannot be empty";
    }
}

//Define the getter
int getLegs_number() {
    return this->_legs_number;
}

void setLegs_number(int value) {
    if (value >= 0) {
        this->_legs_number = value;
    }
    else {
        throw "Cannot be negative";
    }
}

void start_running() {
    cout << "Pet is running" << endl;
}

void stop_running() {
    cout << "Pet stopped" << endl;
};

int main() {
    Pet pet1("dog", 4);

    pet1.start_running();
    pet1.stop_running();

    pet1.setKind(""); //This will throw an error
    pet1.setLegs_number(-1); //This will throw an error
    return 0;
}
```

4. Solution

```
#include <iostream>
using namespace std;
```

```
const int BOXES = 3;

class Box {
private:
    double _width = 0;
    double _length = 0;
    double _height = 0;

public:
    //Define an empty constructor
    Box() { }

    //Define the constructor
    Box(double w, double l, double h) {
        //Initialize fields
        this->_width = w;
        this->_length = l;
        this->_height = h;
    }

    void display_volume() {
        cout << "Volume: " << this->_width * this->_length * this->_height << endl;
    }

    void display_dimensions() {
        cout << this->_width << " x " << this->_length << " x " << this->_height << endl;
    }
};

int main() {
    int i;
    double w, l, h;

    Box list_of_obj[BOXES]; //Create an array of objects. It calls the first constructor (the empty one).

    for (i = 0; i <= BOXES - 1; i++) {
        cout << "Enter width: ";
        cin >> w;
        cout << "Enter length: ";
        cin >> l;
        cout << "Enter height: ";
        cin >> h;

        //Add each new object to the array
        list_of_obj[i] = Box(w, l, h); //It calls the second constructor
    }

    for (i = 0; i <= BOXES - 1; i++) {
        list_of_obj[i].display_dimensions();
        list_of_obj[i].display_volume();
    }

    return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
const int BOXES = 3;

class Box {
private:
    double _width;
    double _length;
    double _height;

public:
    //Define an empty constructor
    Box() { }

    //Define the constructor
    Box(double w, double l, double h) {
        //Initialize fields
        this->setWidth(w);
        this->setLength(l);
        this->setHeight(h);
    }

    //Define the getter
    double getWidth() {
        return this->_width;
    }

    //Define the setter
    void setWidth(double value) {
        if (value > 0) {
            this->_width = value;
        }
        else {
            throw "Cannot be negative or zero";
        }
    }

    //Define the getter
    double getLength() {
        return this->_length;
    }

    //Define the setter
    void setLength(double value) {
        if (value > 0) {
            this->_length = value;
        }
        else {
            throw "Cannot be negative or zero";
        }
    }
}
```

```
}

//Define the getter
double getHeight() {
    return this->_height;
}

//Define the setter
void setHeight(double value) {
    if (value > 0) {
        this->_height = value;
    }
    else {
        throw "Cannot be negative or zero";
    }
}

void display_volume() {
    cout << "Volume: " << this->getWidth() * this->getLength() * this->getHeight() << endl;
}

void display_dimensions() {
    cout << this->getWidth() << " x " << this->getLength() << " x " << this->getHeight() << endl;
}
};

int main() {
    int i;
    double w, l, h;

    Box list_of_obj[BOXES]; //Create an array of objects. It calls the first constructor (the empty one).

    for (i = 0; i <= BOXES - 1; i++) {
        cout << "Enter width: ";
        cin >> w;
        cout << "Enter length: ";
        cin >> l;
        cout << "Enter height: ";
        cin >> h;

        //Add each new object to the array
        list_of_obj[i] = Box(w, l, h); //It calls the second constructor
    }

    for (i = 0; i <= BOXES - 1; i++) {
        list_of_obj[i].display_dimensions();
        list_of_obj[i].display_volume();
    }
    return 0;
}
```

6. Solution

```
#include <iostream>
#include <cmath>
using namespace std;

class Cube {
private:
    double _edge;

public:
    //Define the constructor
    Cube(double edge) {
        this->_edge = edge;
    }

    void display_volume() {
        cout << "Volume: " << pow(this->_edge, 3) << endl;
    }

    void display_one_surface() {
        cout << "One surface: " << pow(this->_edge, 2) << endl;
    }

    void display_total_surface() {
        cout << "Total surface: " << 6 * pow(this->_edge, 2) << endl;
    }
};

int main() {
    double edge;

    cout << "Enter edge length of a cube: ";
    cin >> edge;

    Cube cube1(edge);

    cube1.display_volume();
    cube1.display_one_surface();
    cube1.display_total_surface();
    return 0;
}
```

7. Solution

```
#include <iostream>
#include <cmath>

using namespace std;
class Cube {
private:
    double _edge;

public:
```

```
//Define the constructor
Cube(double edge) {
    this->setEdge(edge);
}

//Define the getter
double getEdge() {
    return this->_edge;
}

//Define the setter
void setEdge(double value) {
    if (value > 0) {
        this->_edge = value;
    }
    else {
        throw "Cannot be negative or zero";
    }
}

void display_volume() {
    cout << "Volume: " << pow(this->getEdge(), 3) << endl;
}

void display_one_surface() {
    cout << "One surface: " << pow(this->getEdge(), 2) << endl;
}

void display_total_surface() {
    cout << "Total surface: " << 6 * pow(this->getEdge(), 2) << endl;
};

int main() {
    double edge;

    cout << "Enter edge length of a cube: ";
    cin >> edge;

    Cube cube1(edge);

    cube1.display_volume();
    cube1.display_one_surface();
    cube1.display_total_surface();
    return 0;
}
```

8. Solution

```
#include <iostream>
#include <cmath>
using namespace std;

class Circle {
```

```
private:
    double _radius = -1;

public:
    //Define the getter
    double getRadius() {
        if (this->_radius > 0) {
            return this->_radius;
        }
        else {
            throw "Radius is not set";
        }
    }

    //Define the setter
    void setRadius(double value) {
        if (value > 0) {
            this->_radius = value;
        }
        else {
            throw "Cannot be negative or zero";
        }
    }

    double get_diameter() {
        return 2 * this->getRadius();
    }

    double get_area() {
        return 3.14 * pow(this->getRadius(), 2);
    }

    double get_perimeter() {
        return 2 * 3.14 * this->getRadius();
    }
};

void display_menu() {
    cout << "1. Enter radius" << endl;
    cout << "2. Display radius" << endl;
    cout << "3. Display diameter" << endl;
    cout << "4. Display area" << endl;
    cout << "5. Display perimeter" << endl;
    cout << "6. Exit" << endl;
}

int main() {
    int choice;
    double radius;

    Circle circle1;

    while (true) {
```

```
display_menu();
cout << "Enter a choice: ";
cin >> choice;

if (choice == 6) {
    cout << "Bye" << endl;
    break;
}
else if (choice == 1) {
    cout << "Enter radius: ";
    cin >> radius;
    circle1.setRadius(radius);
}
else if (choice == 2) {
    cout << "Radius: " << circle1.getRadius() << endl;
}
else if (choice == 3) {
    cout << "Diameter: " << circle1.get_diameter() << endl;
}
else if (choice == 4) {
    cout << "Area: " << circle1.get_area() << endl;
}
else if (choice == 5) {
    cout << "Perimeter: " << circle1.get_perimeter() << endl;
}
}
return 0;
}
```

9. Solution

```
#include <iostream>
#include <string>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;

class Info {
private:
    string _user_text;

public:
    //Define the getter
    string getUser_text() {
        return this->_user_text;
    }

    //Define the setter
    void setUser_text(string value) {
        if (value != "") {
            this->_user_text = value;
        }
    }
}
```

```
    else {
        throw "Cannot be set to empty";
    }
}

int get_spaces_count() {
    int i, count = 0;
    string character;
    string text = this->getUser_text();

    for (i = 0; i <= text.length() - 1; i++) {
        character = text[i];
        if (character == " ") {
            count += 1;
        }
    }
    return count;
}

int get_words_count() {
    return this->get_spaces_count() + 1;
}

int get_vowels_count() {
    int i, count = 0;
    char character;
    string text = this->getUser_text();
    string vowels = "aeiou";

    for (i = 0; i <= text.length() - 1; i++) {
        character = to_lower_copy(text)[i];
        if (vowels.find(character) != -1) {
            count += 1;
        }
    }
    return count;
}

int get_letters_count() {
    string text = this->getUser_text();

    return text.length() - this->get_spaces_count();
}

int main() {
    string text;

    Info inf;

    cout << "Enter a text: ";
    getline(cin, text);
    inf.setUser_text(text);
```

```
    cout << "Text: " << inf.getUser_text() << endl;
    cout << "Spaces: " << inf.get_spaces_count() << endl;
    cout << "Words: " << inf.get_words_count() << endl;
    cout << "Vowels: " << inf.get_vowels_count() << endl;
    cout << "Total number of letters: " << inf.get_letters_count() << endl;
    return 0;
}
```

10. Solution

```
#include <iostream>
#include <string>
using namespace std;

class EncryptDecrypt {
    const string alphabet = " abcdefghijklmnopqrstuvwxyz"; //space is a valid character!
private:
    int _encr_decr_key = -1;

public:
    //Define the getter
    int getEncr_decr_key() {
        if (this->_encr_decr_key != -1) {
            return this->_encr_decr_key;
        }
        else {
            throw "Key is not set";
        }
    }

    //Define the setter
    void setEncr_decr_key(int value) {
        if (value >= 1 && value <= 26) {
            this->_encr_decr_key = value;
        }
        else {
            throw "Must be between 1 and 26";
        }
    }

    string encrypt(string message) {
        string return_value = "";
        char character, new_letter;
        int i, index, new_index;

        for (i = 0; i <= message.length() - 1; i++) {
            character = message[i];
            index = alphabet.find(character);
            new_index = index + this->getEncr_decr_key();
            if (new_index >= 27) {
                new_index -= 27;
            }
        }
    }
}
```

```
    new_letter = alphabet[new_index];
    return_value += new_letter;
}
return return_value;
}

string decrypt(string enc_message) {
    string return_value = "";
    char character, new_letter;
    int i, index, new_index;

    for (i = 0; i <= enc_message.length() - 1; i++) {
        character = enc_message[i];
        index = alphabet.find(character);
        new_index = index - this->getEncr_decr_key();
        if (new_index < 0) {
            new_index += 27;
        }
        new_letter = alphabet[new_index];
        return_value += new_letter;
    }
    return return_value;
};

void display_menu() {
    cout << "1. Enter encryption/decryption key" << endl;
    cout << "2. Encrypt a message" << endl;
    cout << "3. Decrypt a message" << endl;
    cout << "4. Exit" << endl;
}

int main() {
    string text;
    int choice, encr_decr_key;

    EncryptDecrypt ed;

    display_menu();
    cout << "Enter a choice: ";
    cin >> choice;

    //When there is a cin statement before a getline() function
    //you need to write the following statement between them
    //otherwise the next getline() function won't work
    cin.ignore(100, '\n');

    while (choice != 4) {
        if (choice == 1) {
            cout << "Enter encryption/decryption key: ";
            cin >> encr_decr_key;
            ed.setEncr_decr_key(encr_decr_key);
```

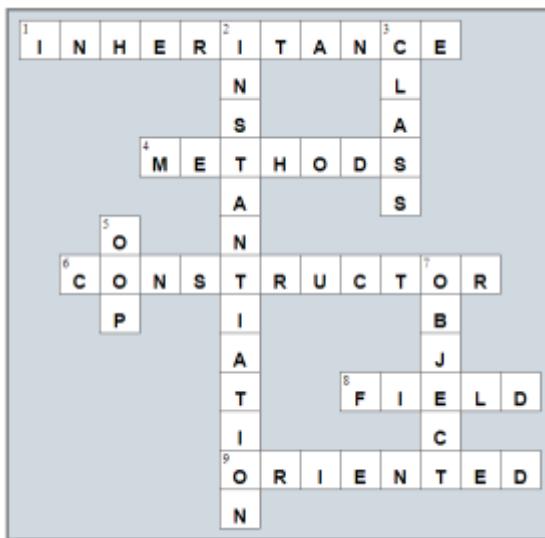
```
    }
else if (choice == 2) {
    cout << "Enter message to encrypt: ";
    getline(cin, text);
    cout << "Encrypted message: " << ed.encrypt(text) << endl;
}
else if (choice == 3) {
    cout << "Enter message to decrypt: ";
    getline(cin, text);
    cout << "Decrypted message: " << ed.decrypt(text) << endl;
}

display_menu();
cout << "Enter a choice: ";
cin >> choice;
}
return 0;
}
```

Review in “Object Oriented Programming”

Review Crossword Puzzle

1.



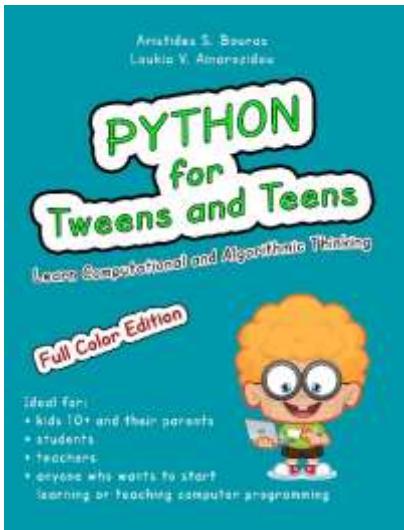
Some Final Words from the Author

I hope you really enjoyed reading this book. I made every possible effort to make it comprehensible even by people that probably have no previous experience in programming.

So if you liked this book, please visit the web store where you bought it and show me your gratitude by writing a good review and giving me as many stars as possible. By doing this, you will encourage me to continue writing and of course you'll help other readers to reach me.

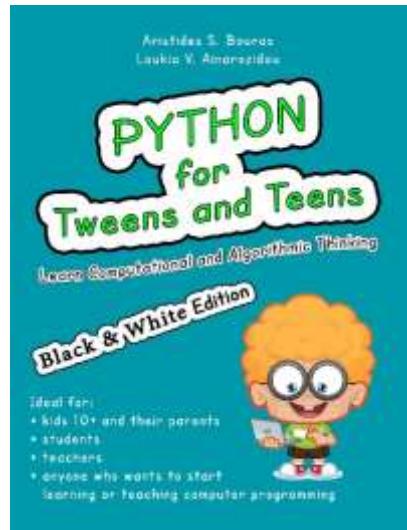
And remember: Learning is a process within an endless loop. It begins at birth and continues throughout your lifetime!

Some of my Books



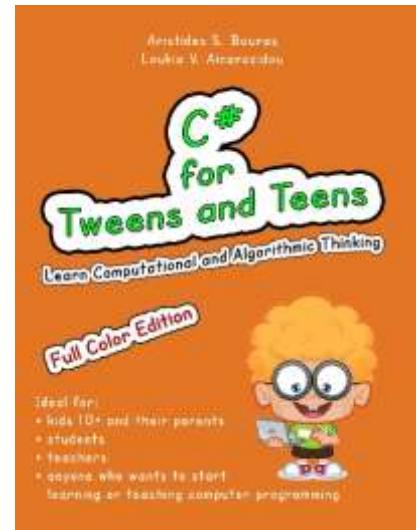
ISBN-10: 1543127940

ISBN-13: 978-1543127942



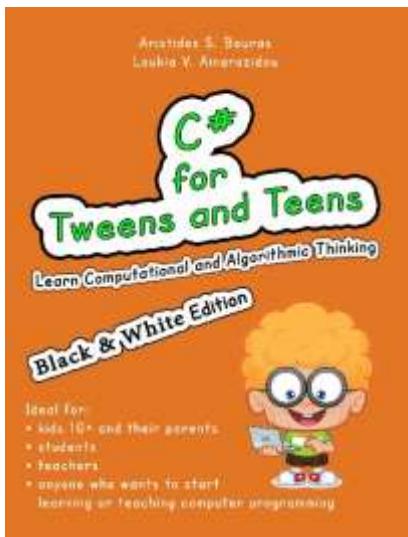
ISBN-10: 1546611215

ISBN-13: 978-1546611219



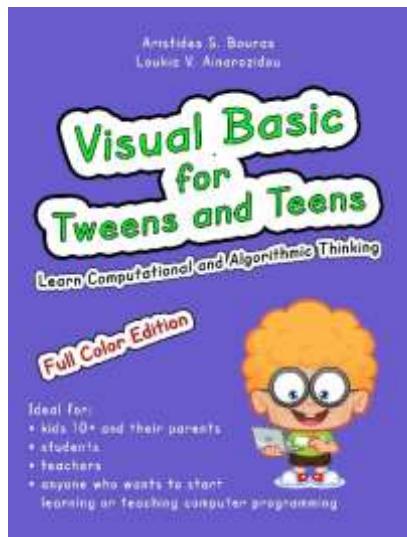
ISBN-10: 1973727684

ISBN-13: 978-1973727682



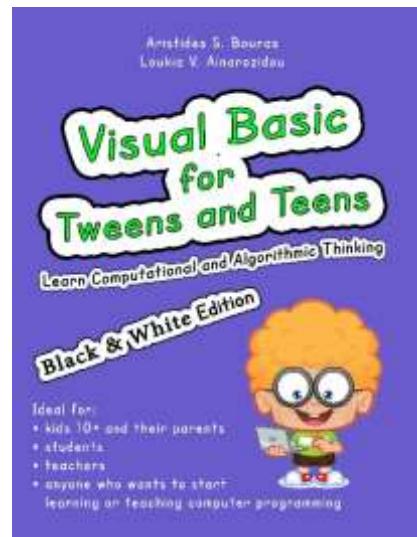
ISBN-10: 1973727765

ISBN-13: 978-1973727767



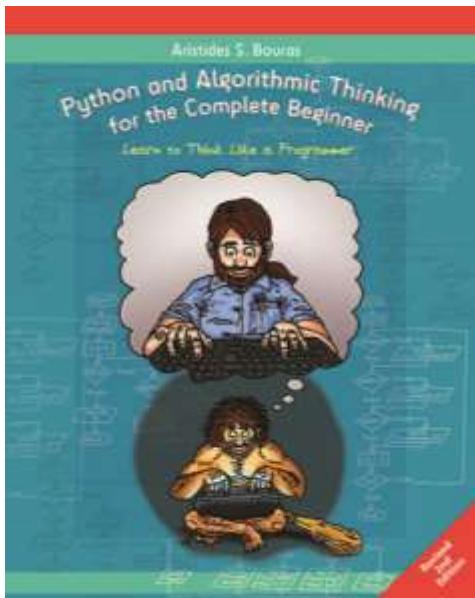
ISBN-10: 1982083670

ISBN-13: 978-1982083670

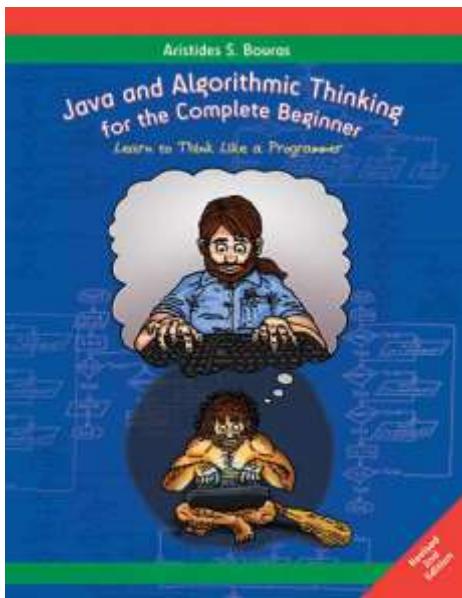


ISBN-10: 1982083697

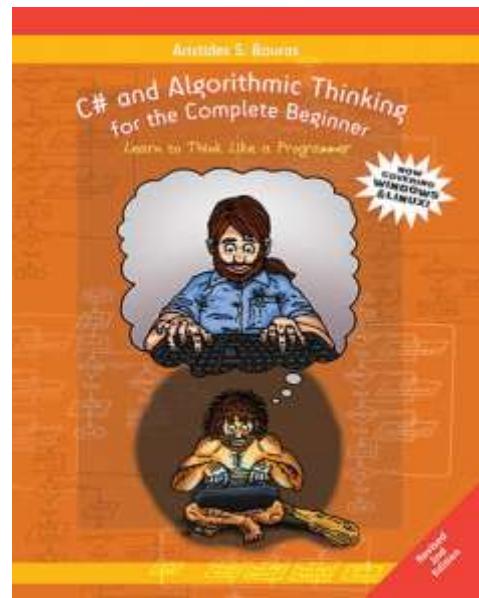
ISBN-13: 978-1982083694



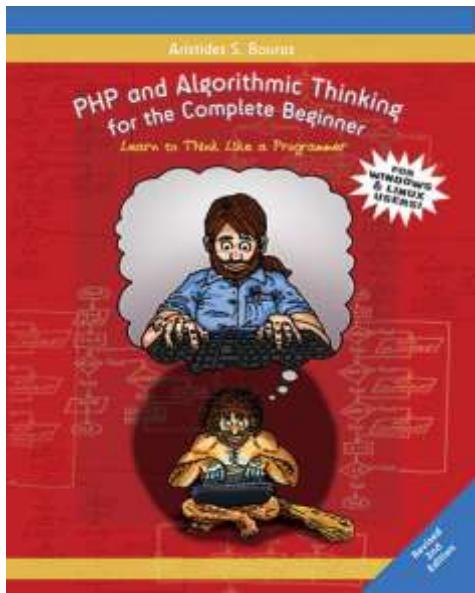
ISBN-10: 1099184878
ISBN-13: 978-1099184871



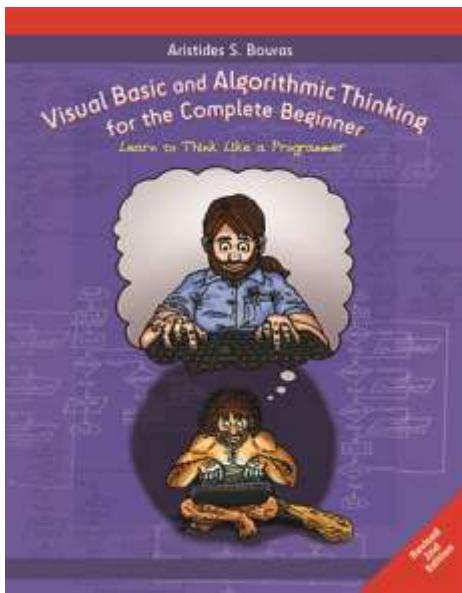
ISBN-10: 1698811233
ISBN-13: 978-1698811239



ISBN-10: 1671594363
ISBN-13: 978-1671594364



ISBN-13: 979-8591248291



ISBN-13: 979-8719394534