C++
AND ALGORITHMIC THINKING FOR THE COMPLETE BEGINNER

The Answers

Aristides S. Bouras
Loukia V. Ainarozidou

RCode: 180927

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". Every effort has been taken to make this book compatible with all previous releases of C++, and it is almost certain to be compatible with any future releases of C++.

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

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

http://www.bouraspage.com

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

Chapter 1

1.7 Answers of Review Questions: True/False

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

1.8 Answers of Review Questions: Multiple Choice

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

Chapter 4

4.16 Answers of Review Questions: True/False

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

4.17 Answers of Review Questions: Multiple Choice

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

Chapter 5

5.9 Answers of Review Questions: True/False

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

5.10 Answers of Review Questions: Multiple Choice

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

5.11 Answers of Review Exercises

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

Chapter 6

6.4 Answers of Review Questions: True/False

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

6.5 Answers of Review Questions: Multiple Choice

1. a
2. b
3. b

Chapter 7

7.7 Answers of Review Questions: True/False

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

7.8 Answers of Review Questions: Multiple Choice

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

7.9 Answers of Review Exercises

1. ii, iv, v, ix
2. i. String, ii. Boolean, iii. String, iv. String, v. Real, 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. Divisio by zero error, vi. 0
9. i. 2, ii. 10.5
10. My name is George Malkovich
11. i. (-3), ii. 1
12. California, California

Chapter 8

8.2 Answers of Review Questions: True/False

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

8.3 Answers of Review Exercises

1. Solution

For the input value of 3

Step	Statement	a	b	c	d
1	`cin >> a`	**3**	?	?	?
2	`a = (a + 1) * (a + 1) + 6 / 3 * 2 + 20`	**40**	?	?	?
3	`b = a % 13`	40	**1**	?	?
4	`c = b % 7`	40	1	**1**	?
5	`d = a * b * c`	40	1	1	**40**
6	`cout << a << ", " << b << ", " << c << ", " << d`	40, 1, 1, 40 is displayed			

For the input value of 4

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

For the input value of 1

Step	Statement	a	b	c	d
1	`cin >> a`	**1**	?	?	?
2	`a = (a + 1) * (a + 1) + 6 / 3 * 2 + 20`	**28**	?	?	?
3	`b = a % 13`	28	**2**	?	?
4	`c = b % 7`	28	2	**2**	?
5	`d = a * b * c`	28	2	2	**112**
6	`cout << a << ", " << b << ", " << c << ", " << d`	28, 2, 2, 112 is displayed			

2. Solution

For the input values of 3, 4

Step	Statement	a	b	c	d	e
1	`cin >> a`	**3**	?	?	?	?
2	`cin >> b`	3	**4**	?	?	?
3	`c = a + b`	3	4	**7**	?	?
4	`d = 1 + a / b * c + 2`	3	4	7	**8.25**	?
5	`e = c + d`	3	4	7	8.25	**15.25**
6	`c += d + e`	3	4	**30.5**	8.25	15.25
7	`e--`	3	4	30.5	8.25	**14.25**
8	`d -= c + (int)d % (int)c`	3	4	30.5	**-30.25**	14.25
9	`cout << c << ", " << d << ", " << e`	30.5, -30.25, 14.25 is displayed				

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 + (int)d % (int)c`	4	4	38	**-38**	18
9	`cout << c << ", " << d << ", " << e`	38, -38, 18 is displayed				

Chapter 9

9.5 Answers of 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	`cin >> a`	**5**	?	?	?	?
2	`cin >> b`	5	**5**	?	?	?
3	`c = a + b`	5	5	**10**	?	?
4	`d = 5 + a / b * c + 2`	5	5	10	**17**	?
5	`e = c - d`	5	5	10	17	**-7**
6	`c -= d + c`	5	5	**-17**	17	-7
7	`e--`	5	5	-17	17	**-8**
8	`d -= c + (int)a % (int)c`	5	5	-17	**29**	-8
9	`cout << c << ", " << d << ", " << e`	-17, 29, -8 is displayed				

For the input values of 4, 8

Step	Statement	a	b	c	d	e
1	`cin >> a`	**4**	?	?	?	?
2	`cin >> b`	4	**8**	?	?	?
3	`c = a + b`	4	8	**12**	?	?
4	`d = 5 + a / b * c + 2`	4	8	12	**13**	?
5	`e = c - d`	4	8	12	13	**-1**
6	`c -= d + c`	4	8	**-13**	13	-1
7	`e--`	4	8	-13	13	**-2**
8	`d -= c + (int)a % (int)c`	4	8	-13	**22**	-2
9	`cout << c << ", " << d << ", " << e`	-13, 22, -2 is displayed				

3. Solution

For the input value of 0.50

Step	Statement	a	b	c
1	`cin >> b`	?	**0.50**	?
2	`c = 5`	?	0.50	**5**

3	`c = c * b`	?	0.50	**2.5**
4	`a = (int)(10 * c) % 10`	**5**	0.50	2.5
5	`cout << a`	Value 5 is displayed		

For the input value of 3

Step	Statement	a	b	c
1	`cin >> b`	?	3	?
2	`c = 5`	?	3	**5**
3	`c = c * b`	?	3	**15**
4	`a = (int)(10 * c) % 10`	**0**	3	15
5	`cout << a`	Value 0 is displayed		

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 = (int)(10 * c) % 10`	**0**	15	75
5	`cout << a`	Value 0 is displayed		

Chapter 10

10.2 Answers of Review Exercises

1. Solution

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

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

  area = 0.5 * base * height;

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

4. Solution

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

int main() {
  double r, perimeter;

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

  perimeter = 2 * PI * r;

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

5. 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;
  return 0;
}
```

6. 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;
  return 0;
}
```

7. 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;
  return 0;
}
```

8. 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;
  return 0;
}
```

9. 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: ";
  cin >> g_rate;

  tip = s_total * g_rate / 100;

  total = s_total + tip;

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

10. 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;
  return 0;
}
```

11. 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;
  return 0;
}
```

12. 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;
  return 0;
}
```

13. 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;
  return 0;
}
```

14. Solution

```
#include <iostream>
```

```
using namespace std;
int main() {
  int day, month, days_passed, days_left;

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

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

  cout << days_left;
  return 0;
}
```

Chapter 11

11.3 Answers of Review Questions: True/False

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

11.4 Answers of Review Questions: Multiple Choice

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

11.5 Answers of Review Exercises

1. Solution

For the input value of 9

Step	Statement	a	b	c
1	`cin >> a`	**9**	?	?
2	`a += 6 / sqrt(a) * 2 + 20`	**33**	?	**?**
3	`b = round(a) % 4`	33	**1**	**?**
4	`c = b % 3`	33	1	**1**
5	`cout << a << ", " << b << ", " << c`	33, 1, 1 is displayed		

For the input value of 4

Step	Statement	a	b	c
1	`cin >> a`	**4**	?	?
2	`a += 6 / sqrt(a) * 2 + 20`	**30**	?	**?**
3	`b = round(a) % 4`	30	**2**	**?**
4	`c = b % 3`	30	2	**2**
5	`cout << a << ", " << b << ", " << c`	30, 2, 2 is displayed		

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`	18, 3 is displayed

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`	84, 4 is displayed		

3. Solution

```
#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;
  return 0;
}
```

4. Solution

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

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

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

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

5. Solution

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

```
  double adjacent, opposite, th;

  cout << "Enter angle (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;
  return 0;
}
```

Chapter 12

12.2 Answers of Review Exercises

1. Solution

i. a, e, g, h

ii. c, f

2. Solution

i. `y = pow(x + 3, 5 * w) / (7 * (x - 4))`

ii. `y = pow(3 * pow(x, 2) - pow(x, 3) / 4, 1 / 5.0)`

iii. `y = sqrt(pow(x, 4) - 2 * pow(x, 3) - 7 * x * x +x) / pow(4 * (7 * pow(x, 4) - 3 / 4.0 * pow(x, 3)) * (7 * x * x + x), 1/3.0)`

iv. `y = x / (x - 3 * (x - 1)) + x * pow(x - 1, 1 / 5.0) / ((pow(x, 3) - 2) * pow(x - 1, 3))`

v. `y = pow(sin(M_PI / 3) - cos(M_PI / 2 * w) , 2)`

vi. `y = pow(sin(M_PI / 2 * x) + cos(3 * M_PI / 2 * w), 3) / pow(tan(2 * M_PI / 3 * w) - sin(M_PI / 2 * x), 0.5)`

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;
  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;
```

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

Chapter 13

13.2 Answers of 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;
  return 0;
}
```

2. Solution

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

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

5. Solution

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

  cout << "Enter a period of 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 << " weeks " << days << " days " << hours << " hours ";
  cout << minutes << " minutes and " << seconds << " seconds";
  return 0;
```

```
}
```

6. Solution

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

  cout << "Enter amount 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 << " notes of $20 " << usd10 << " notes of $10 ";
  cout << usd5 << " notes of $5 and " << usd1 << " notes of $1";
  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 << " miles " << yards << " yards ";
  cout << feet << " feet and " << inches << " inches";
  return 0;
}
```

Chapter 14

14.5 Answers of Review Questions: True/False

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

14.6 Answers of Review Questions: Multiple Choice

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

14.7 Answers of 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;
  return 0;
}
```

2. Solution

```
#include <iostream>
#include <ctime>
#include <cstdlib>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
#include <cctype>

int main() {
  string alphabet_lower, alphabet_upper;

  srand(time(NULL));

  alphabet_lower = "abcdefghijklmnopqrstuvwxyz";
  alphabet_upper = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";

  cout << alphabet_upper.at(rand() % 26) <<
          alphabet_lower.at(rand() % 26) <<
          alphabet_lower.at(rand() % 26) <<
          alphabet_lower.at(rand() % 26) <<
          alphabet_lower.at(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;
  srand(time(NULL));

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

  name = to_lower_copy(name);

  cout <<  name.at(rand() % name.length()) <<
           name.at(rand() % name.length()) <<
           name.at(rand() % name.length()) <<
           (1000 + rand() % (9999 - 1000 + 1));

  return 0;
}
```

Chapter 15

15.8 Answers of Review Questions: True/False

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

15.9 Answers of Review Questions: Multiple Choice

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

15.10 Answers of Review Exercises

1. Solution

i. c, e, g
ii. a
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
`pow(x + y, 3)`	8
`(x + y) / (pow(x, 2) - 14)`	1
`x - 1 == y + 5`	true
`x > 2 && y == 1`	false
`x == 1 || y == -2 && !(flag == false)`	true
`!(x >= 3) && (x % 2 > 1)`	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 Answers of Review Questions: True/False

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

16.3 Answers of Review Questions: Multiple Choice

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

16.4 Answers of 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--;
    x -= 4 * x * x;
  }

  cout << x + y;
  return 0;
}
```

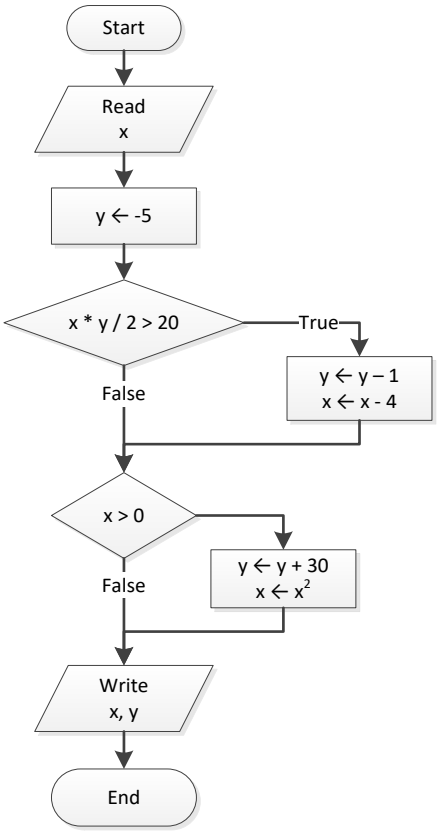
2. Solution

For the input value of 10

Step	Statement	x	y
1	`cin >> x`	**10**	?
2	`y = - 5`	10	**-5**
3	`if (x * y / 2 > 20)`	False	
4	`if (x > 0)`	True	
5	`y += 30`	10	**25**
6	`x = pow(x, 2)`	**100**	25
7	`cout << x << ", " << y`	100, 25 is displayed	

For the input value of -10

Step	Statement	x	y
1	`cin >> x`	**-10**	?
2	`y = - 5`	-10	**-5**
3	`if (x * y / 2 > 20)`	True	
4	`y--`	-10	**-6**
5	`x -= 4`	**-14**	-6
6	`if (x > 0)`	False	
7	`cout << x << ", " << y`	-14, -6 is displayed	



3. Solution

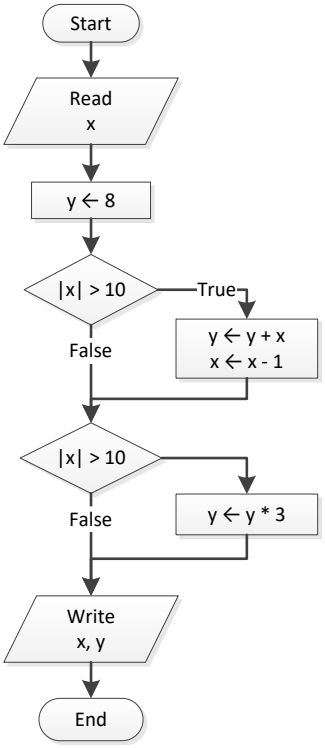
For the input value of -11

Step	Statement	x	y
1	`cin >> x`	**-11**	?
2	`y = 8`	-11	**8**
3	`if (abs(x) > 10)`	True	
4	`y += x`	-11	**-3**
5	`x--`	**-12**	-3

6	`if (abs(x) > 10)`	True	
7	`y *= 3`	-12	**-9**
8	`cout << x << ", " << y`	-12, -9 is displayed	

For the input value of 11

Step	Statement	x	y
1	`cin >> x`	**11**	?
2	`y = 8`	11	**8**
3	`if (abs(x) > 10)`	True	
4	`y += x`	11	**19**
5	`x--`	**10**	19
6	`if (abs(x) > 10)`	False	
7	`cout << x << ", " << y`	10, 19 is displayed	



4. *Solution*

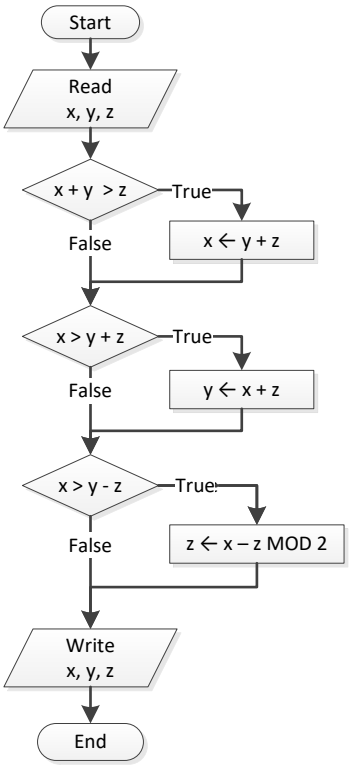
For input values of 1, 2 and 3

Step	Statement	x	y	z
1	`cin >> x`	**1**	?	?
2	`cin >> y`	1	**2**	?
3	`cin >> z`	1	**2**	**3**
4	`if (x + y > z)`	False		

5	`if (x > y + z)`		False	
6	`if (x > y - z)`		True	
7	`z = x - z % 2`	1	2	**0**
8	`cout << x << ", " << y << ", " << z`	1, 2, 0 is displayed		

For input values of 4, 2 and 1

Step	Statement	x	y	z
1	`cin >> x`	**4**	?	?
2	`cin >> y`	4	**2**	?
3	`cin >> z`	4	**2**	**1**
4	`if (x + y > z)`		True	
5	`x = y + z`	**3**	2	1
6	`if (x > y + z)`		False	
7	`if (x > y - z)`		True	
8	`z = x - z % 2`	3	2	**2**
9	`cout << x << ", " << y << ", " << z`	3, 2, 2 is displayed		



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() {
  string x;

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

  if (cin.fail() != true) {
    cout << "Numeric" << endl;
  }

  //Clear the error state of cin
  cin.clear();
  cin.ignore(100, '\n');

  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;
  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 Answers of Review Questions: True/False

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

17.3 Answers of Review Questions: Multiple Choice

1. b
2. c
3. c

17.4 Answers of Review Exercises

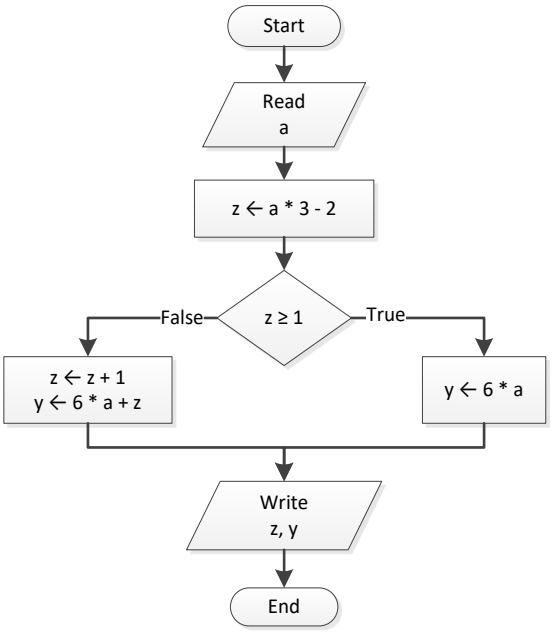
1. Solution

For input value of 3

Step	Statement	a	y	z
1	`cin >> a`	**3**	?	?
2	`z = a * 3 - 2`	3	?	**7**
3	`if (z >= 1)`		True	
4	`y = 6 * a`	3	**18**	7
5	`cout << z << ", " << y`	7, 18 is displayed		

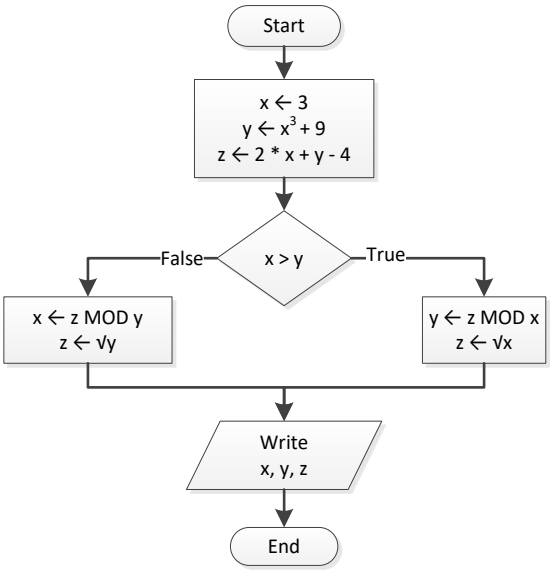
For input value of 0.5

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`	0.5, 3.5 is displayed		



2. Solution

Step	Statement	x	y	z
1	`x = 3`	**3**	?	?
2	`y = pow(x, 3) + 9`	3	**36**	?
3	`z = 2 * x + y - 4`	3	36	**38**
4	`if (x > y)`		False	
5	`x = z % y`	**2**	36	38
6	`z = sqrt(y)`	2	36	**6**
7	`cout << x << ", " << y << ", " << z`	2, 36, 6 is displayed		



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;
  return 0;
}
```

For input value of 10

Step	Statement	x	y	w	z
1	cin >> x	**10**	?	?	?
2	w = x * 3 - 15	10	?	**15**	?
3	z = (w + 7) * (x + 4) - 10	10	?	15	**298**
4	if (w > x && z > x)	True			
5	x++	**11**	?	15	298
6	y = x / 2 + 4	11	**9.5**	15	298
7	cout << y	9.5 is displayed			

For input value of 2

Step	Statement	x	y	w	z
1	`cin >> x`	**2**	?	?	?
2	`w = x * 3 - 15`	2	?	**-9**	?
3	`z = (w + 7) * (x + 4) - 10`	2	?	-9	**-22**
4	`if (w > x && z > x)`	False			
5	`y = x / 4 + 2`	2	**2.5**	-9	-22
6	`cout << y`	2.5 is displayed			

4. 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;
}
```

5. 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;
}
```

6. Solution

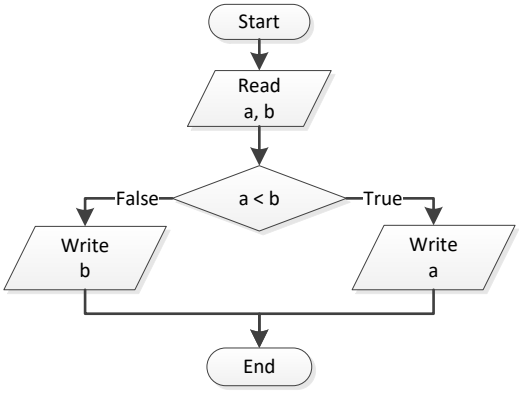
```
#include <iostream>
using namespace std;
int main() {
  int x, y;
```

```
  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;
  return 0;
}
```

7. 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;
}
```

8. Solution




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

  cin >> a;
  cin >> b;
```

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

9. Solution

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

  cin >> a;
  cin >> b;
  cin >> 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;
}
```

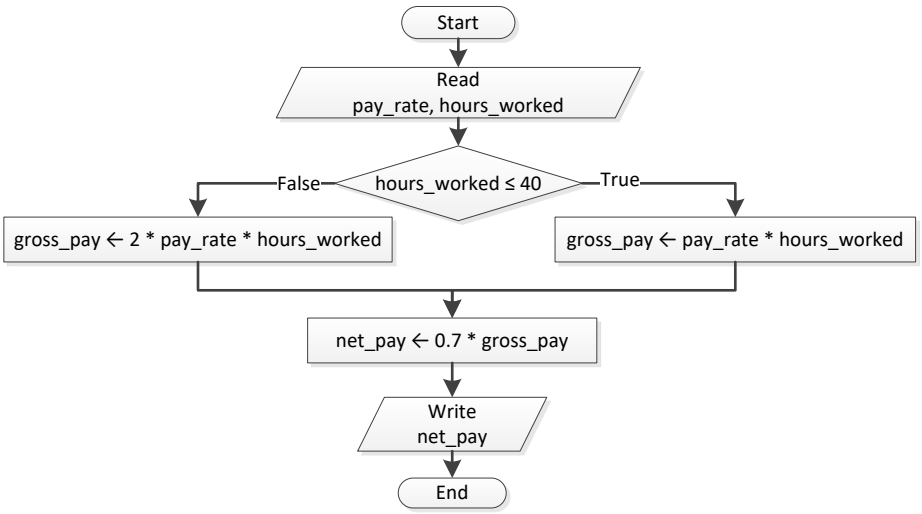
10. Solution

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

  cin >> a;
  cin >> b;
  cin >> 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;
}
```

11. 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 << "Disqualified" << endl;
  }
  else {
    cout << "Qualified" << endl;
  }
  return 0;
}
```

12. Solution




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

  cin >> pay_rate;
  cin >> hours_worked;

  if (hours_worked <= 40) {
```

```
    gross_pay = pay_rate * hours_worked;
  }
  else {
    gross_pay = 2 * pay_rate * hours_worked;
  }

  net_pay = 0.7 * gross_pay;
  cout << "Net Pay: " << net_pay;
  return 0;
}
```

13. 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;
}
```

14. Solution

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

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

  s1 = 0.5 * a1 * t;
  s2 = 0.5 * a2 * t;

  if (s1 > s2) {
    cout << "Car A is first" << endl;
```

```
  }
  else {
    cout << "Car B is first" << endl;
  }
  return 0;
}
```

Chapter 18

18.2 Answers of Review Questions: True/False

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

18.3 Answers of 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`	1 is displayed	

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`	3 is displayed	

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`	4 is displayed	

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`	4 is displayed	

2. *Solution*

For input value of 5

Step	Statement	amount	discount	payment
1	`cin >> amount`	**5**	?	?
2	`discount = 0`	5	**0**	?
3	`if (amount < 20)`	True		
4	`discount = 0`	5	**0**	?
5	`payment = amount - amount * discount / 100`	5	0	**5**
6	`cout << discount << ", " << payment`	0, 5 is displayed.		

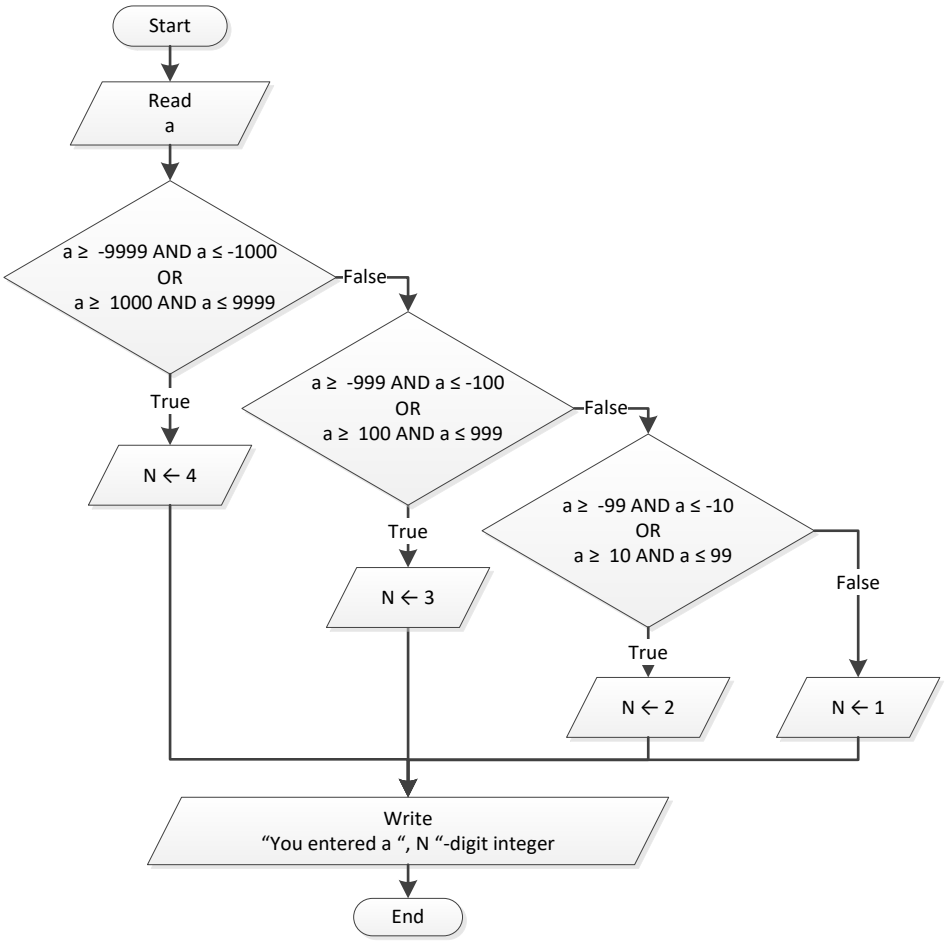
For input value of 150

Step	Statement	amount	discount	payment
1	`cin >> amount`	**150**	?	?
2	`discount = 0`	150	**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	**15**	?
8	`payment = amount - amount * discount / 100`	150	15	**5**
9	`cout << discount << ", " << payment`	15, 127.5 is displayed.		

For input value of -1

Step	Statement	amount	discount	payment
1	`cin >> amount`	**-1**	?	?
2	`discount = 0`	-1	**0**	?
3	`if (amount < 20)`	True		
4	`discount = 0`	-1	**0**	?
5	`payment = amount - amount * discount / 100`	-1	0	**-1**
6	`cout << discount << ", " << payment`	0, -1 is displayed.		

3. Solution




```
#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";
```

```
  return 0;
}
```

4. 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 / 0.72;
    cout << "$" << usd << " = " << eur << " EUR" << endl;
  }
  else if (ch == 2) {
    gbp = usd / 0.60;
    cout << "$" << usd << " = " << gbp << " GBP" << endl;
  }
  else if (ch == 3) {
    jpy = usd / 102.15;
    cout << "$" << usd << " = " << jpy << " JPY" << endl;
  }
  else {
    cad = usd / 1.10;
    cout << "$" << usd << " = " << cad << " CAD" << endl;
  }
  return 0;
}
```

5. 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;
```

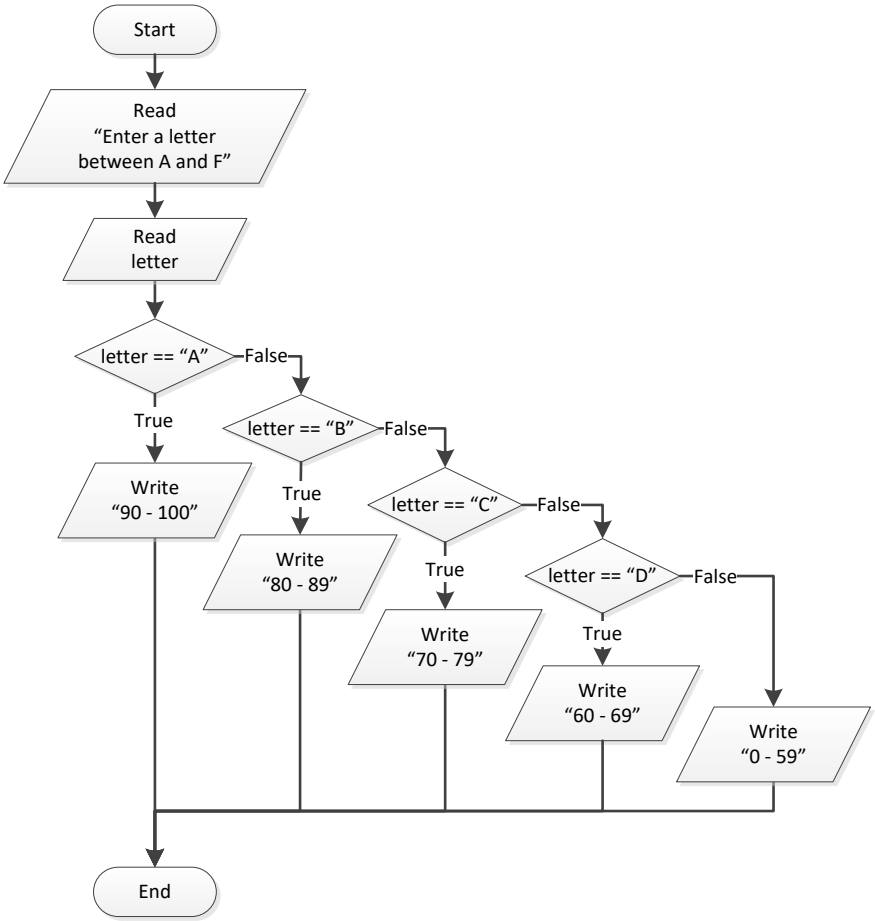
```
  }
  else if (m <= 5) {
    cout << "Spring" << endl;
  }
  else if (m <= 8) {
    cout << "Summer" << endl;
  }
  else {
    cout << "Fall (Autumn)" << endl;
  }
  return 0;
}
```

6. 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;
}
```

7. 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 {
    cout << "0 - 59" << endl;
  }
  return 0;
}
```

Chapter 19

19.2 Answers of Review Questions: True/False

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

19.3 Answers of 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`	5, 5 is displayed		

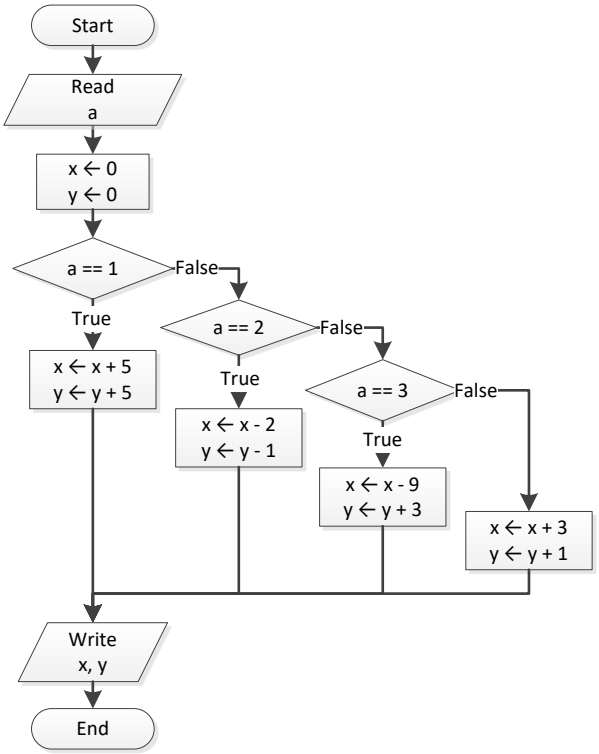
For input value of 3

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`	-9, 3 is displayed		

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	3, 1 is displayed		



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**
4	case a == 10	True		
5	x = x % 2	10	**0**	5
6	y = pow(y, 2)	10	0	**25**
7	cout << x << ", " << y	0, 25 is displayed		

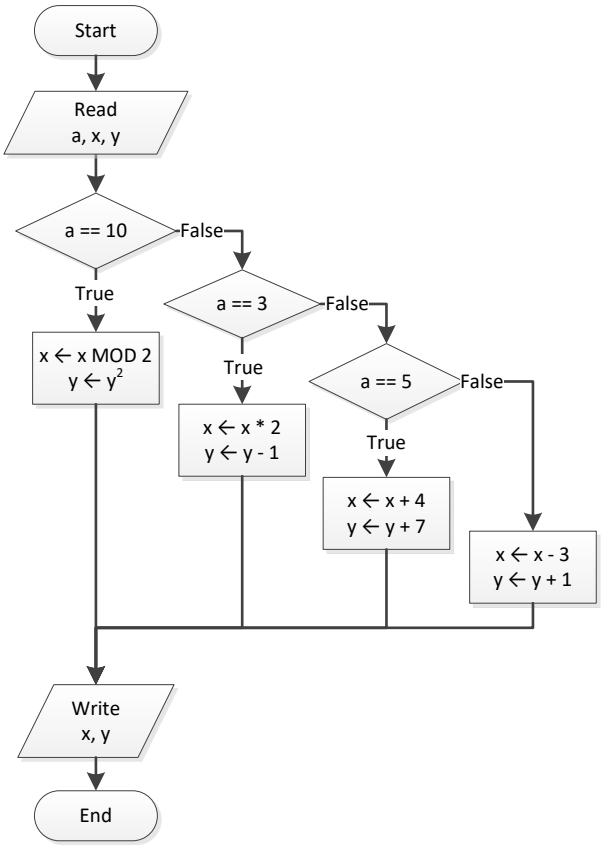
For input values of 5, 2, 3

Step	Statement	a	x	y
1	cin >> a	**5**	?	?
2	cin >> x	5	**2**	?

3	cin >> y	5	2	**3**
4	case a == 10		False	
5	case a == 3		False	
6	case a == 5		True	
7	x = x + 4	5	**6**	3
8	y += 7	5	6	**10**
9	cout << x << ", " << y	6, 10 is displayed		

For input values of 4, 6, 2

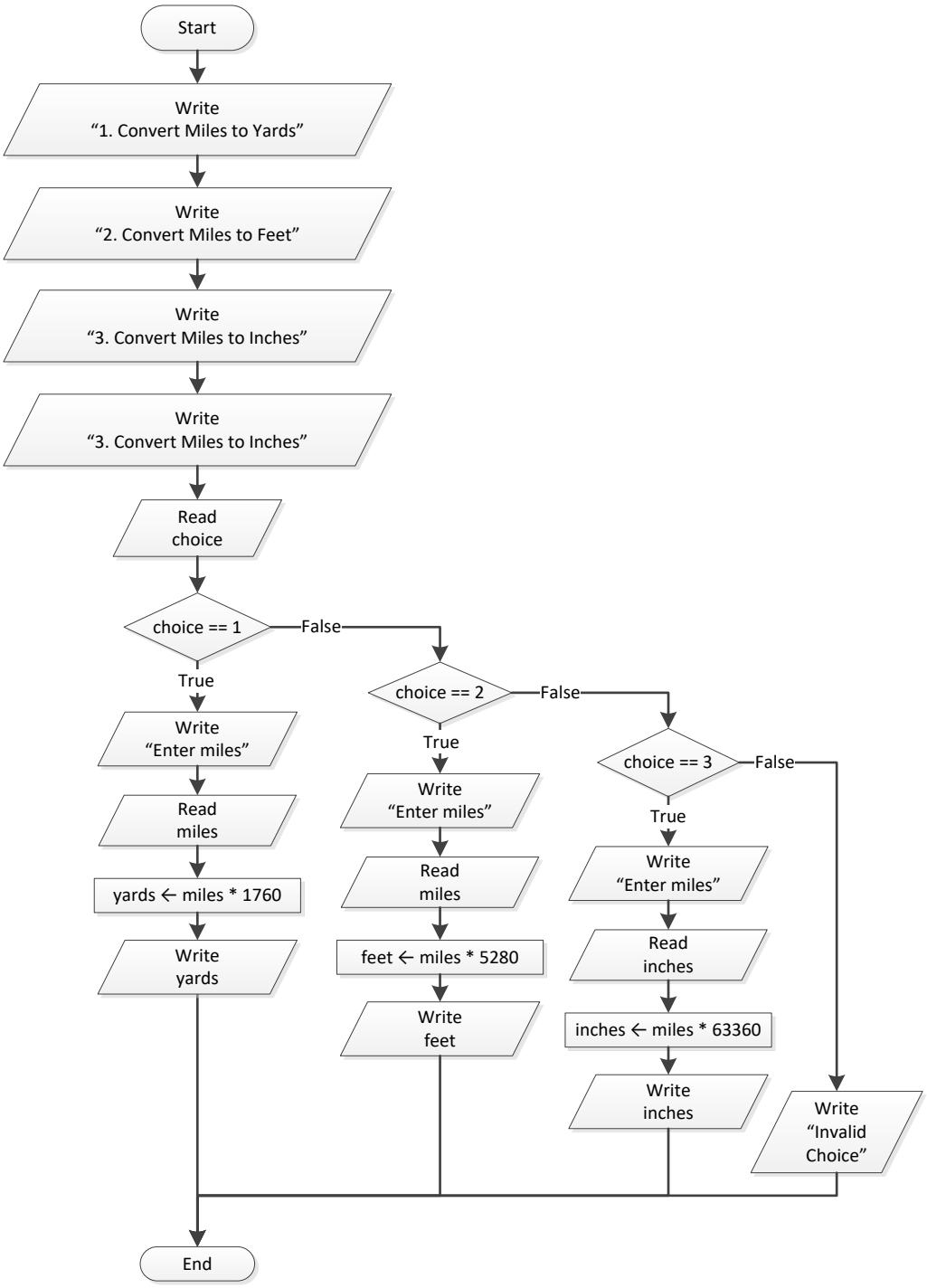
Step	Statement	a	x	y
1	cin >> a	**4**	?	?
2	cin >> x	4	**6**	?
3	cin >> y	4	6	**2**
4	case a == 10		False	
5	case a == 3		False	
6	case a == 5		False	
7	x -= 3	4	**3**	2
8	y++	4	3	**3**
9	cout << x << ", " << y	3, 3 is displayed		



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 Answers of Review Questions: True/False

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

20.4 Answers of 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`	2, 5 is displayed	

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`	25, 6 is displayed	

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`	10, 9 is displayed	

For input values of 50, 0

Step	Statement	x	y
1	`cin >> x`	**50**	?
2	`cin >> y`	50	**0**
3	`y++`	50	**1**
4	`cout << x << ", " << y`	50, 1 is displayed	

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`	125, 26 is displayed	

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`	2, 2 is displayed	

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`	100, 17 is displayed	

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`	8, 33 is displayed	

3. Solution

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

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

4. 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 << " notes of $10 " << usd5 << " notes of $5 ";
    cout << "and " << usd1 << " notes of $1" << endl;
  }
  return 0;
}
```

5. 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;
  return 0;
}
```

Chapter 21

21.13 Answers of Review Questions: True/False

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

21.14 Answers of Review Questions: Multiple Choice

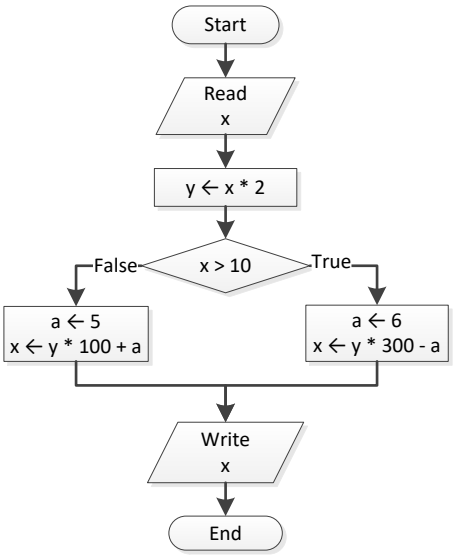
1. c
2. b
3. a
4. a
5. b
6. c
7. c
8. d

21.15 Answers of 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;
  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;
  cin >> month;
  cin >> 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 `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;
  cin >> b;
  cin >> 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;
  cin >> b;
  cin >> 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;
  return 0;
}
```

8. Solution

```
#include <iostream>
#include <cmath>
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;
  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;
}
```

10. Solution

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

  cin >> a;
  cin >> x;
  cin >> y;

  if (a == 3) {
    x = x / 4;
    y = pow(y, 5);
  }
  else if (a == 7) {
    x = x * 3;
    y++;
  }
  else if (a == 22) {
    x = x % 4;
    y += 9;
  }
  else {
    x -= 9;
    y++;
  }
```

```
  cout << x << ", " << y;
  return 0;
}
```

11. Solution

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

  cin >> a;
  cin >> x;
  cin >> y;

  if (a == 3) {
    x = x / 4;
    y = pow(y, 5);
  }
  else {
    if (a == 7) {
      x = x * 3;
      y++;
    }
    else {
      if (a == 22) {
        x = x % 4;
        y += 9;
      }
      else {
        x -= 9;
        y++;
      }
    }
  }

  cout << x << ", " << y;
  return 0;
}
```

12. Solution

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

  cout << "1. Red" << endl;
  cout << "2. Green" << endl;
  cout << "3. Blue" << endl;
```

```
  cout << "4. White" << endl;
  cout << "5. Black" << endl;
  cout << "6. Gray" << endl;
  cout << "Select a color: ";
  cin >> color;

  cout << "Your color in hexadecimal is: ";

  switch (color) {
    case 1:
      cout << "FF0000" << endl;
      break;
    case 2:
      cout << "00FF00" << endl;
      break;
    case 3:
      cout << "0000FF " << endl;
      break;
    case 4:
      cout << "FFFFFF " << endl;
      break;
    case 4:
      cout << "000000" << endl;
      break;
    case 6:
      cout << "7F7F7F " << endl;
      break;
    default:
      cout << "Unknown color!" << endl;
  }
  return 0;
}
```

13. Solution

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

  cout << "1. Red" << endl;
  cout << "2. Green" << endl;
  cout << "3. Blue" << endl;
  cout << "4. White" << endl;
  cout << "5. Black" << endl;
  cout << "6. Gray" << endl;
  cout << "Select a color: ";
  cin >> color;

  cout << "Your color in hexadecimal is: ";

  if (color == 1) {
    cout << "FF0000" << endl;
  }
```

```
  else {
    if (color == 2) {
      cout << "00FF00" << endl;
    }
    else {
      if (color == 3) {
        cout << "0000FF " << endl;
      }
      else {
        if (color == 4) {
          cout << "FFFFFF " << endl;
        }
        else {
          if (color == 5) {
            cout << "000000" << endl;
          }
          else {
            if (color == 6) {
              cout << "7F7F7F " << endl;
            }
            else {
              cout << "Unknown color!" << endl;
            }
          }
        }
      }
    }
  }
  return 0;
}
```

14. 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;
}
```

```
#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";
  return 0;
}
```

15. Solution

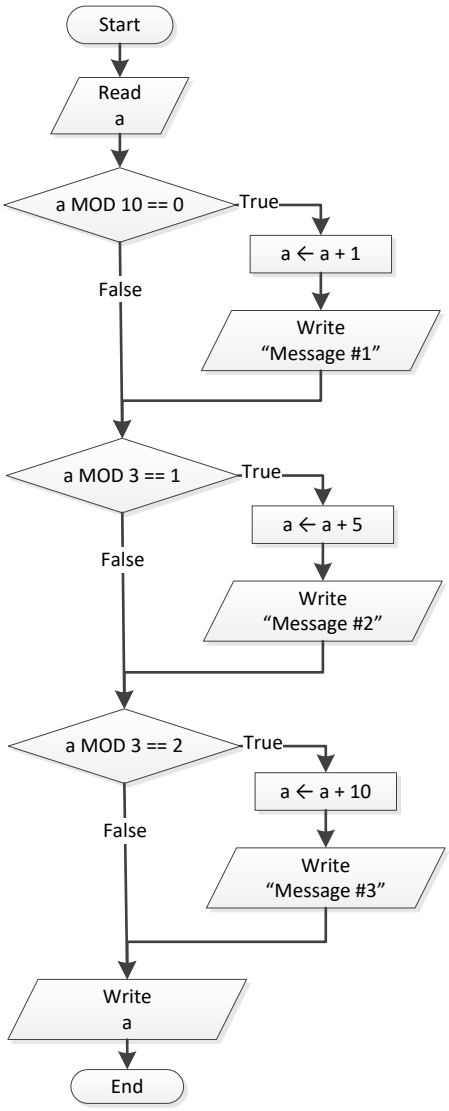
```
#include <iostream>
using namespace std;
int main() {
  double a, y;

  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;
}
```

```
#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;
}
```

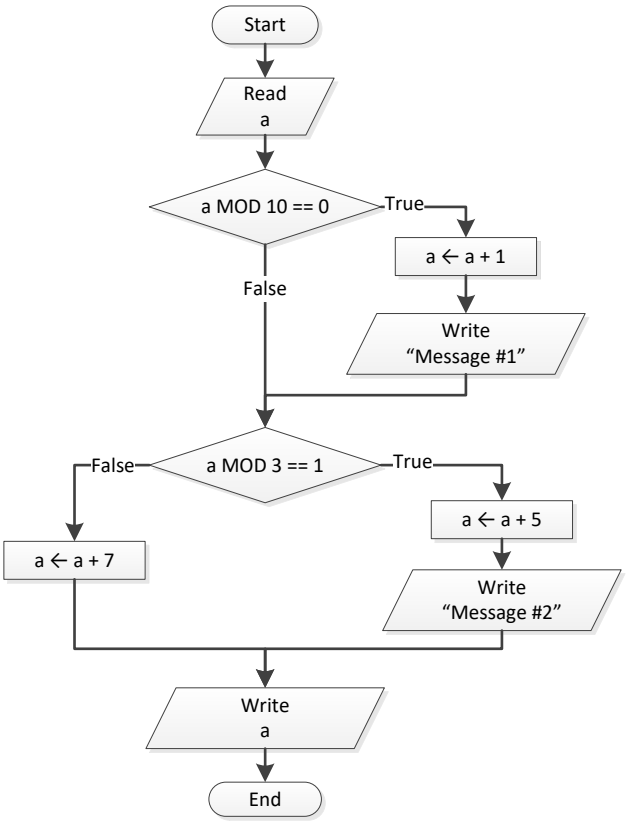
Chapter 22

22.4 Answers of Review Exercises

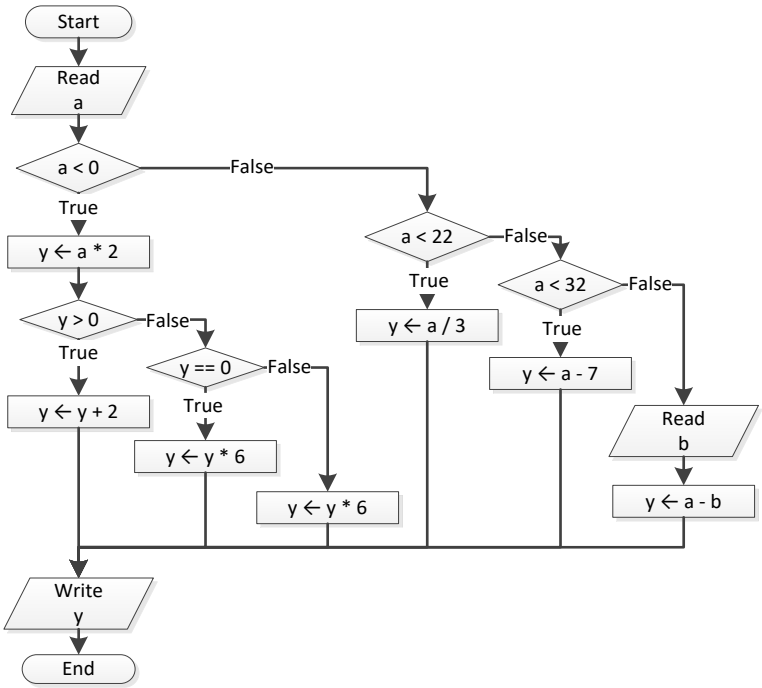
1. Solution



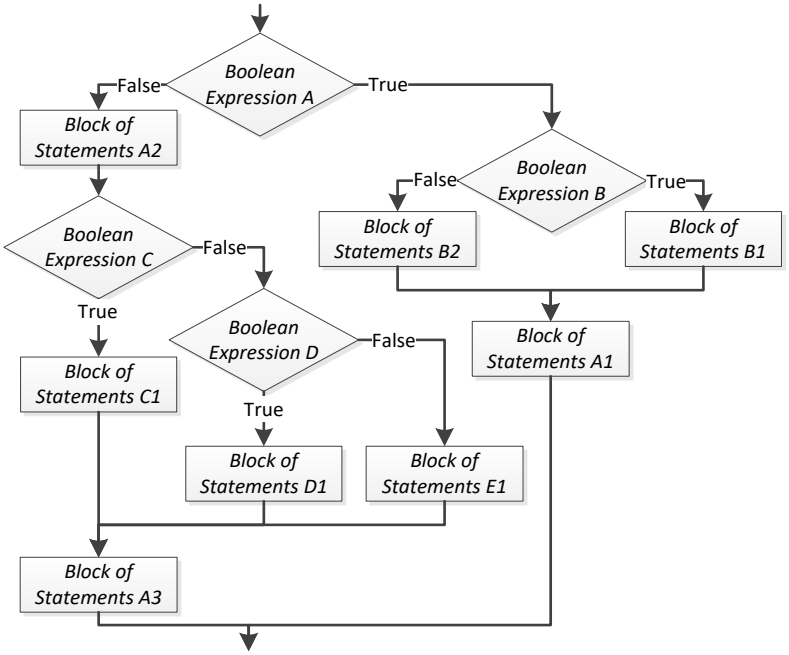
2. Solution



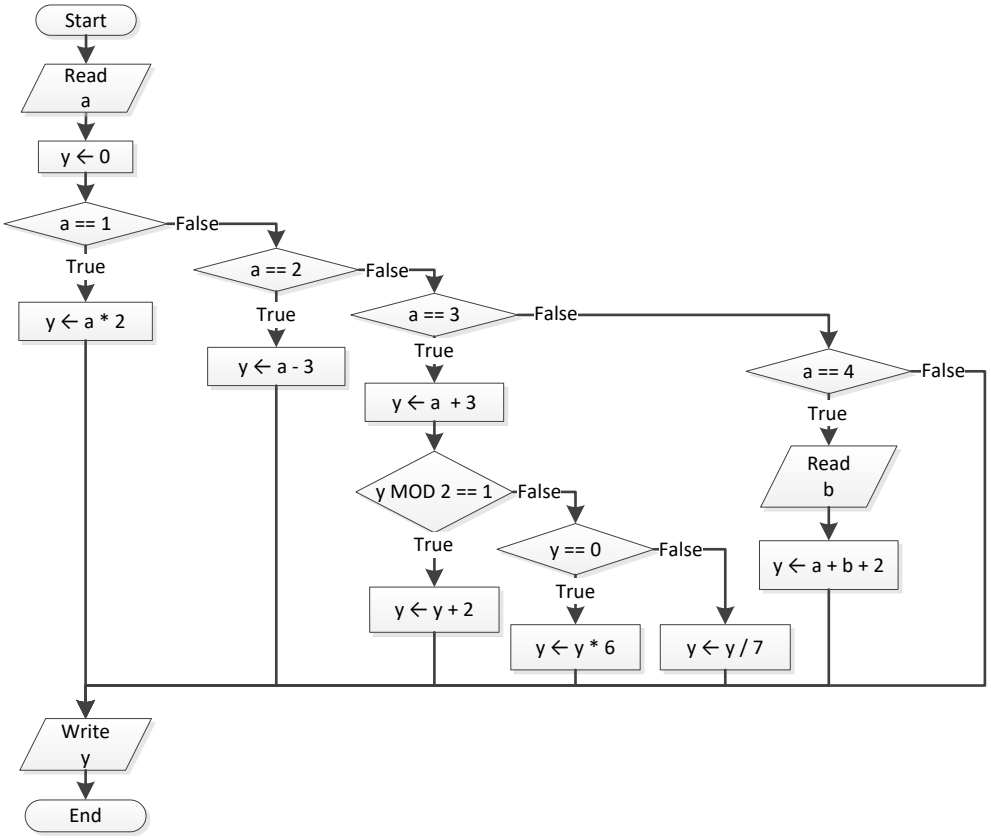
3. Solution



4. Solution



5. Solution



6. Solution

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

  cin >> x;
  cin >> y;

  if (x != 100 || y <= 10) {
    cin >> z;
    if (z <= x + y) {
      x -= 3;
      y = x + 4;
    }
  }
  cout << x << ", " << y;
  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 x;
  string x_str;

  cin >> x;
  if (cin.fail() != true) {
    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 {
    cin.clear();
    //Do not ignore data input using the cin.ignore(100, 'n') statement.
    //Send data input to x_str variable instead.
    cin >> x_str;

    if (x_str == "Exit") {
      cout << "Bye" << endl;
    }
    else {
      cout << "Invalid Number" << endl;
    }
  }
  return 0;
}
```

9. Solution

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

  cin >> a;
  cin >> b;

  y = a * b;

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

```
    }
    else {
      y *= 2;
    }
  }
  return 0;
}
```

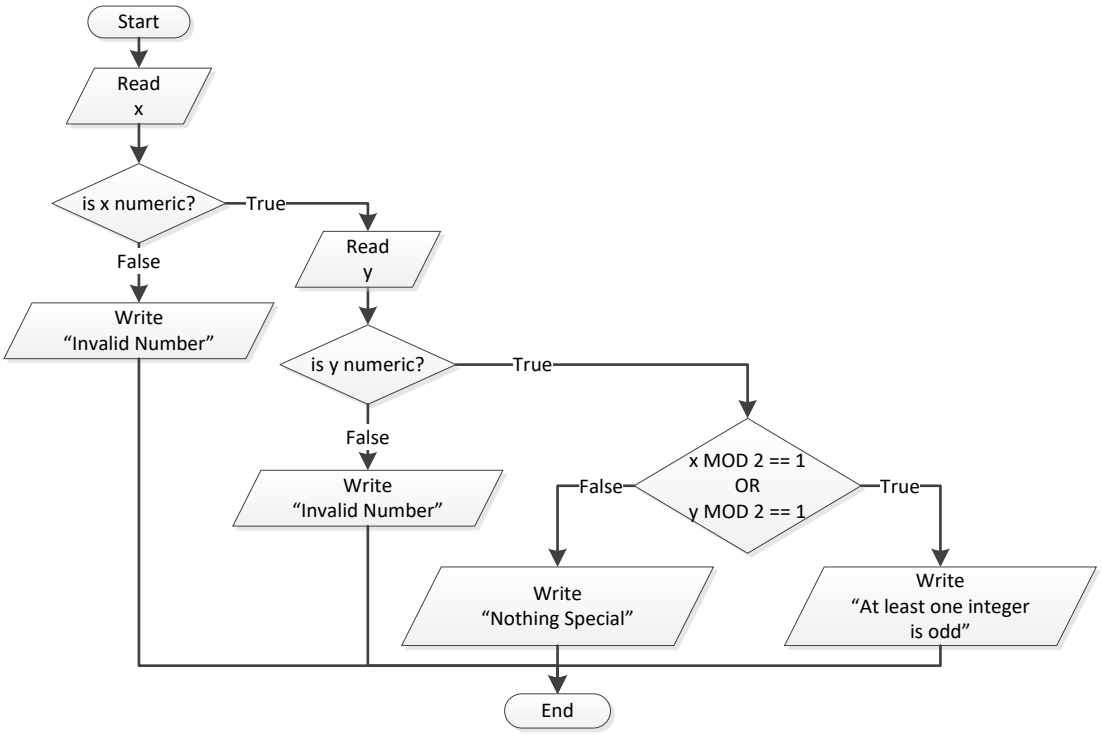
10. Solution

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

  cin >> a;
  cin >> b;
  cin >> 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;
  return 0;
}
```

Chapter 23

23.6 Answers of Review Exercises

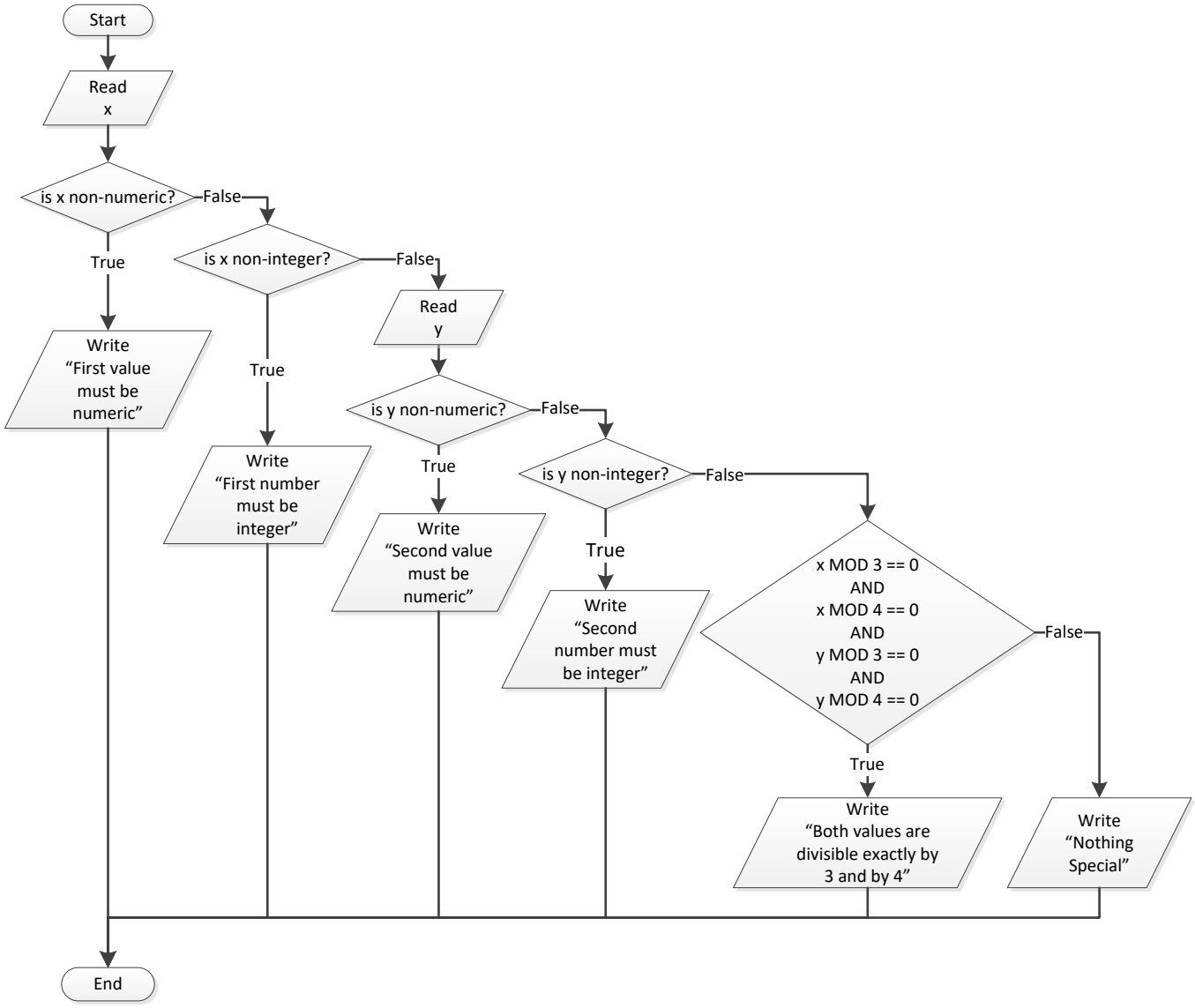
1. Solution




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

int main() {
  int x, y;

  cin >> x;

  if (cin.fail() != true) {
    cin >> y;
    if (cin.fail() != true) {
      if (x % 2 == 1 || y % 2 == 1) {
        cout << "At least one integer is odd" << endl;
      }
      else {
        cout << "Nothing Special" << endl;
      }
    }
    else {
      cin.clear();
      cin.ignore(100, '\n');
      cout << "Invalid Number" << endl;
    }
```

```
  }
  else {
    cin.clear();
    cin.ignore(100, '\n');
    cout << "Invalid Number" << endl;
  }

  return 0;
}
```

2. Solution




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

int main() {
  double x, y;
```

```
  cin >> x;

  if (cin.fail() == true) {
    cin.clear();
    cin.ignore(100, '\n');
    cout << "First value must be numeric" << endl;
  }
  else if (x != (int)x) {
    cin.clear();
    cin.ignore(100, '\n');
    cout << "First number must be integer" << endl;
  }
  else {
    cin >> y;
    if (cin.fail() == true) {
      cin.clear();
      cin.ignore(100, '\n');
      cout << "Second value must be numeric" << endl;
    }
    else if (y != (int)y) {
      cout << "Second number must be integer" << endl;
    }
    else {
      if ((int)x % 3 == 0 && (int)x % 4 == 0 &&
          (int)y % 3 == 0 && (int)y % 4 == 0 ) {

        cout << "Both values are divisible exactly by 3 and by 4" << endl;
      }
      else {
        cout << "Nothing Special" << endl;
      }
    }
  }
  return 0;
}
```

3. 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: ";
```

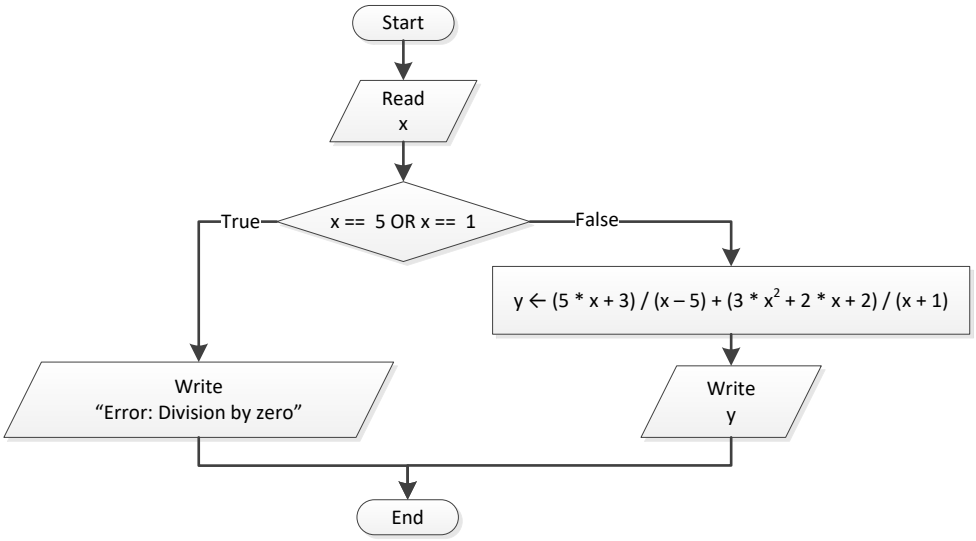
```
  cin >> t;

  if (choice < 1 || choice > 4) {
    cout << "Wrong choice" << endl;
  }
  else if (cin.fail() == true) {
    cin.clear();
    cin.ignore(100, '\n');
    cout << "Wrong temperature" << endl;
  }
  else {
    switch (choice) {
      case 1:
        cout << 1.8 * t - 459.67 << endl;
        break;
      case 2:
        cout << (t + 459.57) / 1.8 << endl;
        break;
      case 3:
        cout << 5 / 9 * (t - 32) << endl;
        break;
      case 4:
        cout << 9 / 5 * t + 32 << endl;
        break;
    }
  }
  return 0;
}
```

4. 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;
}
```

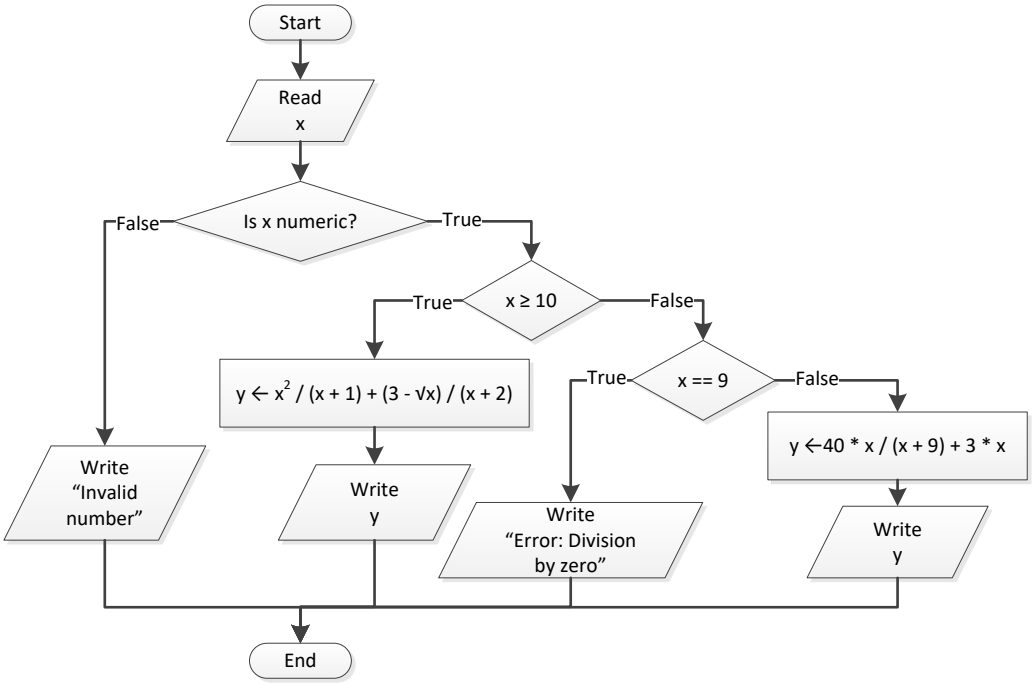
5. Solution




```
#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;
}
```

6. Solution




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

int main() {
  double x, y;

  cin >> x;

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

```
        y = 40 * x / (x + 9) + 3 * x;
        cout << y << endl;
      }
    }
  }
  else {
    cin.clear();
    cin.ignore(100, '\n');
    cout << "Invalid number" << endl;
  }
  return 0;
}
```

7. 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;
}
```

8. Solution

```
#include <iostream>
```

```
using namespace std;
int main() {
  int age1, age2, age3, max, middle, min;

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

  min = age1;
  if (age2 < min) {
    min = age2;
  }
  if (age3 < min) {
    min = age3;
  }
  max = age1;
  if (age2 > max) {
    max = age2;
  }
  if (age3 > max) {
    max = age3;
  }

  middle = age1 + age2 + age3 - min - max;
  cout << middle;
  return 0;
}
```

9. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
  int a1, a2, a3, max, middle, min;
  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;

  min = a1;
```

```
  min_name = n1;
  if (a2 > min) {
    min = a2;
    min_name = n2;
  }
  if (a3 > min) {
    min = a3;
    min_name = n3;
  }

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

  middle = a1 + a2 + a3 - min - max;

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

10. Solution

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

int main() {
  int digit1, digit2, digit3, r, x, sum;

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

  if (cin.fail() == true) {
    cin.clear();
    cin.ignore(100, '\n');
    cout << "Entered value contains non-numeric characters" << endl;
  }
  else if (x < 100 || x > 999) {
    cout << "Entered integer is not a three-digit integer" << endl;
  }
  else {
    digit1 = (int)(x / 100);
```

```
    r = x % 100;

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

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

    if (sum == x) {
      cout << "You entered an Armstrong number!" << endl;
    }
    else {
      cout << "You entered a non-Armstrong number!" << endl;
    }
  }
  return 0;
}
```

11. 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;
}
```

12. Solution

```
#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
```

```
using namespace std;
int main() {
  string word, word1, word2;

  cin >> word;

  //Using substr() instead of at() method
  //is more convenient in this case
  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;
}
```

13. 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;
  return 0;
}
```

14. Solution

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

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

  cout << "Enter total amount: ";
  cin >> amount;

  if (cin.fail() == true) {
    cin.clear();
    cin.ignore(100, '\n');
    cout << "Entered value contains non-numeric characters" << endl;
  }
  else if (amount < 0) {
    cout << "Entered non-negative value" << endl;
  }
  else {
    if (amount < 50) {
      discount = 0;
    }
    else if (amount < 100) {
      discount = 1;
    }
    else if (amount < 250) {
      discount = 2;
    }
    else {
      discount = 3;
    }

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

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

```
  return 0;
}
```

15. 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;
}
```

16. 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 (cin.fail() == true) {
    cin.clear();
    cin.ignore(100, '\n');
    cout << "Entered value contains non-numeric characters" << endl;
  }
  else if (water < 0) {
    cout << "Entered value is negative" << 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;
}
```

17. 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;
  return 0;
}
```

18. Solution

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

int main() {
  double wind;

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

  if (cin.fail() == true) {
    cin.clear();
    cin.ignore(100, '\n');
    cout << "Entered value contains non-numeric characters" << endl;
  }
  else if (wind < 0) {
    cout << "Entered value is negative" << 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;
    }
    else {
      cout << "Beaufort: 12\nHurricane force" << endl;
    }

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

Chapter 24

24.3 Answers of Review Questions: True/False

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

Chapter 25

25.2 Answers of Review Questions: True/False

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

25.3 Answers of Review Questions: Multiple Choice

1. c
2. b
3. d
4. b
5. c
6. b

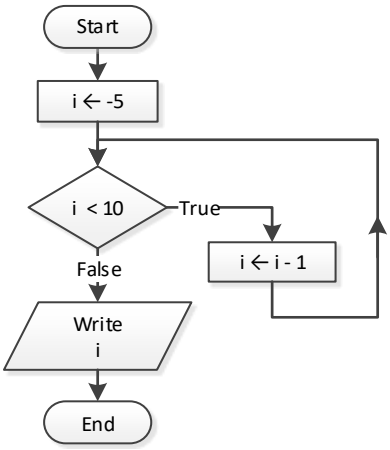
25.4 Answers of Review Exercises

1. 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`	2 is displayed	

It performs 3 iterations

2. 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

3. 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`	3, 6 is displayed			
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`	6, 7 is displayed			
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`	10, 11, 12 is displayed			
26	`a += 4`	**14**	11	22	12
27	`while (a <= 10)`	False			

4. 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				

5. Solution

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

6. Solution

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

  cin >> n;
  sum = 0;

  i = 1;
  while (i <= n) {
    cin >> a;
    sum = sum + a;
    i++;
  }

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

7. Solution

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

  cin >> n;
  p = 1;

  i = 1;
  while (i <= n) {
    cin >> a;
    if (a % 2 == 0) {
```

```
      p = p * a;
    }
    i++;
  }
  cout << p;
  return 0;
}
```

8. Solution

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

  sum = 0;

  i = 1;
  while (i <= 100) {
    cin >> a;
    if (a % 10 == 0) {
      sum = sum + a;
    }
    i++;
  }
  cout << sum;
  return 0;
}
```

9. Solution

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

  sum = 0;

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

10. 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;
  return 0;
}
```

Step	Statement	a	p
1	p = 1	?	**1**
2	cin >> a	**3**	1
3	while (a != 0)	True	
4	p = p * a	3	**3**
5	cin >> a	**2**	3
6	while (a != 0)	True	
7	p = p * a	2	**6**
8	cin >> a	**9**	6
9	while (a != 0)	True	
10	p = p * a	9	**54**
11	cin >> a	**0**	54
12	while (a != 0)	False	
13	cout << p	54 is displayed	

11. 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;
```

```
  return 0;
}
```

Chapter 26

26.2 Answers of Review Questions: True/False

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

26.3 Answers of Review Questions: Multiple Choice

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

26.4 Answers of Review Exercises

1. Solution

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

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

2. 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	10 is displayed	
6	while (y < x)	True	
7	y *= 2	39	**20**
8	x++	**40**	20
9	cout << y	20 is displayed	
10	while (y < x)	True	
11	y *= 2	40	**40**
12	x++	**41**	40

13	`cout << y`	40 is displayed	
14	`while (y < x)`	True	
15	`y *= 2`	41	**80**
16	`x++`	**42**	80
17	`cout << y`	80 is displayed	
18	`while (y < x)`	False	

3. Solution

Step	Statement	Notes	x
1	`x = 1`		**1**
2	`if (x % 2 == 0)`	False	
3	`x += 3`		**4**
4	`cout << x`	4 is displayed	
5	`while (x < 12)`	True	
6	`if (x % 2 == 0)`	True	
7	`x++`		**5**
8	`cout << x`	5 is displayed	
9	`while (x < 12)`	True	
10	`if (x % 2 == 0)`	False	
11	`x += 3`		**8**
12	`cout << x`	8 is displayed	
13	`while (x < 12)`	True	
14	`if (x % 2 == 0)`	True	
15	`x++`		**9**
16	`cout << x`	9 is displayed	
17	`while (x < 12)`	True	
18	`if (x % 2 == 0)`	False	
19	`x += 3`		**12**
20	`cout << x`	12 is displayed	
21	`while (x < 12)`	False	

4. 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`	4, 4 is displayed	
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`	20, 16 is displayed	
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`	276, 256 is displayed	
17	`while (y < 65535)`	True	
18	`y = pow(y, 2)`	276	**65536**
19	`if (x < 256)`	False	
20	`cout << x << ", " << y`	276, 65536 is displayed	
21	`while (y < 65535)`	False	

5. 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	`x = a + b`	5	5	8	5	**10**
25	`c = c + 3`	5	5	**11**	5	10
26	`a = b`	**5**	5	11	5	10
27	`b = d`	5	**5**	11	5	10
28	`while (c < 11)`	False				

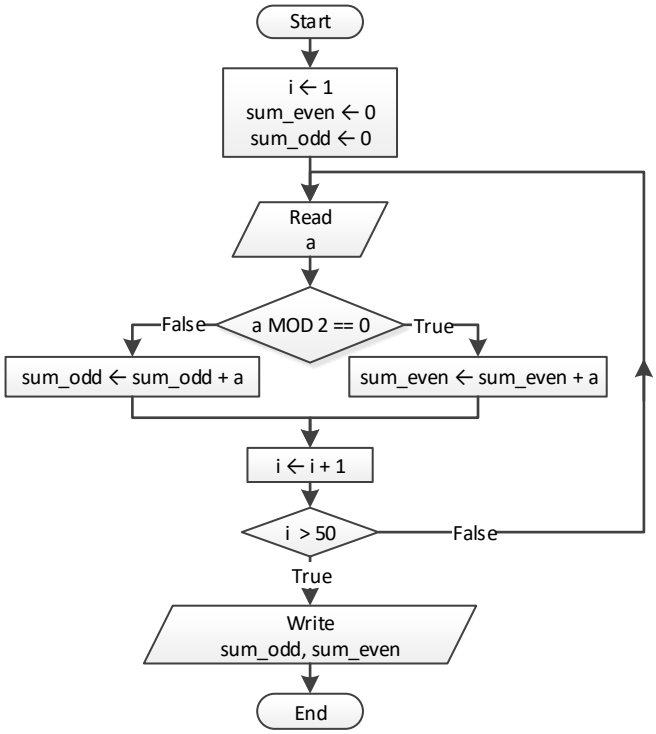
6. Solution

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

7. Solution

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

8. 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;
  return 0;
}
```

9. Solution

```
#include <iostream>
#include <cmath>
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);
  return 0;
}
```

10. 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;
  return 0;
}
```

11. Solution

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

  sum = 0;

  cin >> a;
  if (a > 0) {
    do {
      sum = sum + a;
      cin >> a;
    } while (a > 0);
  }
```

```
  cout << sum;
  return 0;
}
```

Step	Statement	a	sum
1	sum = 0	?	**0**
2	cin >> a	**5**	0
3	if (a > 0)	True	
4	sum = sum + a	5	**5**
5	cin >> a	**2**	5
6	while (a > 0)	True	
7	sum = sum + a	2	**7**
8	cin >> a	**3**	7
9	while (a > 0)	True	
10	sum = sum + a	3	**10**
11	cin >> a	**0**	10
12	while (a > 0)	False	

12. Solution

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

  population = 50000;

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

Chapter 27

27.3 Answers of Review Questions: True/False

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

27.4 Answers of Review Questions: Multiple Choice

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

27.5 Answers of 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	12, 3 is displayed		

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	24 is displayed		

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	1 is displayed		

3. Solution

For input value of 12

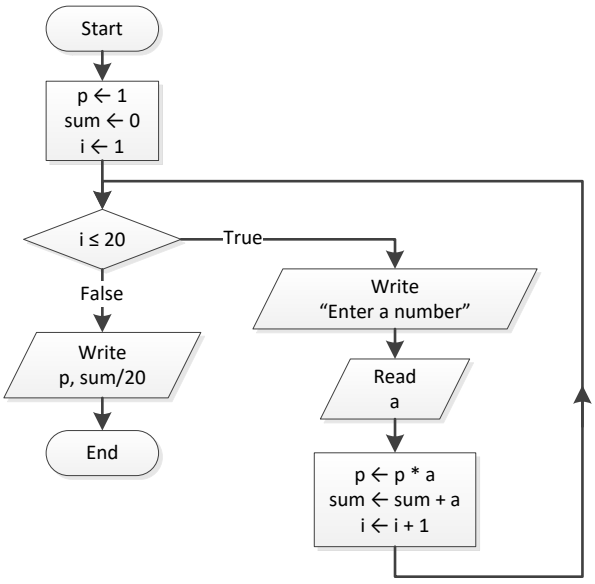
Step	Statement	a	x	y	j
1	cin >> a	**12**	?	?	?
2	j = 2	12	?	?	**2**
3	j <= a - 1	True			
4	x = j * 3 + 3	12	**9**	?	2
5	y = j * 2 + 10	12	9	**14**	2
6	if (y - x > 0 || x > 30)	True			
7	y *= 2	12	9	**28**	2
8	x += 4	12	**13**	28	2
9	cout << x << ", " << y	13, 28 is displayed			
10	j += 3	12	13	28	**5**
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	22, 40 is displayed			
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	`cout << x << ", " << y`	31, 26 is displayed			
25	`j += 3`	12	31	26	**11**
26	`j <= a - 1`	True			
27	`x = j * 3 + 3`	12	**36**	26	11
28	`y = j * 2 + 10`	12	36	**32**	11
29	`if (y - x > 0 || x > 30)`	True			
30	`y *= 2`	12	36	**64**	11
31	`x += 4`	12	**40**	64	11
32	`cout << x << ", " << y`	40, 64 is displayed			
33	`j += 3`	12	40	64	**14**
34	`j <= a - 1`	False			

4. Solution

i. 9
ii. 2
iii. -7
iv. -1

5. Solution



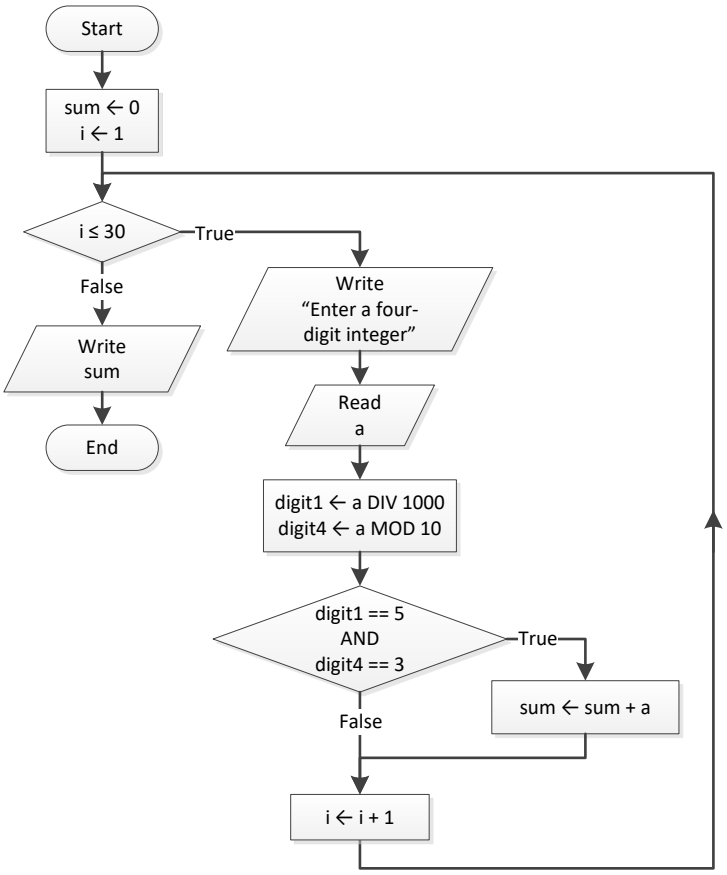

```
#include <iostream>
using namespace std;
int main() {
```

```
  double a, p, sum;
  int i;

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

6. Solution

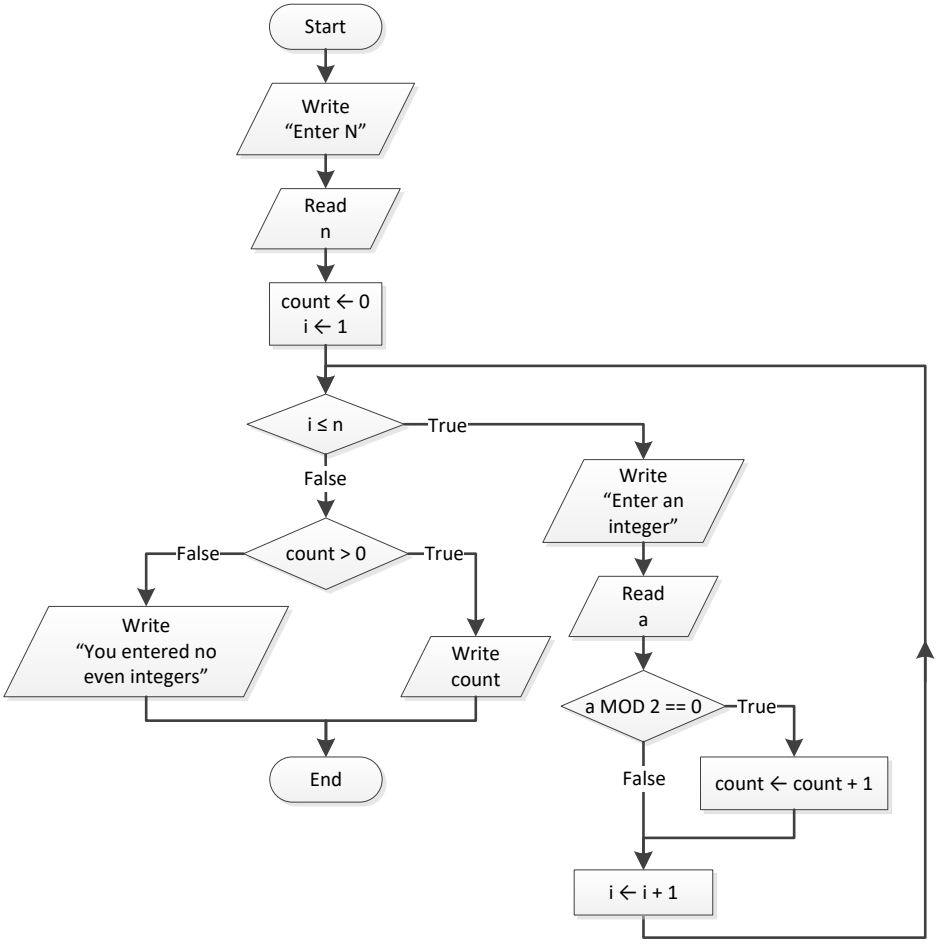
```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
  double deg, i;

  cout << "Enter degrees: ";
  cin >> deg;
  for (i = 0 ; i <= deg; i += 0.5) {
    cout << sin(i * M_PI / 180) << endl;
  }
  return 0;
}
```

7. Solution




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

  sum = 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) {
      sum += a;
    }
  }
  cout << sum;
  return 0;
}
```

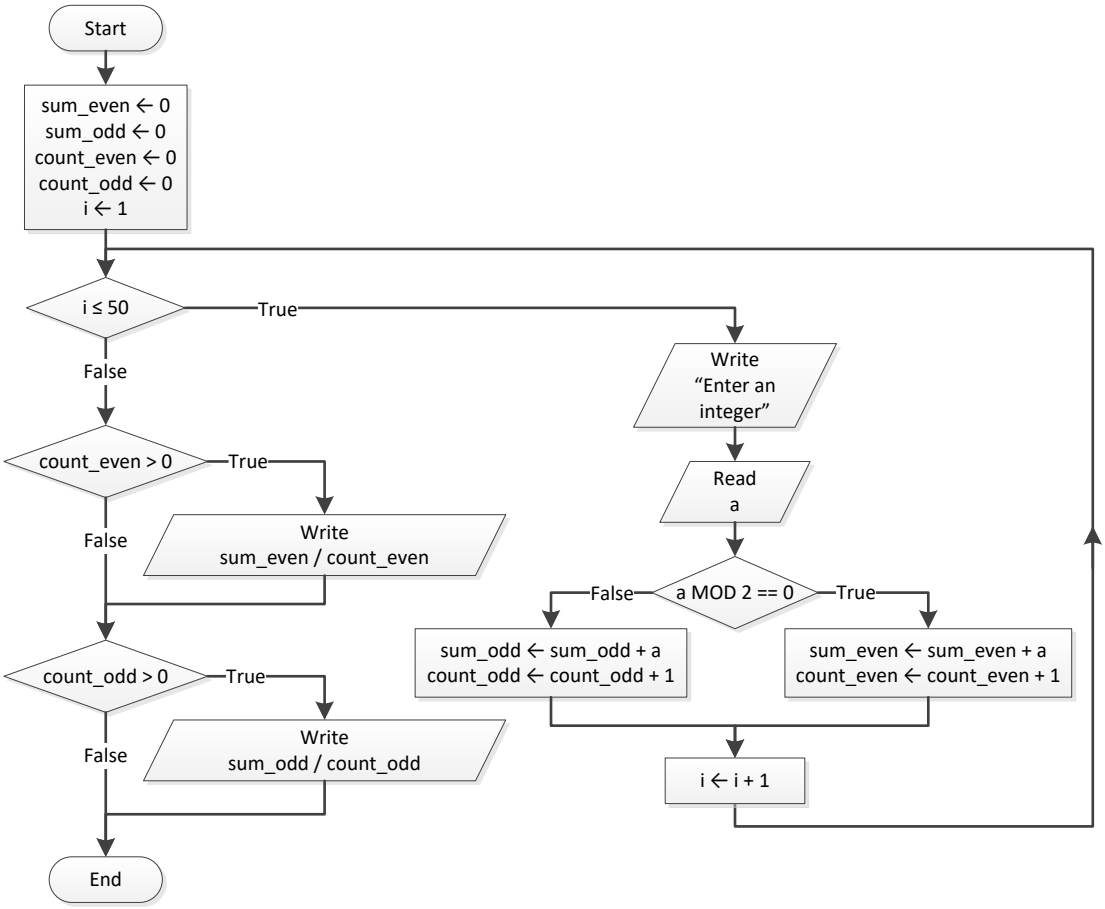
8. Solution




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

  cout << "Enter N: ";
  cin >> n;
  count = 0;
  for (i = 0 ; 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;
}
```

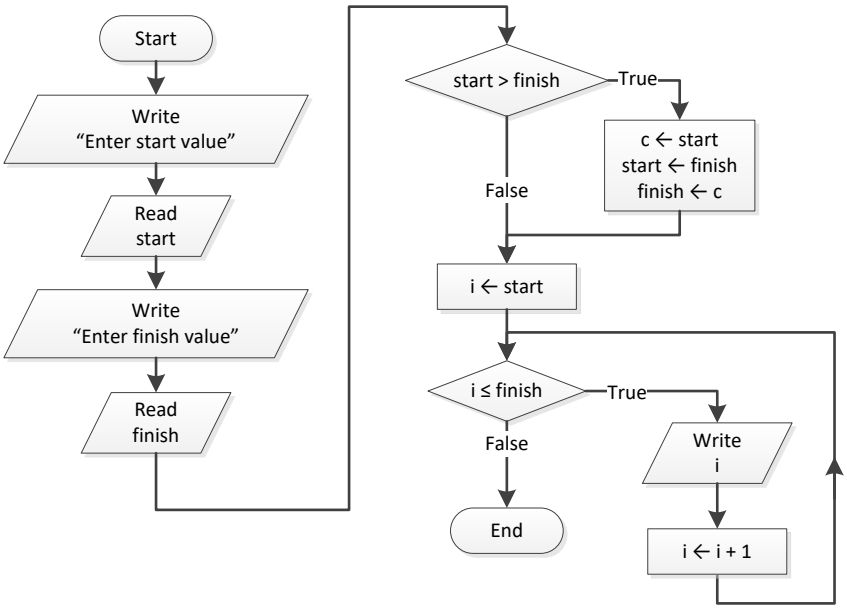
9. Solution




```
#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++;
    }
```

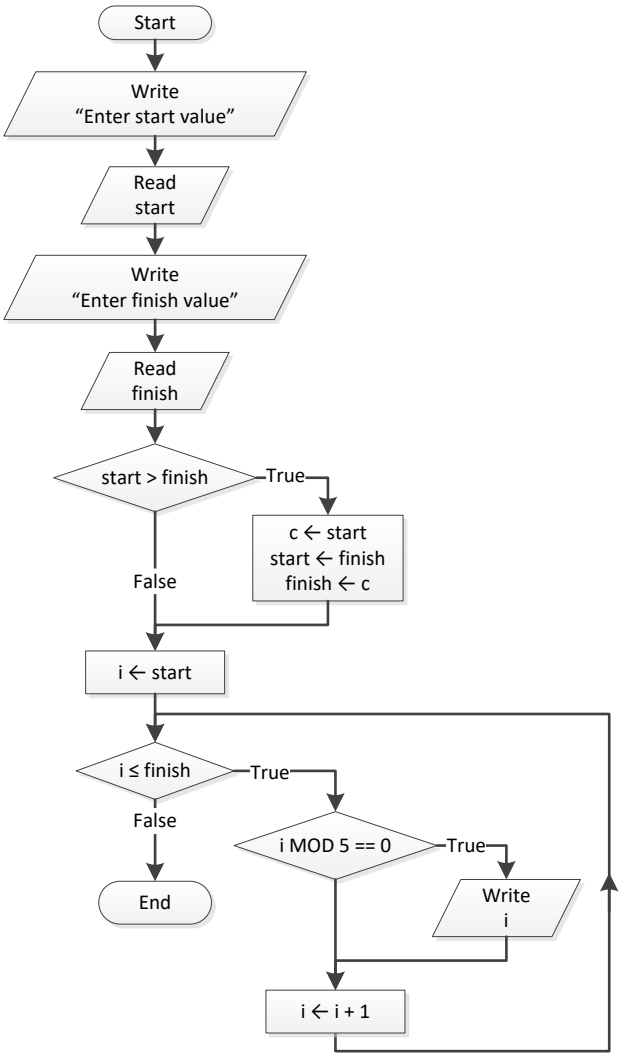
```
  }
  if (count_even > 0) {
    cout << sum_even / (double)count_even << endl;
  }
  if (count_odd > 0) {
    cout << sum_odd / (double)count_odd << endl;
  }
  return 0;
}
```

10. 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;
}
```

11. 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;
}
```

12. Solution

First Approach

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

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

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

Second Approach

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

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

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

```
    p = 1 / p;
  }
  cout << p;
  return 0;
}
```

13. Solution

```
#include <iostream>
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.substr(i, 1) == " ") {
      count++;
    }
  }
  words = count + 1;

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

14. Solution

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

  cout << "Enter a message: ";
  cin >> msg;

  characters = msg.length();
  count = 0;
  for (i = 0; i <= characters - 1; i++) {
    if (msg.substr(i, 1) == " ") {
      count++;
    }
  }
  words = count + 1;
  cout << "The average number of letters in each word is ";
  cout << (characters - count) / (double)words;
  return 0;
}
```

Chapter 28

28.3 Answers of Review Questions: True/False

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

28.4 Answers of Review Questions: Multiple Choice

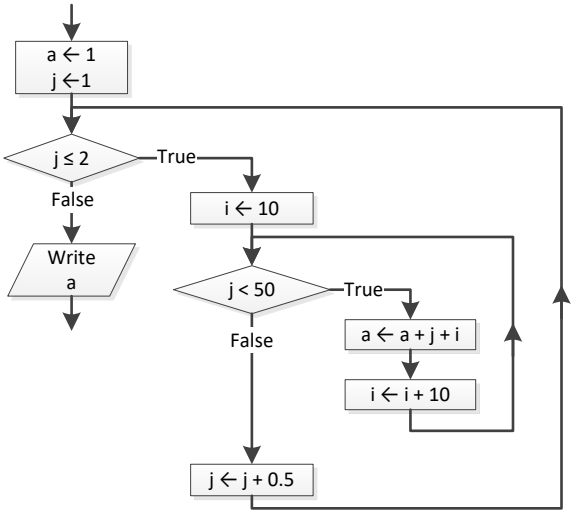
1. b
2. a
3. c
4. a
5. b

28.5 Answers of Review Exercises

1. Solution

i. 10
ii. 4.5
iii. -7
iv. 138

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	100 is displayed		

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	`s = s + i * j`	**3**	1	3
7	`j--`	3	1	**2**
8	`j >= i`		True	
9	`s = s + i * j`	**5**	1	2
10	`j--`	5	1	**1**
11	`j >= i`		True	
12	`s = s + i * j`	**6**	1	1
13	`j--`	6	1	**0**
14	`j >= i`		True	
15	`i++`	6	**2**	0
16	`i <= 4`		True	
17	`j = 3`	6	2	**3**
18	`j >= i`		True	
19	`s = s + i * j`	**12**	2	3
20	`j--`	12	2	**2**
21	`j >= i`		True	
22	`s = s + i * j`	**16**	2	2
23	`j--`	16	2	**1**
24	`j >= i`		False	
25	`i++`	16	**3**	1
26	`i <= 4`		True	
27	`j = 3`	16	3	**3**
28	`j >= i`		True	
29	`s = s + i * j`	**25**	3	3
30	`j--`	25	3	**2**
31	`j >= i`		False	
32	`i++`	25	**4**	2
33	`i <= 4`		True	
34	`j = 3`	25	4	**3**
35	`j >= i`		False	
36	`i++`	25	**5**	3
37	`i <= 4`		False	
38	`cout << s`	25 is displayed		

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`	61 is displayed			

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`	76 is displayed			

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	46 is displayed			

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, n;

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

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

10. Solution

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

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

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

  for (i = 1; i <= n - 2; i++) {
```

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

  for (j = 1; j <= n; 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 29

29.14 Answers of Review Questions: True/False

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

29.15 Answers of Review Questions: Multiple Choice

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

29.16 Answers of Review Exercises

1. Solution

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

2. Solution

```
#include <iostream>
using namespace std;
int main() {
  int denom, i;
  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;
  return 0;
}
```

3. Solution

```
s = 10;
i = 1;
while (i <= 10) {
  s += sqrt(i);
  i++;
}
cout << s << endl;
```

4. Solution

```
cin >> start;
cin >> end;
i = start;
while (i <= end) {
  cout << i << endl;
  i++;
}
```

5. Solution

```
s = 0;
for (i = 100; i >= 5; i -= 5) {
  s = s + sqrt(i);
}
cout << s << endl;
```

6. Solution

```
s = 0;
y = 0;
for (i = 1; i <= 10;  i++) {
  s = s + sqrt(y + i + 1);
  y = y + (i + 1) * 2;
}
cout << s << endl;
```

7. Solution

```
y = 0;
for (i = 1; i <= 9; i += 2) {
  cin >> a;
  a += i;
```

```
  y = y + pow(a + i + 2, 3);
}
cout << y << endl;
```

8. Solution

This conversion should not be carried out.

9. Solution

```
s = 0;
cin >> a;
s += a;
cin >> a;
while (a <= s) {
  s += a;
  cin >> a;
}
cout << s << endl;
```

10. Solution

```
a = 100;
count = 0;
cout << a << endl;
cin >> b;
count++;
a -= sqrt(b);
while (a >= 0) {
  cout << a << endl;
  cin >> b;
  count++;
  a -= sqrt(b);
}
cout << count << endl;
```

11. Solution

```
cin >> a;
cin >> b;
if (b <= 1000) {
  do {
    a += 2;
    b = b * a;
    cout << b << endl;
  } while (b <= 1000);
}
```

12. Solution

```
s = 0;
cin >> a;
if (a != -99) {
```

```
  do {
    s = s + pow(a, 2);
    cin >> a;
  } while (a != -99);
}
cout << s << endl;
```

13. Solution

```
x = 0;
y = -10;
do {
  x = x + pow(2, y);
  y = y + 1;
} while (y < 10);
cout << x << endl;
```

14. Solution

```
cin >> start;
x = 1;
i = start;
while (i <= start * 2) {
  x = pow(x, 1.1) + i;
  i++;
}
cout << x << endl;
```

15. Solution

```
x = 42;
i = 1;
while (i <= 100) {
  x = sqrt(x) + i;
  cout << x << endl;
  i++;
}
```

16. Solution

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

  p = 1;
  cin >> a;
  i = a;
  p = p * pow(i, 2);
  i = i + 5;
  p = p + i;
```

```
  for (i = a + 5; i <= 19; i += 5) {
    p = p * pow(i, 2);
    p = p + i + 5;
  }
  cout << p;
  return 0;
}
```

17. Solution

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

  cin >> start;
  cin >> end;
  x = 1000;

  for (i = start; i <= end; i+=2) {
    x = sqrt(x);
  }
  cout << x;
  return 0;
}
```

18. 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;
}
```

19. Solution

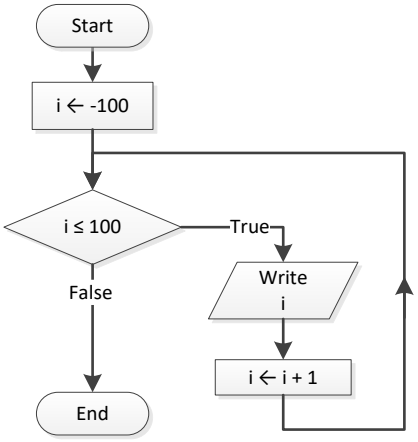
```
#include <iostream>
using namespace std;
int main() {
  int i, j;

  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;
}
```

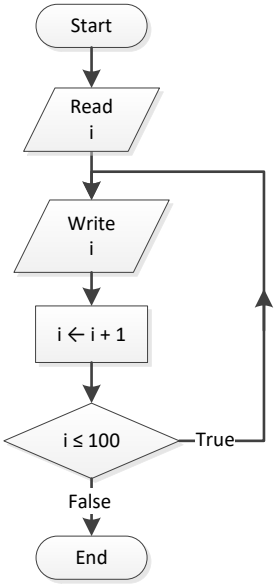
Chapter 30

30.4 Answers of Review Exercises

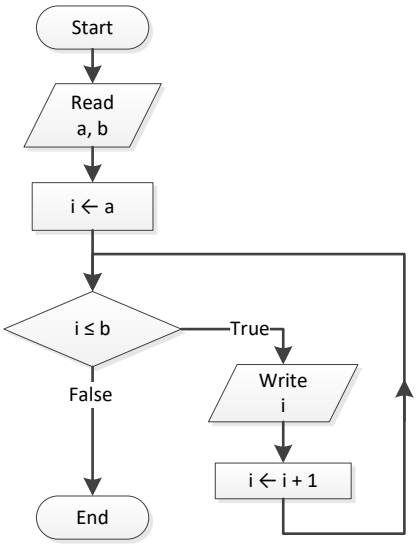
1. Solution



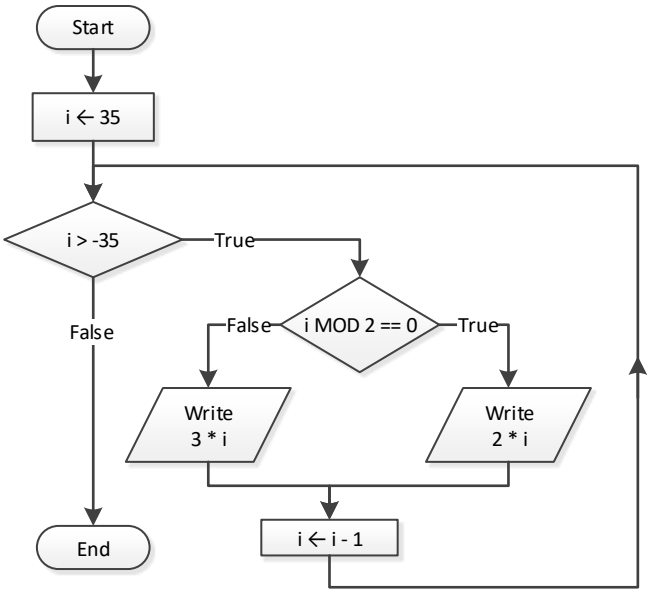
2. Solution



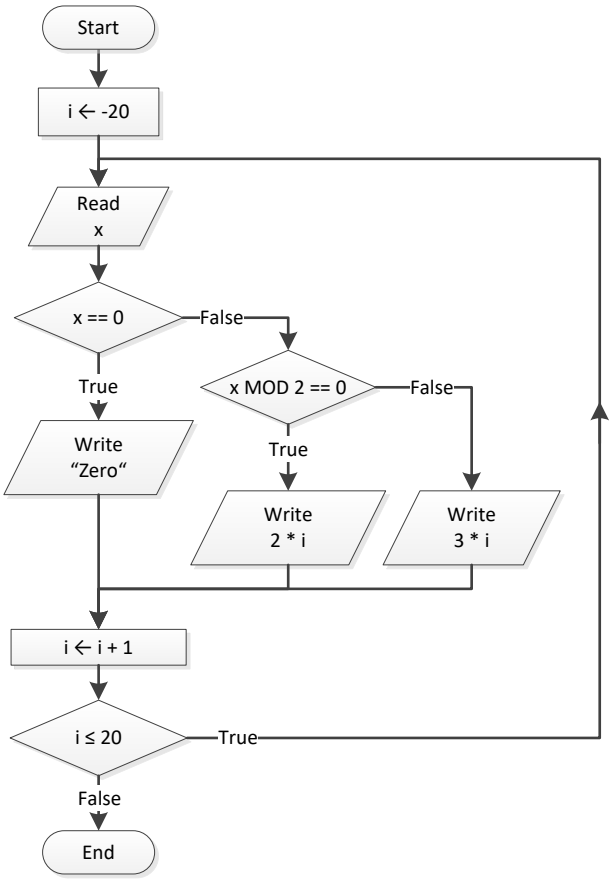
3. Solution



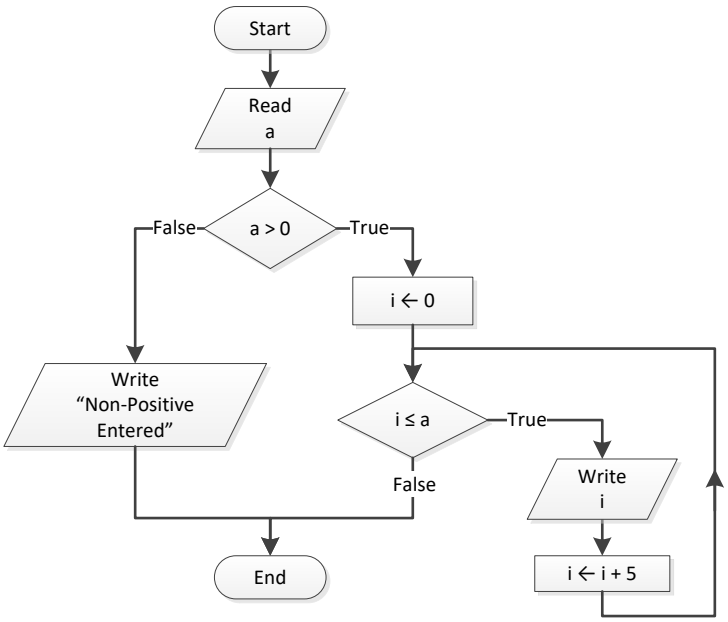
4. Solution



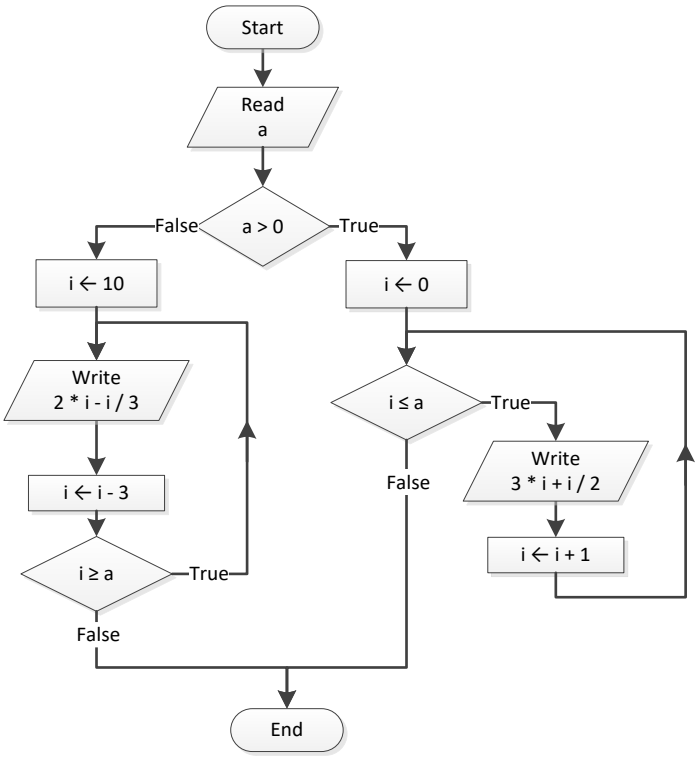
5. Solution



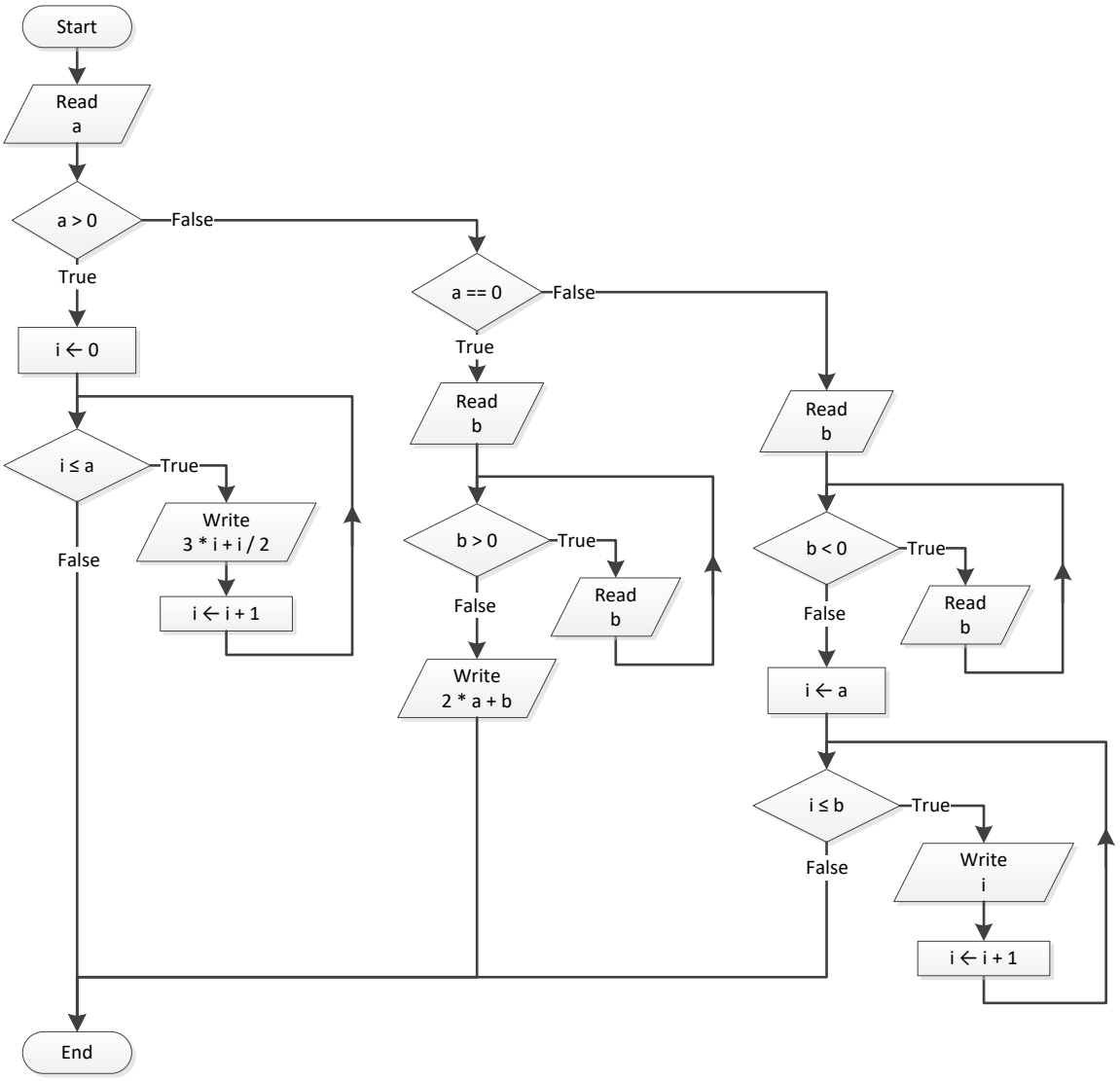
6. Solution



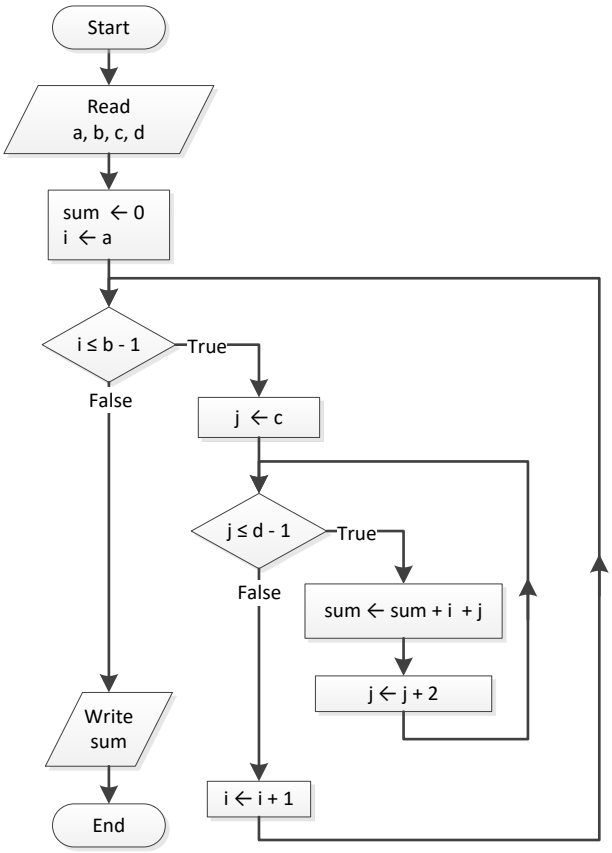
7. Solution


Start
Read
a
a > 0
False
True
i ← 10
i ← 0
Write
2 * i - i / 3
i ≤ a
True
Write
3 * i + i / 2
False
i ← i - 3
i ≥ a
True
i ← i + 1
False
End

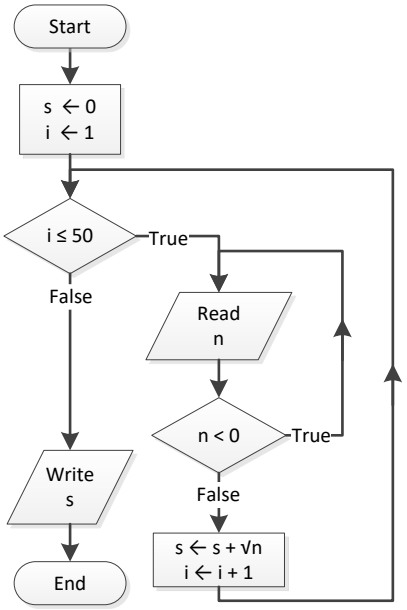
8. Solution


Start
Read a
a > 0
False
True
i ← 0
i ≤ a
True
Write 3 * i + i / 2
i ← i + 1
False
a == 0
False
True
Read b
b > 0
True
Read b
False
Write 2 * a + b
Read b
b < 0
True
Read b
False
i ← a
i ≤ b
True
Write i
i ← i + 1
False
End

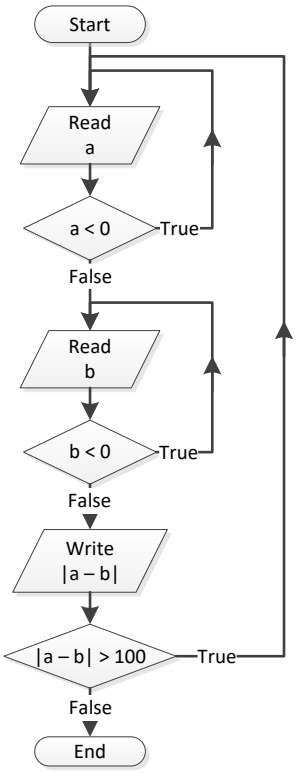
9. Solution



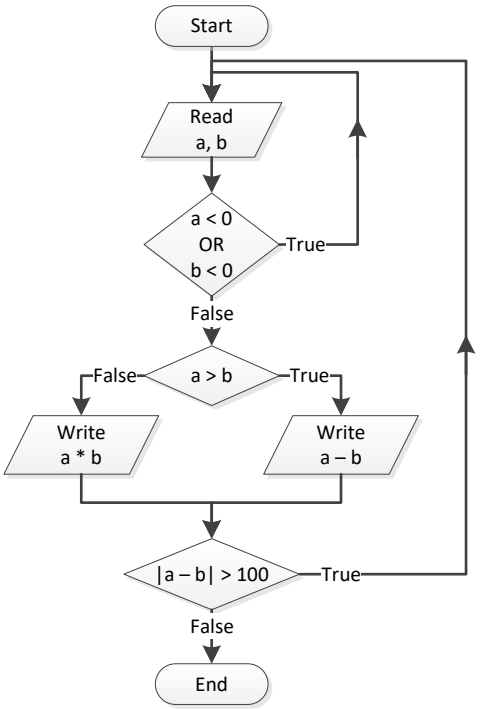
10. Solution



11. Solution



12. Solution



13. Solution

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

14. 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;
}
```

15. 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;
}
```

16. Solution

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

```
  double P, S, a;

  i = 1;
  S = 0;
  P = 1;
  a = 0;
  if (i < 45) {
    S += a;
  }
  else {
    P *= a;
  }
  i++;
  while (i < 90) {
    cin >> a;
    if (i < 45) {
      S += a;
    }
    else {
      P *= a;
    }
    i++;
  }
  cout << S << ", " << P;
  return 0;
}
```

Chapter 31

31.7 Answers of Review Questions: True/False

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

31.8 Answers of Review Exercises

1. Solution

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

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

2. Solution

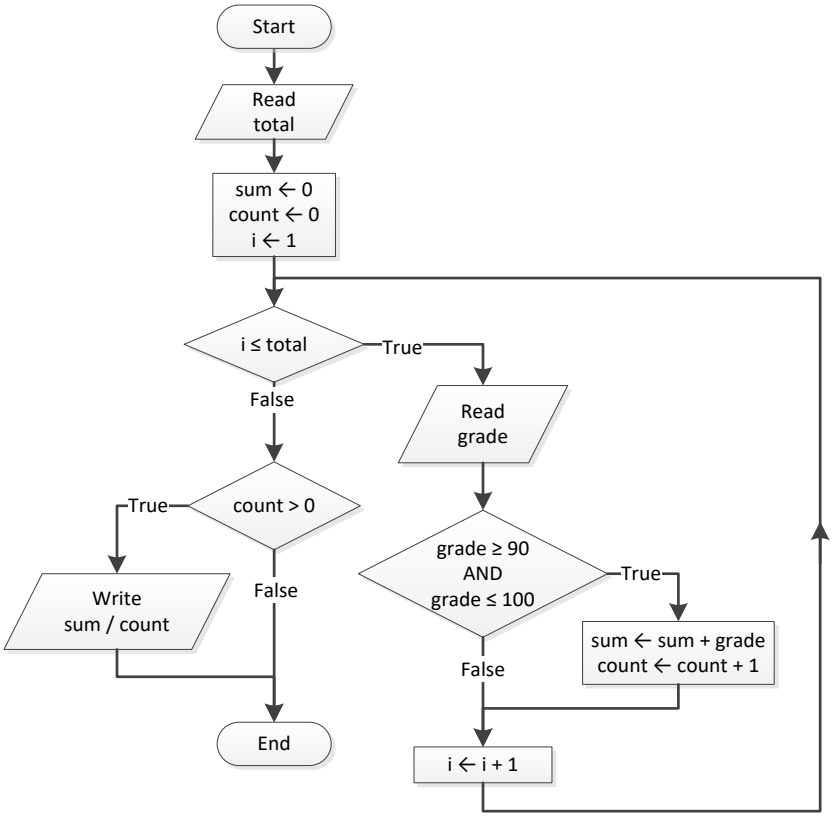
```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
  int i, n;
  double p;

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

3. 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;
  return 0;
}
```

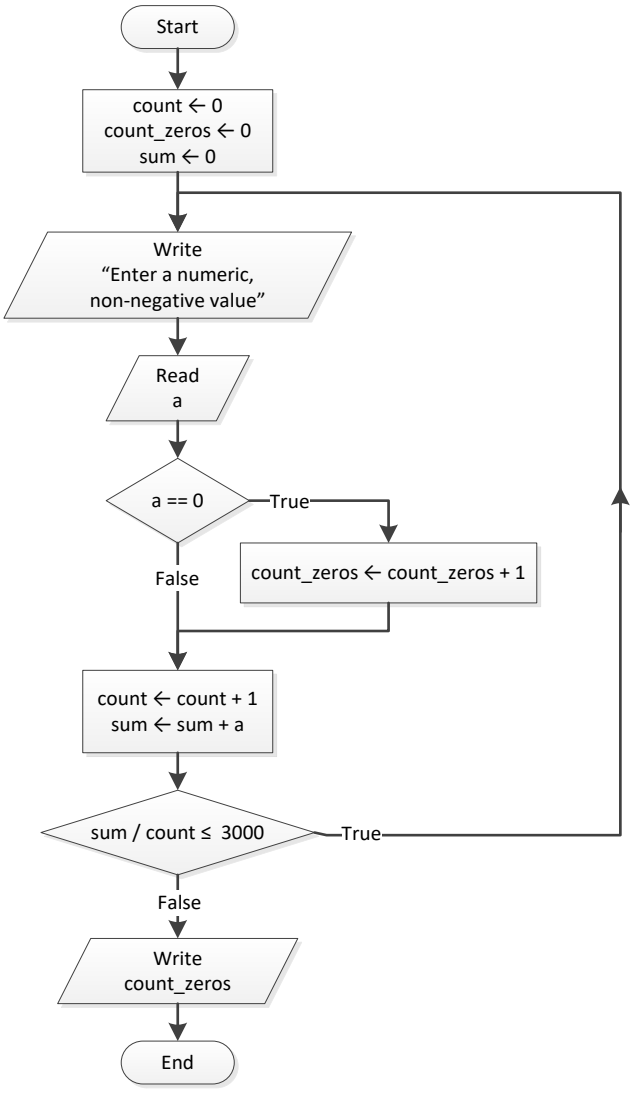
4. Solution




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

  cin >> total;
  sum = 0;
  count = 0;
  for (i = 1; i <= total; i++) {
    cin >> grade;
    if (grade >= 90 && grade <= 100) {
      sum += grade;
      count++;
    }
```

```
  }
  if (count > 0) {
    cout << sum / (double)count << endl;
  }
  return 0;
}
```

5. Solution




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

  count = 0;
  count_zeros = 0;
  sum = 0;
  do {
```

```
    cout << "Enter a numeric, non-negative value: ";
    cin >> a;
    if (a == 0) {
      count_zeros++;
    }
    count++;
    sum += a;
  } while (sum / count <= 3000);
  cout << count_zeros;
  return 0;
}
```

6. 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;
}
```

7. 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;
}
```

8. Solution

```
cin >> input;
while (cin.fail() == true || input != 1 && input != 0) {
    cout << "Error" << endl;
    cin >> input;
}
```

9. Solution

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

10. Solution

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

int main() {
  int count, x;
  double y;

  cout << "Enter a non-negative number: ";
  cin >> x;
  count = 1;
  while (count < 3 && cin.fail() == true || x < 0) {
    cout << "Error: Invalid number!" << endl;
    cout << "Enter a non-negative number: ";
    cin.clear();
    cin.ignore(100, '\n');
    cin >> x;
    if (cin.fail() == true || x < 0) {
      count++;
    }
  }

  if (count < 3) {
    y = sqrt(x);
    cout << y << endl;
  }
  else {
    cout << "Dude, you are dumb!" << endl;
  }
  return 0;
}
```

11. Solution

```
#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;
}
```

12. 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;
}
```

13. 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;
}
```

14. Solution

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

  cin >> m1;
  cin >> m2;
  cin >> 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;
  return 0;
}
```

15. 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 (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;
}
```

16. Solution

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

int main() {
  int a, b, c, d1, d2, d3, d4, r, x;

  cout << "Enter a four-digit integer: ";
  cin >> a;
  while (cin.fail() == true || a < 1000 || a > 9999) {
    cout << "Wrong number. Please enter a four-digit integer: ";
    cin.clear();
    cin.ignore(100, '\n');
    cin >> a;
  }

  cout << "Enter a second four-digit integer: ";
  cin >> b;
  while (cin.fail() == true || b < 1000 || b > 9999) {
    cout << "Wrong number. Please enter a second four-digit integer: ";
    cin.clear();
    cin.ignore(100, '\n');
    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;
}
```

17. 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;
}
```

18. 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;
}
```

19. Solution

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

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

20. 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;
}
```

21. 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;
}
```

22. 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;
}
```

23. Solution

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

int main() {
  int denominator, i, n, nominator;
  double y;

  cout << "Enter a positive integer: ";
  cin >> n;
  while (cin.fail() == true || n < 1) {
    cout << "Wrong number. Please enter a positive integer: ";
    cin.clear();
    cin.ignore(100, '\n');
    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;
  return 0;
}
```

24. Solution

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

int main() {
  int i, n, nominator, sign;
  double y;

  cout << "Enter a positive integer: ";
  cin >> n;
  while (cin.fail() == true || n < 1) {
    cout << "Wrong number. Please enter a positive integer: ";
    cin.clear();
    cin.ignore(100, '\n');
    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;
  return 0;
}
```

25. Solution

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

int main() {
  int i, n, sign;
  double y;
```

```
  cout << "Enter a positive integer: ";
  cin >> n;
  while (cin.fail() == true || n < 1) {
    cout << "Wrong number. Please enter a positive integer: ";
    cin.clear();
    cin.ignore(100, '\n');
    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;
  return 0;
}
```

26. Solution

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

int main() {
  int i, n;
  double y;

  cout << "Enter a positive integer: ";
  cin >> n;
  while (cin.fail() == true || n < 1) {
    cout << "Wrong number. Please enter a positive integer: ";
    cin.clear();
    cin.ignore(100, '\n');
    cin >> n;
  }

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

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

27. 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;
  return 0;
}
```

***Notice**: Please note that this C++ code operates properly for all non-negative integers, including zero.*

28. 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;
  return 0;
}
```

Second 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 = 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;
  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 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;
  return 0;
}
```

Second 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 = 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;
  return 0;
}
```

30. 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, j, 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;
  return 0;
```

```
}
```

31. Solution

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

int main() {
  int i;
  double max, sum, t;
  bool failure;

  max = -460;
  sum = 0;
  for (i = 1; i <= 31; i++) {
    do {
      cout << "Enter temperature for day " << i << ": ";
      cin >> t;

      failure = false;
      if (cin.fail() == true) {
        cout << "Please enter numeric values!" << endl;
        cin.clear();
        cin.ignore(100, '\n');
        failure = true;
      }
      else if (t < -459.67) {
        cout << "Please enter a value greater than 459.67" << endl;
        failure = true;
      }
    } while (failure == true);

    sum += t;
    if (t > max) {
      max = t;
    }
  }

  cout << sum / 31 << ", " << max;
  return 0;
}
```

32. Solution

```
#include <iostream>
using namespace std;
int main() {
  int hour, max_hour, max_minutes, min_hour, min_minutes, minutes;
  double level, max, min;

  cin >> level;
  if (level != 9999) {
    cin >> hour;
```

```
    cin >> minutes;

    max = level;
    max_hour = hour;
    max_minutes = minutes;

    min = level;
    min_hour = hour;
    min_minutes = minutes;

    cin >> level;
    while (level != 9999) {
      cin >> hour;
      cin >> minutes;

      if (level > max) {
        max = level;
        max_hour = hour;
        max_minutes = minutes;
      }

      if (level < min) {
        min = level;
        min_hour = hour;
        min_minutes = minutes;
      }

      cin >> level;
    }

    cout << max << ", " << max_hour << ", " << max_minutes << endl;
    cout << min << ", " << min_hour << ", " << min_minutes << endl;
  }
  return 0;
}
```

33. Solution

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

int main() {
  int a, b, c, i;
  string alphabet;
  bool failure;

  alphabet = "abcdefghijklmnopqrstuvwxyz";

  do {
    cout << "Enter an integer between 1 and 26: ";
    cin >> a;

    failure = false;
    if (cin.fail() == true) {
```

```
      cout << "Please enter numeric values!" << endl;
      cin.clear();
      cin.ignore(100, '\n');
      failure = true;
    }
    else 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 == true);

  do {
    cout << "Enter an integer between 1 and 26: ";
    cin >> b;

    failure = false;
    if (cin.fail() == true) {
      cout << "Please enter numeric values!" << endl;
      cin.clear();
      cin.ignore(100, '\n');
      failure = true;
    }
    else 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 == true);

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

  for (i = a; i <= b; i++) {
    cout << alphabet.at(i - 1);
  }
  return 0;
}
```

34. 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;
  return 0;
}
```

35. 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++;
      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;
}
```

36. Solution

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

  do {
    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 == 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;
    }
  } while (choice != 3);
  return 0;
}
```

37. 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, max, min, n, sum, sum_a, sum_a_boys, sum_b;
  string sex;

  cout << "Enter total number of students: ";
  cin >> n;
  while (cin.fail() == true || n < 1) {
    cout << "Wrong number. Please enter total number of students: ";
    cin.clear();
    cin.ignore(100, '\n');
    cin >> n;
  }

  sum = 0;
  sum_a = 0;
  count_a = 0;
  sum_b = 0;
  count_b = 0;
  sum_a_boys = 0;
  count_a_boys = 0;
  count_cdef_girls = 0;

  max = -1;
  min = 101;

  for (i = 1; i <= n; i++) {
    cout << "Enter grade for student No " << i << ": ";
    cin >> grade;
    while (cin.fail() == true || grade < 0 || grade > 100) {
      cout << "Wrong grade. Please enter grade for student No " << i << ": ";
      cin.clear();
      cin.ignore(100, '\n');
      cin >> grade;
    }

    cout << "Enter sex for student No " << i << ": ";
    cin >> sex;
    sex = to_upper_copy(sex);
    while (sex != "M"  && sex != "F") {
      cout << "Wrong sex. Please enter sex for student No " << i << ": ";
      cin.clear();
      cin.ignore(100, '\n');
      cin >> sex;
      sex = to_upper_copy(sex);
    }

    if (grade >= 90 && grade <= 100) {
      sum_a += grade;
      count_a++;
```

```
      if (sex == "M") {
        sum_a_boys += grade;
        count_a_boys++;
      }
    }
    else if (grade >= 80 && grade <= 89) {
      sum_b += grade;
      count_b++;
    }
    else {
      if (sex == "F") {
        count_cdef_girls++;
      }
    }

    if (grade > max) {
      max = grade;
    }

    if (grade < min) {
      min = grade;
    }

    sum += grade;
  }

  if (count_a > 0) {
    cout << "The average value of those who got an 'A' is: ";
    cout << sum_a / (double)count_a << endl;
  }
  if (count_b > 0) {
    cout << "The average value of those who got a 'B' is: ";
    cout << sum_b / (double)count_b << endl;
  }
  if (count_a_boys > 0) {
    cout << "The average value of boys who got an 'A' is: ";
    cout << sum_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: " << max << endl;
  cout << "The lowest grade is: " << min << endl;
  cout << "The average grade of the whole class is: " << sum / (double)n;
  return 0;
}
```

38. 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 (cin.fail() == true || amount < 0) {
      cout << "Wrong amount. Please enter amount: " << endl;
      cin.clear();
      cin.ignore(100, '\n');
      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;
  } while (to_upper_copy(answer) == "YES");
  return 0;
}
```

39. 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 (cin.fail() == true || kwh < 0 && kwh != -1) {
    cout << "Wrong value. Please enter number of Kilowatt-hours consumed: ";
    cin.clear();
    cin.ignore(100, '\n');
    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 (cin.fail() == true || kwh < 0 && kwh != -1) {
      cout << "Wrong value. Please enter number of Kilowatt-hours consumed: ";
      cin.clear();
      cin.ignore(100, '\n');
      cin >> kwh;
    }
  }
  return 0;
}
```

Chapter 32

32.3 Answers of Review Questions: True/False

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

32.4 Answers of Review Exercises

1. Solution

Weights =

170	*0*
190	*1*
193	*2*
165	*3*
200	*4*

People

2. Solution

Names =

John Thompson
Ava Brown
Ryan Miller
Antony Harris
Alexander Lewis
Samantha Clark
Chloe Parker

Weights =

170	*0*
190	*1*
193	*2*
165	*3*
200	*4*
170	*5*
172	*6*

People

3. Solution

Months

0 *1* *2*

Names =

Toba
Issyk Kul
Baikal
Crater
Karakul

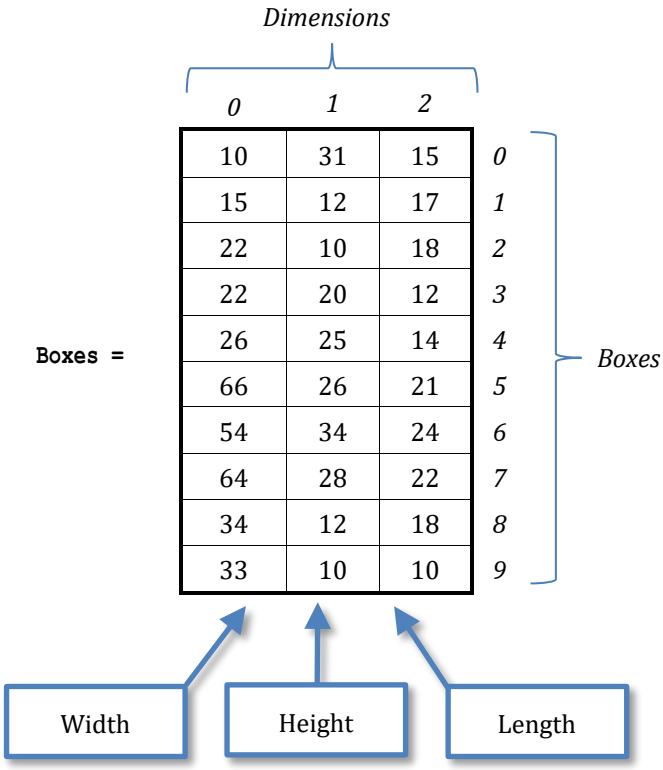
Areas =

440	438	437	*0*
2408	2405	2402	*1*
12248	12247	12240	*2*
21	20	18	*3*
150	145	142	*4*

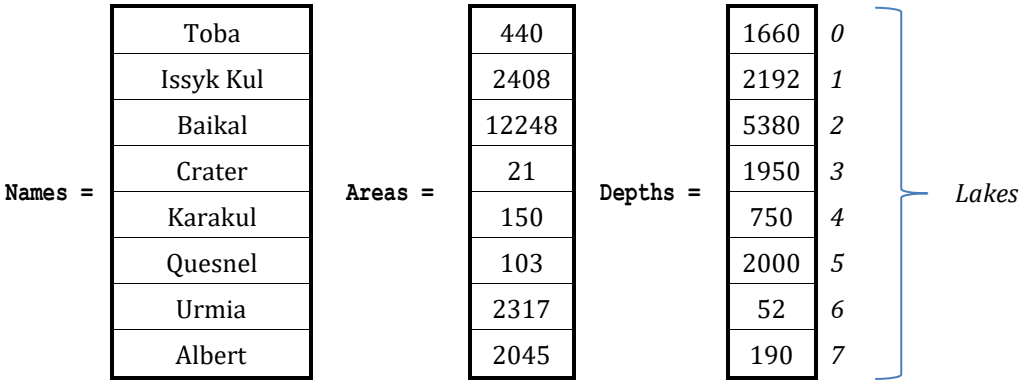
Lakes

June July August

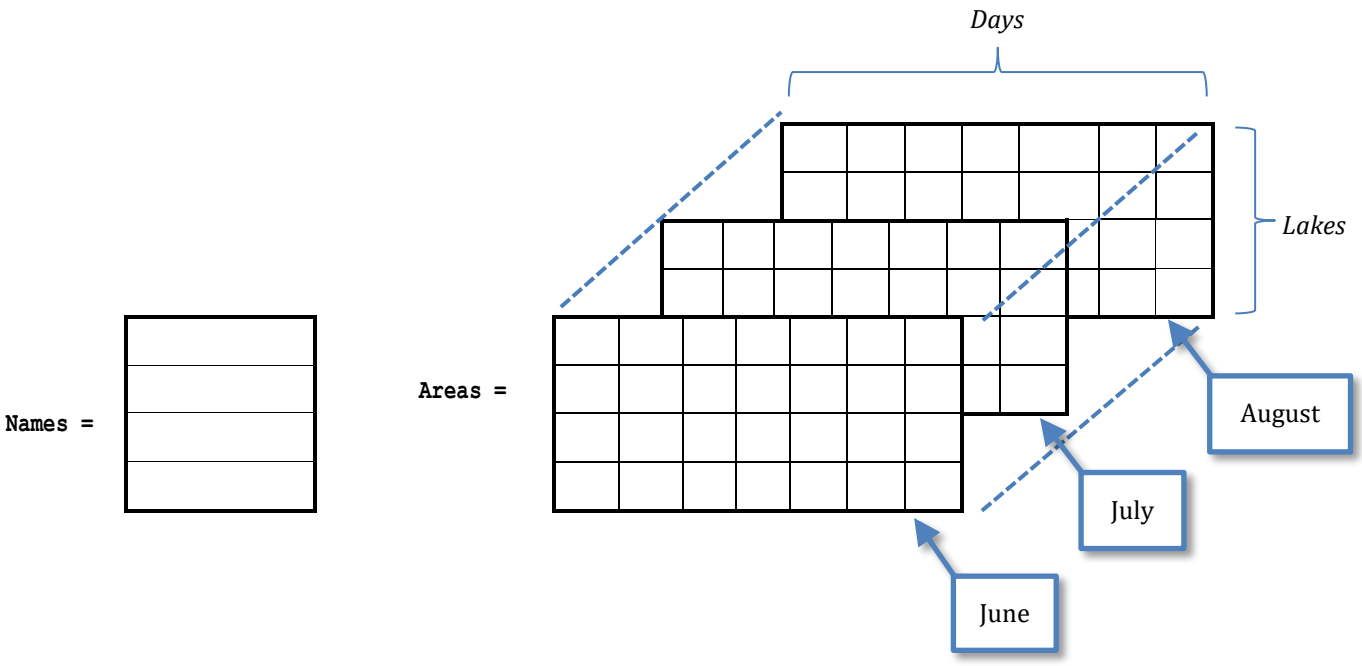
4. Solution



5. Solution



6. Solution



Chapter 33

33.5 Answers of Review Questions: True/False

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

33.6 Answers of Review Questions: Multiple Choice

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

33.7 Answers of Review Exercises

1. 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

2. 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

3. 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

4. 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

5. 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)`	4	3	**1**	?	?

	a[0] = 1;
	else
	a[0] = y;					

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**

6. Solution

Step	Statement	i	a[0]	a[1]	a[2]	a[3]	a[4]	a[5]
1	int a[6] = {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			

7. Solution

Step	Statement	i	a[0]	a[1]	a[2]	a[3]	a[4]	a[5]
1	`int a[6] = {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			

8. 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;
}
```

9. 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 = ELEMENTS - 1; i >= 0; i--) {
    cout << pow(a[i], 2) << endl;
  }
  return 0;
}
```

10. 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;
}
```

11. 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;
}
```

12. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 30;

int main() {
  int i;
  double sum;

  double a[ELEMENTS];
  for (i = 0; i <= ELEMENTS - 1; i++) {
    cin >> a[i];
  }

  sum = 0;
  for (i = 0; i <= ELEMENTS - 1; i++) {
    if (a[i] > 0) {
      sum += a[i];
    }
  }
  cout << sum;
  return 0;
}
```

13. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 50;

int main() {
  int i, sum;

  int a[ELEMENTS];
  for (i = 0; i <= ELEMENTS - 1; i++) {
    cin >> a[i];
  }

  sum = 0;
  for (i = 0; i <= ELEMENTS - 1; i++) {
    if (a[i] >= 10 && a[i] <= 99) {
      sum += a[i];
    }
  }
  cout << sum;
  return 0;
}
```

14. 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 = 0;
  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;
  return 0;
}
```

15. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 20;

int main() {
  int i;
  double sum;

  double a[ELEMENTS];
  for (i = 0; i <= ELEMENTS - 1; i++) {
    cin >> a[i];
  }

  sum = 0;
  for (i = 0; i <= ELEMENTS - 1; i++) {
    sum += a[i];
  }
  cout << sum / ELEMENTS;
  return 0;
}
```

16. 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;
}
```

17. 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++) {
    cin >> a[i];
  }

  for (i = 0; i <= WORDS - 1; i++) {
    count = 0;
    for (j = 0; j <= a[i].length() - 1; j++) {
      if (a[i].substr(j, 1) == "w") {
        count++;
      }
    }
    if (count >= 2) {
      cout << a[i] << endl;
    }
  }
  return 0;
}
```

Chapter 34

34.7 Answers of Review Questions: True/False

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

34.8 Answers of Review Questions: Multiple Choice

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

34.9 Answers of Review Exercises

1. Solution

Step	Statement	x	a
1	`int a[2][3]`	?	? ? ? ? ? ?
2	`a[0][2] = 1`	?	? ? **1** ? ? ?
3	`x = 0`	**0**	? ? 1 ? ? ?
4	`a[0][x] = 9`	0	**9** ? 1 ? ? ?

5	a[0][x + a[0][2]] = 4	0	9 **4** 1 ? ? ?
6	a[a[0][2]][2] = 19	0	9 4 1 ? ? **19**
7	a[a[0][2]][x + 1] = 13	0	9 4 1 ? **13** 19
8	a[a[0][2]][x] = 15	0	9 4 1 **15** 13 19

2. Solution

Step	Statement	i	j	a
1	int a[2][3]	?	?	? ? ? ? ? ?
2	i = 0	**0**	?	? ? ? ? ? ?
3	i <= 1			True
4	j = 0	0	**0**	? ? ? ? ? ?
5	j <= 2			True
6	a[i][j] = (i + 1) * 5 + j	0	0	**5** ? ? ? ? ?
7	j++	0	**1**	5 ? ? ? ? ?
8	j <= 2			True
9	a[i][j] = (i + 1) * 5 + j	0	1	5 **6** ? ? ? ?
10	j++	0	**2**	5 6 ? ? ? ?
11	j <= 2			True

12	`a[i][j] = (i + 1) * 5 + j`	0	2	5 6 **7** ? ? ?
13	`j++`	0	**3**	5 6 7 ? ? ?
14	`j <= 2`			False
15	`i++`	**1**	3	5 6 7 ? ? ?
16	`i <= 1`			True
17	`j = 0`	1	**0**	5 6 7 ? ? ?
18	`j <= 2`			True
19	`a[i][j] = (i + 1) * 5 + j`	1	0	5 6 7 **10** ? ?
20	`j++`	1	**1**	5 6 7 10 ? ?
21	`j <= 2`			True
22	`a[i][j] = (i + 1) * 5 + j`	1	1	5 6 7 10 **11** ?
23	`j++`	1	**2**	5 6 ? 10 11 ?
24	`j <= 2`			True
25	`a[i][j] = (i + 1) * 5 + j`	1	2	5 6 ? 10 11 **12**
26	`j++`	1	**3**	5 6 7 10 11 12
27	`j <= 2`			False

3. Solution

Step	Statement	i	j	a
1	`int a[3][3]`	?	?	? ? ? ? ? ? ? ? ?
2	`j = 0`	?	**0**	? ? ? ? ? ? ? ? ?
3	`j <= 2`			True
4	`i = 0`	0	**0**	? ? ? ? ? ? ? ? ?
5	`i <= 2`			True
6	`a[i][j] = (i + 1) * 2 + j * 4`	0	0	**2** ? ? ? ? ? ? ? ?
7	`i++`	**1**	0	2 ? ? ? ? ? ? ? ?
8	`i <= 2`			True
9	`a[i][j] = (i + 1) * 2 + j * 4`	1	0	2 ? ? **4** ? ? 6 ? ?
10	`i++`	**2**	0	2 ? ? 4 ? ? 6 ? ?
11	`i <= 2`			True
12	`a[i][j] = (i + 1) * 2 + j * 4`	2	0	2 ? ? 4 ? ? **6** ? ?
13	`i++`	**3**	0	2 ? ? 4 ? ? 6 ? ?
14	`i <= 2`			False

15	`j++`	3	**1**	2 ? ? 4 ? ? 6 ? ?
16	`j <= 2`			True
17	`i = 0`	**0**	1	2 ? ? 4 ? ? 6 ? ?
18	`i <= 2`			True
19	`a[i][j] = (i + 1) * 2 + j * 4`	0	1	2 **6** ? 4 ? ? 6 ? ?
20	`i++`	**1**	1	2 6 ? 4 ? ? 6 ? ?
21	`i <= 2`			True
22	`a[i][j] = (i + 1) * 2 + j * 4`	1	1	2 6 ? 4 **8** ? 6 ? ?
23	`i++`	**2**	1	2 6 ? 4 8 ? 6 ? ?
24	`i <= 2`			True
25	`a[i][j] = (i + 1) * 2 + j * 4`	2	1	2 6 ? 4 8 ? 6 **10** ?
26	`i++`	**3**	1	2 6 ? 4 8 ? 6 10 ?
27	`i <= 2`			False
28	`j++`	3	**2**	2 6 ? 4 8 ? 6 10 ?

29	`j <= 2`			True
30	`i = 0`	**0**	2	2 6 ? 4 8 ? 6 10 ?
31	`i <= 2`			True
32	`a[i][j] = (i + 1) * 2 + j * 4`	0	2	2 6 **10** 4 8 ? 6 10 ?
33	`i++`	**1**	2	2 6 10 4 8 ? 6 10 ?
34	`i <= 2`			True
35	`a[i][j] = (i + 1) * 2 + j * 4`	1	2	2 6 10 4 8 **12** 6 10 ?
36	`i++`	**2**	2	2 6 10 4 8 12 6 10 ?
37	`i <= 2`			True
38	`a[i][j] = (i + 1) * 2 + j * 4`	2	2	2 6 10 4 8 12 6 10 **14**
39	`i++`	**3**	2	2 6 10 4 8 12 6 10 14
40	`i <= 2`			False
41	`j++`	3	**3**	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 sum;

  double a[ROWS][COLUMNS];
  for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS - 1; j++) {
      cin >> a[i][j];
    }
  }

  sum = 0;
  for (i = 1; i <= ROWS - 1; i += 2) {
    for (j = 0; j <= COLUMNS - 1; j += 2) {
      sum += a[i][j];
    }
  }
  cout << sum;
  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;
  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;
  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[3] = {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;
        }
      }
    }
  }
  return 0;
}
```

Chapter 35

35.7 Answers of Review Questions: True/False

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

35.8 Answers of Review Questions: Multiple Choice

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

35.9 Answers of 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;
    }
  }
  return 0;
}
```

2. Solution

```
#include <iostream>
using namespace std;
const int OBJECTS = 5;
const int FALLS = 10;

int main() {
  int i, j, sum;

  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++) {
    sum = 0;
    for (j = 0; j <= FALLS - 1; j++) {
      sum += g[i][j];
    }
    cout << "Average g for object No " << (i + 1) << ": " << (sum / (double)FALLS) << endl;
  }

  for (j = 0; j <= FALLS - 1; j++) {
    sum = 0;
    for (i = 0; i <= OBJECTS - 1; i++) {
      sum += g[i][j];
    }
    cout << "Average g for fall No " << (j + 1) << ": " << (sum / (double)OBJECTS) << endl;
  }

  sum = 0;
  for (i = 0; i <= OBJECTS - 1; i++) {
    for (j = 0; j <= FALLS - 1; j++) {
      sum += g[i][j];
    }
  }
  cout << "Overall average g: " << (sum / (double)(OBJECTS * FALLS));
```

```
  return 0;
}
```

3. Solution

```
#include <iostream>
using namespace std;
const int PLAYERS = 15;
const int MATCHES = 12;

int main() {
  int i, j, sum;

  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++) {
    sum = 0;
    for (j = 0; j <= MATCHES - 1; j++) {
      sum += points[i][j];
    }
    cout << "Total number of points for player No " << (i + 1) << ": " << sum << endl;
  }

  for (j = 0; j <= MATCHES - 1; j++) {
    sum = 0;
    for (i = 0; i <= PLAYERS - 1; i++) {
      sum += points[i][j];
    }
    cout << "Total number of points for match No " << (j + 1) << ": " << sum << endl;
  }
  return 0;
}
```

4. Solution

```
#include <iostream>
using namespace std;
const int CITIES = 20;
const int HOURS = 24;

int main() {
  int i, j;
  double sum;

  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++) {
    sum = 0;
    for (i = 0; i <= CITIES - 1; i++) {
      sum += temperatures[i][j];
    }
    if (sum / 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, sum;

  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++) {
    sum = 0;
    for (j = 0; j <= MATCHES - 1; j++) {
      sum += goals[i][j];
    }
    cout << names[i] << ": " << (sum / (double)MATCHES) << endl;
  }

  for (j = 0; j <= MATCHES - 1; j++) {
    sum = 0;
    for (i = 0; i <= PLAYERS - 1; i++) {
      sum += goals[i][j];
    }
    cout << "Match No " << (j + 1) << ": " << sum << endl;
  }
  return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
const int STUDENTS = 24;
const int LESSONS = 10;

int main() {
  int i, j, sum;

  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++) {
    sum = 0;
    for (j = 0; j <= LESSONS - 1; j++) {
      sum += grades[i][j];
    }
    average[i] = sum / (double)LESSONS;
    cout << names[i] << ": " << average[i] << endl;
  }

  for (j = 0; j <= LESSONS - 1; j++) {
    sum = 0;
    for (i = 0; i <= STUDENTS - 1; i++) {
      sum += grades[i][j];
    }
    cout << sum / (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, sum;

  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++) {
    sum = 0;
    for (j = 0; j <= JUDGES - 1; j++) {
      sum += score[i][j];
    }
    cout << artist_names[i] << ", " << song_titles[i] << ": " << sum << endl;
  }

  for (j = 0; j <= JUDGES - 1; j++) {
    sum = 0;
    for (i = 0; i <= ARTISTS - 1; i++) {
      sum += score[i][j];
    }
    cout << judge_names[j] << ": " << sum / (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, sum;


  int meter_reading[CONSUMERS][2];
  for (i = 0; i <= CONSUMERS - 1; i++) {
    cin >> meter_reading[i][0];
    cin >> meter_reading[i][1];
  }
```

```
  sum = 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;

    sum += consumed;
  }

  cout << sum << ", " << (sum * 0.07 + sum * 0.07 * VAT);
  return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
const int CURRENCIES = 4;
const int DAYS = 5;

int main() {
  int i, j;
  double average, sum, usd;

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

  string currency[4] = {"British Pound Sterling", "Euro", "Canadian Dollar", "Australian Dollar"};

  double rate[4][5] = { {1.579, 1.577, 1.572, 1.580, 1.584},
                        {1.269, 1.270, 1.265, 1.240, 1.255},
                        {0.895, 0.899, 0.884, 0.888, 0.863},
                        {0.811, 0.815, 0.822, 0.829, 0.819}
                      };

  for (i = 0; i <= CURRENCIES - 1; i++) {
    sum = 0;
    for (j = 0; j <= DAYS - 1; j++) {
      sum += rate[i][j];
    }
    average = sum / DAYS;
    cout << usd << " US dollars = " << (usd / average) << " " << currency[i] << "s" << endl;
  }
  return 0;
}
```

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, sum;

  string days[5] = {"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++) {
    sum = 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);
      }
      sum += gross_pay;
    }
    cout << days[j] << ", " << sum << endl;
  }
  return 0;
}
```

12. Solution

```
#include <iostream>
using namespace std;
const int ROWS = 3;
const int COLUMNS = 4;
const int ELEMENTS = ROWS * COLUMNS;

int main() {
  int i, j, k;

  int a[3][4] = { {9, 9, 2, 6},
                  {4, 1, 10, 11},
                  {12, 15, 7, 3}
                };

  int b[ELEMENTS];
  k = 0;
  for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS - 1; j++) {
      b[k] = a[i][j];
      k++;
    }
  }

  for (k = 0; k <= ELEMENTS - 1; k++) {
    cout << b[k] << " ";
  }
  return 0;
}
```

13. Solution

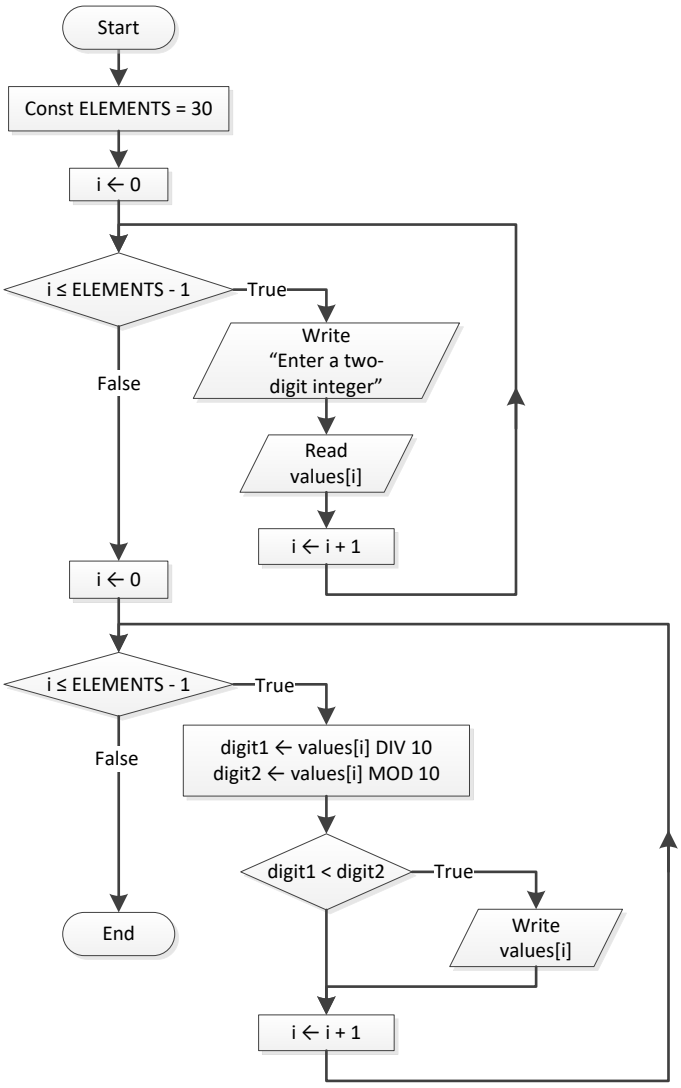
```
#include <iostream>
using namespace std;
const int ROWS = 3;
const int COLUMNS = 3;

int main() {
  int i, j, k;
```

```
  int a[9] = {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];
      k++;
    }
  }

  for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS - 1; j++) {
      cout << b[i][j] << "\t";
    }
    cout << endl;
  }
  return 0;
}
```

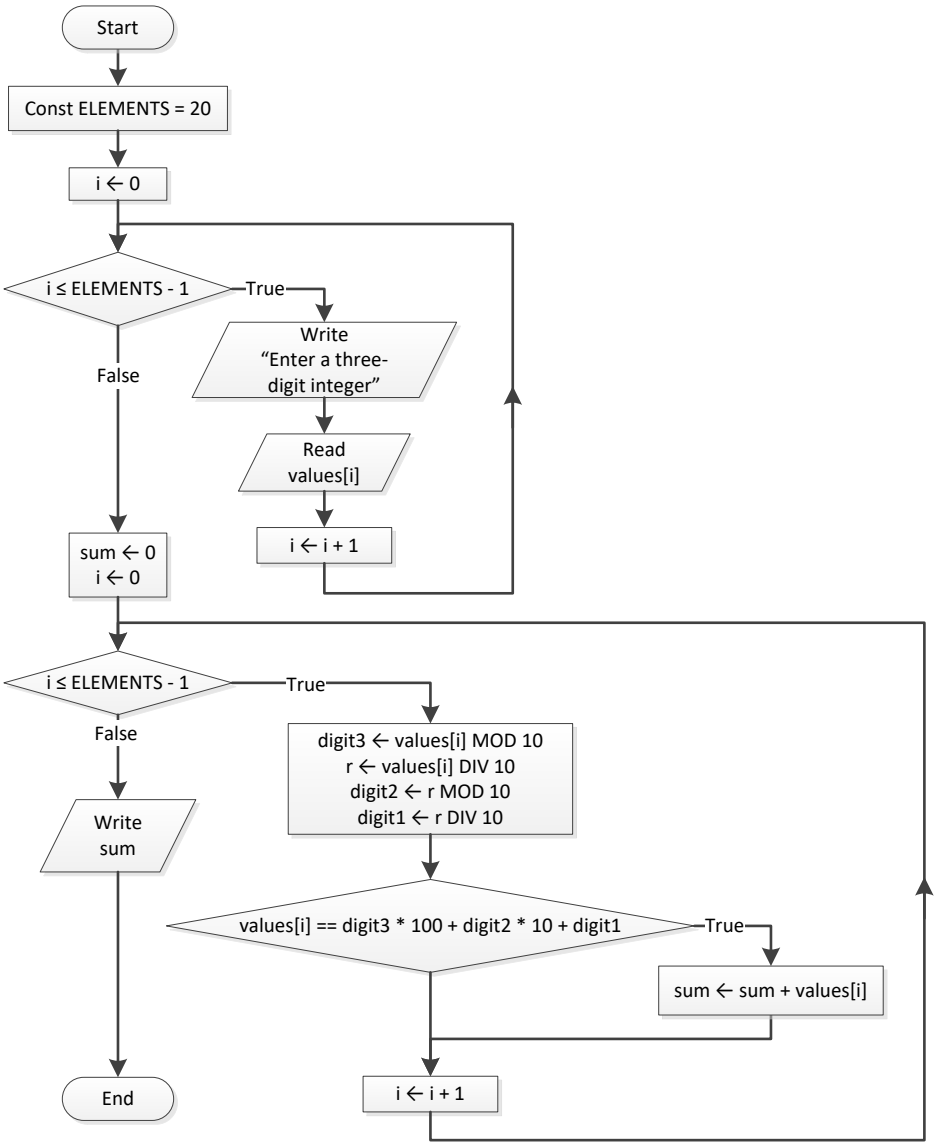
Chapter 36

36.4 Answers of Review Exercises

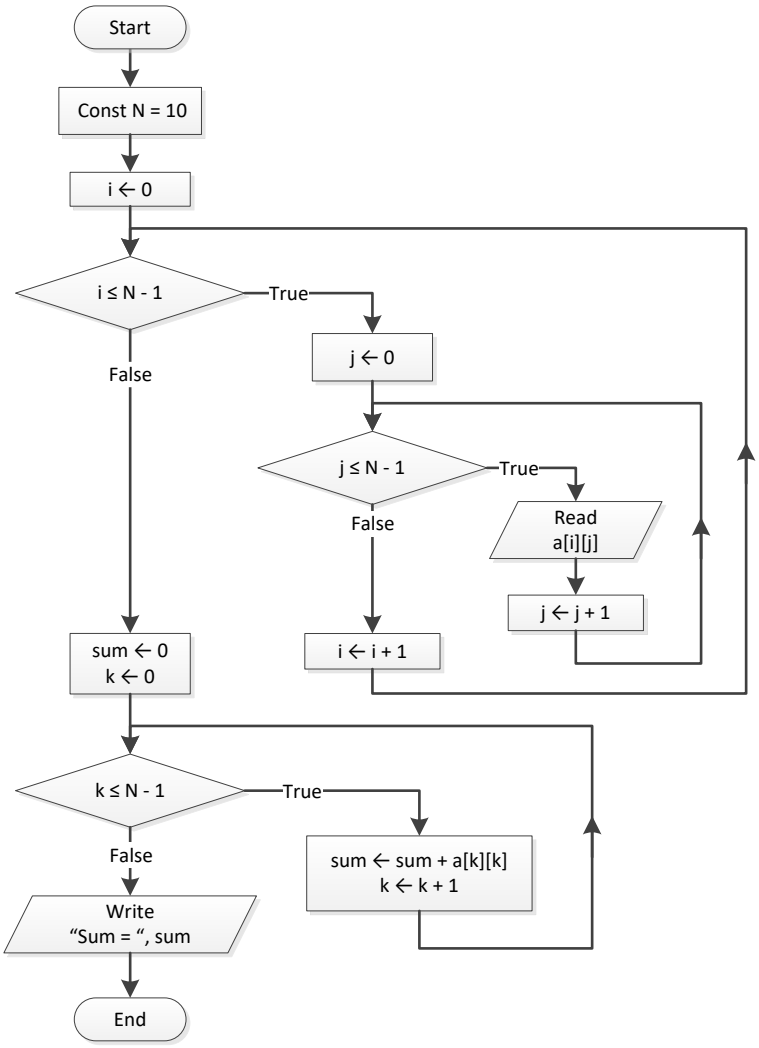
1. Solution



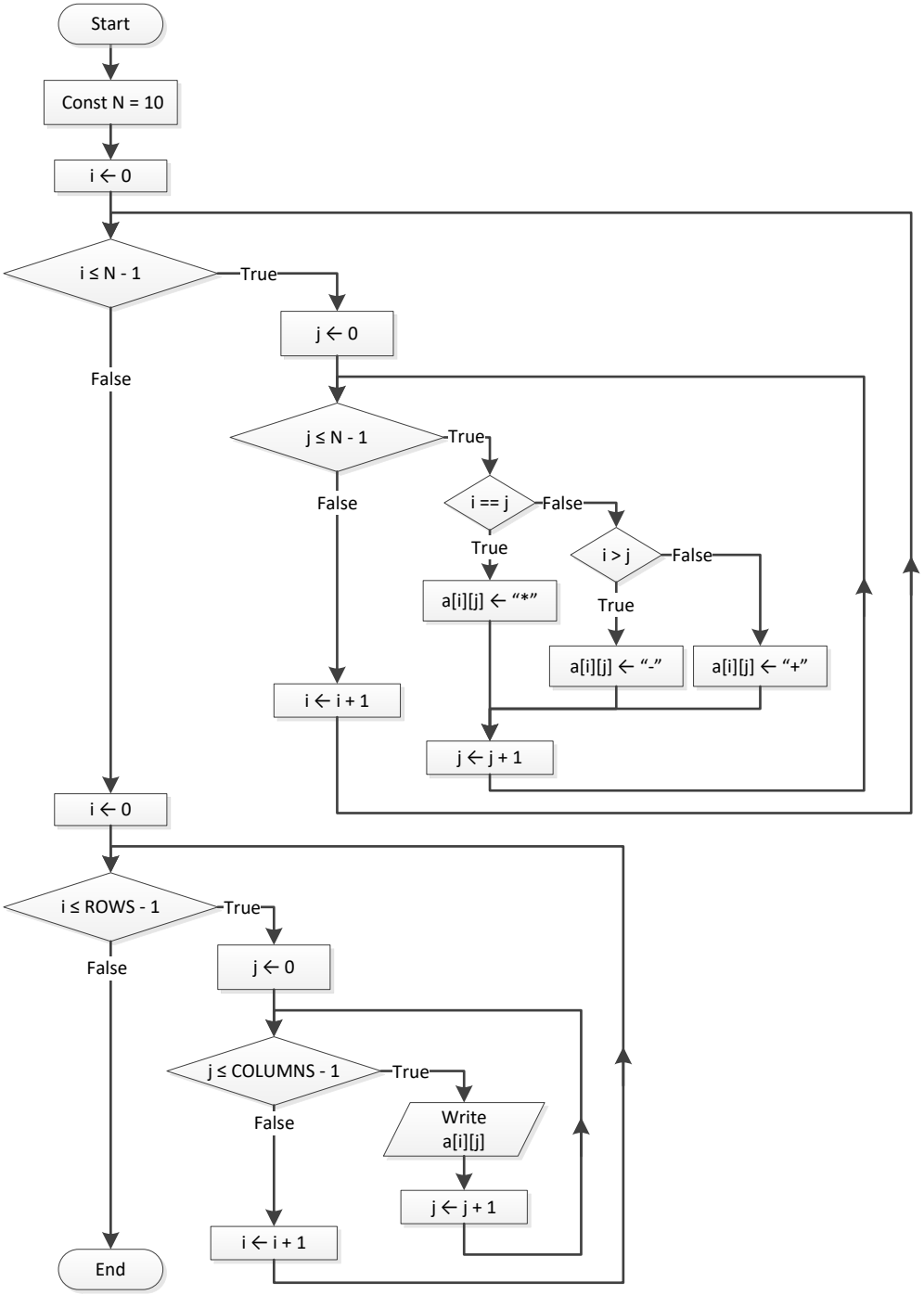
2. Solution


Start
Const ELEMENTS = 20
i ← 0
i ≤ ELEMENTS - 1
True
False
Write "Enter a three-digit integer"
Read values[i]
i ← i + 1
sum ← 0
i ← 0
i ≤ ELEMENTS - 1
True
False
digit3 ← values[i] MOD 10
r ← values[i] DIV 10
digit2 ← r MOD 10
digit1 ← r DIV 10
Write sum
values[i] == digit3 * 100 + digit2 * 10 + digit1
True
sum ← sum + values[i]
End
i ← i + 1

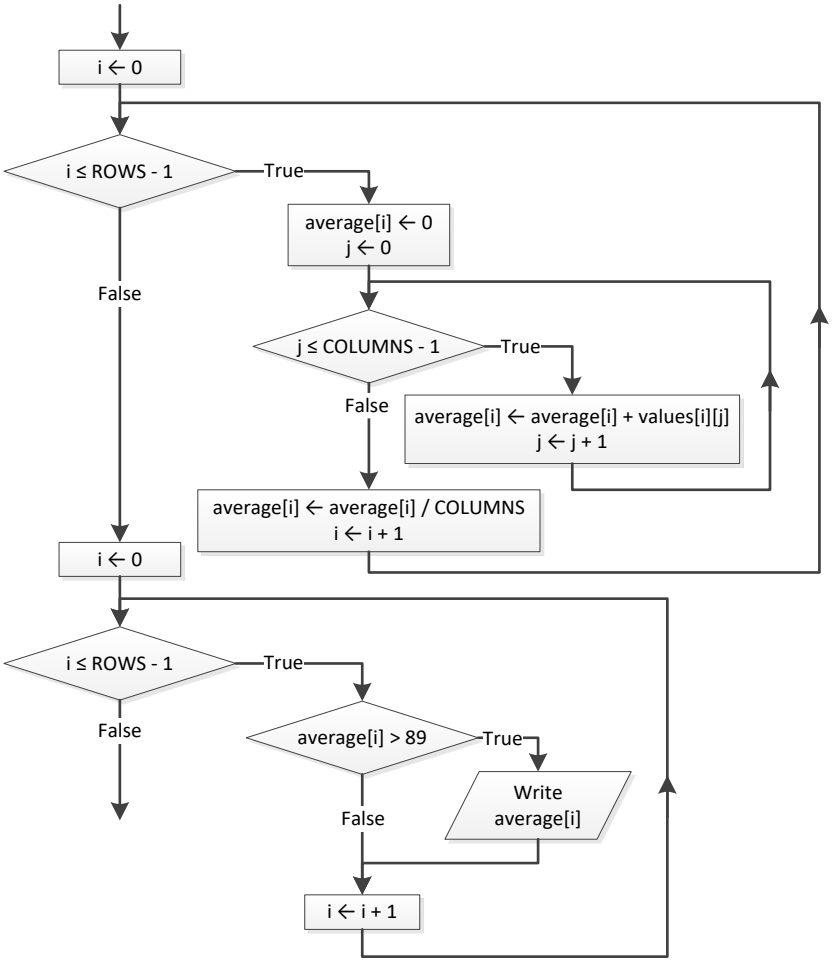
3. Solution


Start
Const N = 10
i ← 0
i ≤ N - 1
True
False
j ← 0
j ≤ N - 1
True
False
Read
a[i][j]
j ← j + 1
i ← i + 1
sum ← 0
k ← 0
k ≤ N - 1
True
False
sum ← sum + a[k][k]
k ← k + 1
Write
“Sum = “, sum
End

4. Solution


Start
Const N = 10
i ← 0
i ≤ N - 1
True
False
j ← 0
j ≤ N - 1
True
False
i == j
False
True
i > j
False
True
a[i][j] ← "*"
a[i][j] ← "-"
a[i][j] ← "+"
i ← i + 1
j ← j + 1
i ← 0
i ≤ ROWS - 1
True
False
j ← 0
j ≤ COLUMNS - 1
True
False
Write
a[i][j]
j ← j + 1
i ← i + 1
End

5. Solution



6. Solution

```
for (i = 0; i <= PEOPLE - 1; i++) {
  do {
    cin >> a[i];
  } while (a[i] % 2 == 2);
}
```

7. Solution

```
for (i = 0; i <= ELEMENTS - 1; i++) {
  cin >> a[i];
  while (a[i] < 0) {
    cout << "Error" << endl;
    cin >> a[i];
  }
}
```

8. Solution

```
i = 0;
S = 0;
cin >> a[i];
i++;
while (i < 90) {
  S += a[i - 1] * i;
  cin >> a[i];
  i++;
}
cout << S << endl;
while (i >= 0) {
  cout << a[i] << endl;
  i -= 5;
}
```

9. Solution

```
for (i = 0; i <= ROWS - 1; i++) {
  max = a[i][0];
  for (j = 1; j <= COLUMNS - 1; j++) {
    if (a[i][j] > max) {
      max = a[i][j];
    }
  }
  cout << max << endl;
}
```

10. 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];
    }
  }
}
```

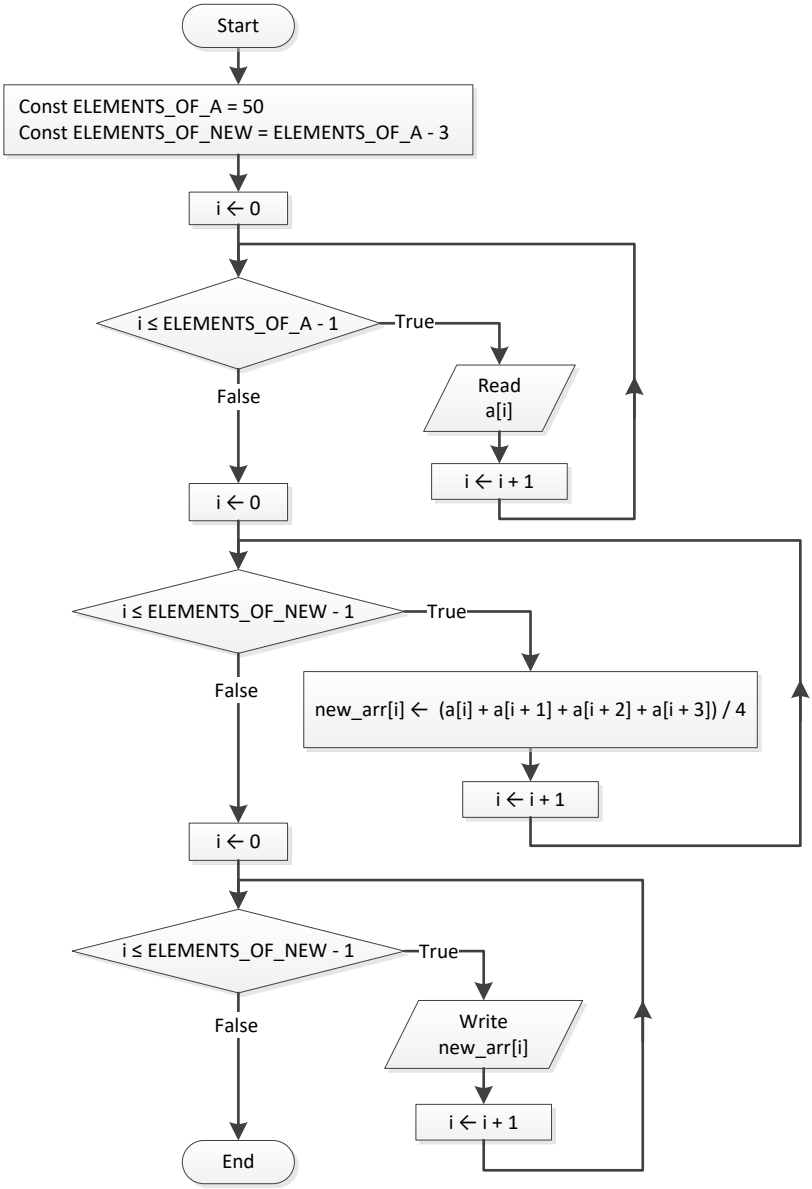
Chapter 37

37.7 Answers of Review Questions: True/False

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

37.8 Answers of Review Exercises

1. Solution

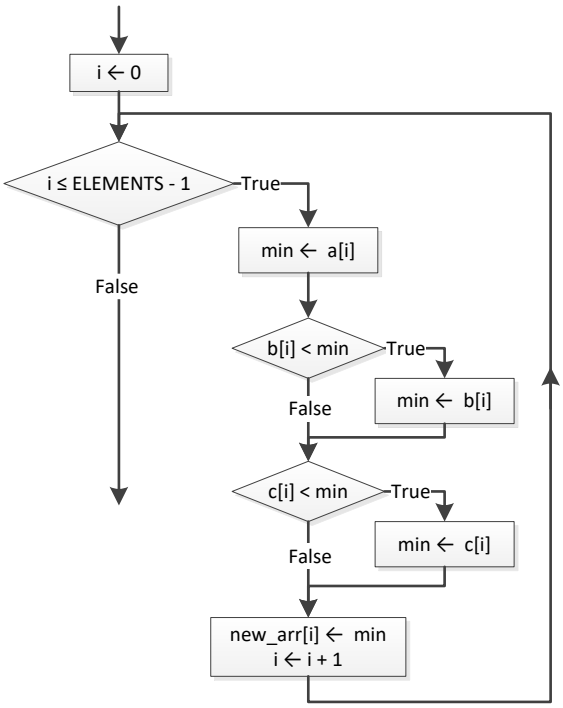



```
#include <iostream>
using namespace std;
const int ELEMENTS_OF_A = 50;
const int ELEMENTS_OF_NEW = ELEMENTS_OF_A - 3;

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;
}
```

2. Solution

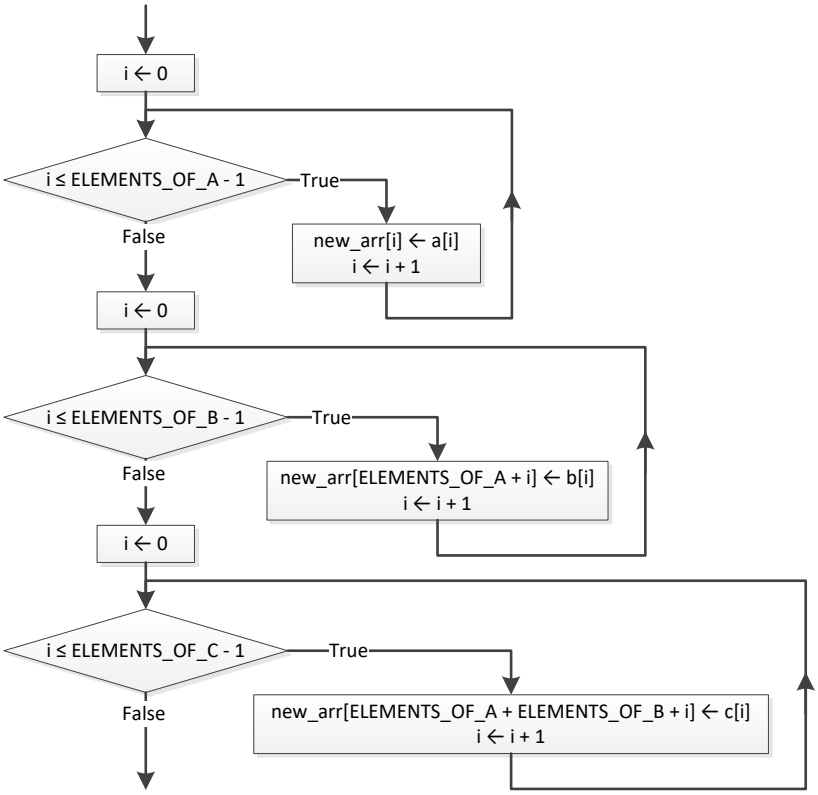
```
#include <iostream>
using namespace std;
const int ELEMENTS = 15;

int main() {
  int i;
  double min;

  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++) {
    min = a[i];
    if (b[i] < min) {
      min = b[i];
    }
    if (c[i] < min) {
      min = c[i];
    }
    new_arr[i] = min;
  }

  for (i = 0; i <= ELEMENTS - 1; i++) {
    cout << new_arr[i] << endl;
  }
  return 0;
}
```



3. 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;
}
```



4. Solution

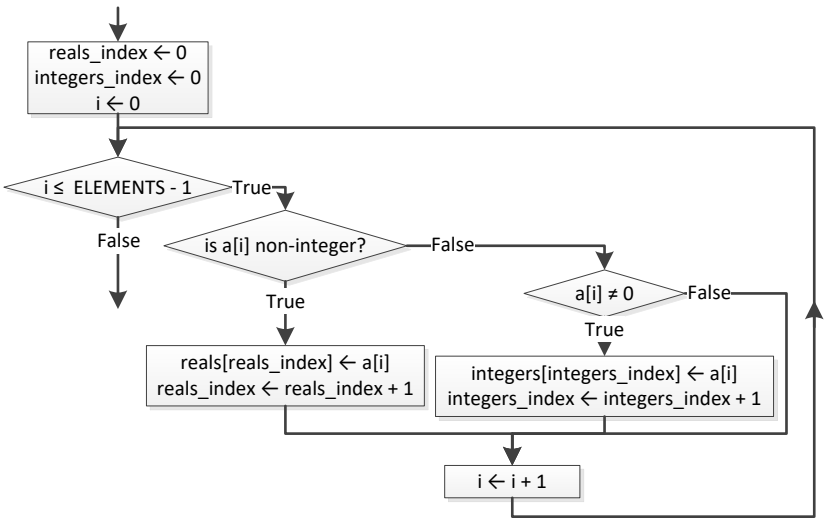
```
#include <iostream>
using namespace std;
const int COLUMNS_OF_A = 10;
const int COLUMNS_OF_B = 15;
const int COLUMNS_OF_C = 20;
const int ROWS = 5;
const int COLUMNS = COLUMNS_OF_A + COLUMNS_OF_B + COLUMNS_OF_C;

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;
}
```

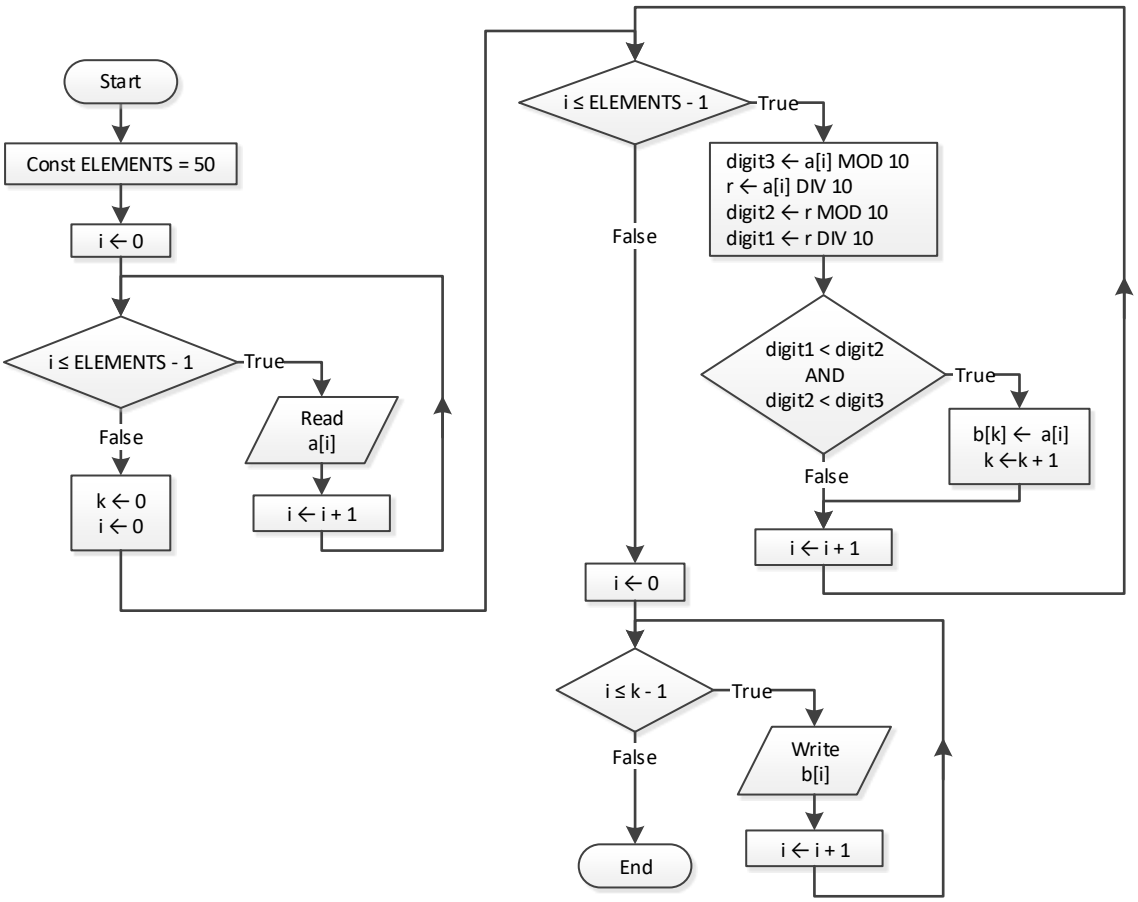
5. Solution

```
#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;
}
```



6. 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;
}
```

7. Solution

```
#include <iostream>
using namespace std;
const int PRODUCTS = 10;
const int CITIZENS = 1000;

int main() {
  int count_B, i, j, max;

  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;
  }
```

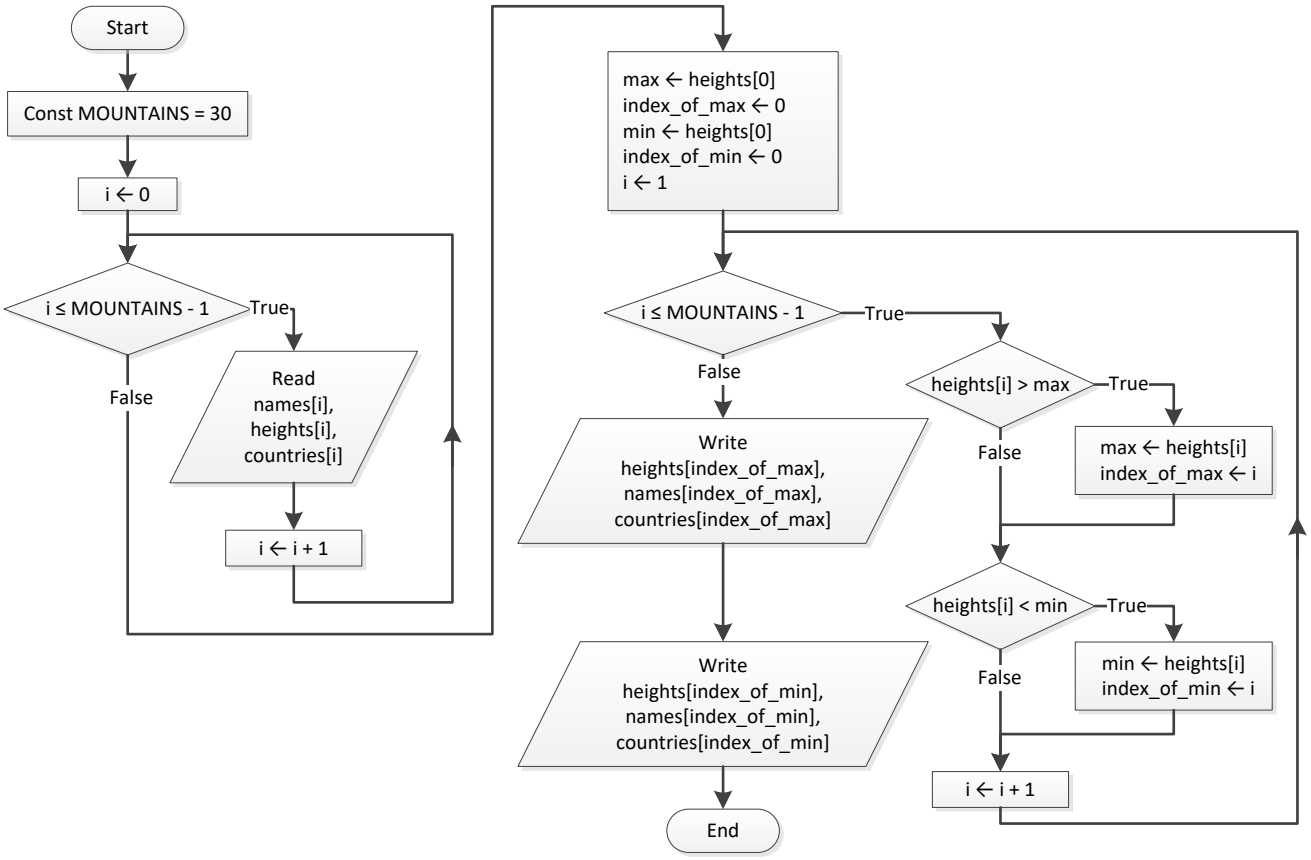
```
  max = count_A[0];
  for (i = 1; i <= PRODUCTS - 1; i++) {
    if (count_A[i] > max) {
      max = count_A[i];
    }
  }
  for (i = 0; i <= PRODUCTS - 1; i++) {
    if (count_A[i] == max) {
      cout << prod_names[i] << endl;
    }
  }
  return 0;
}
```

8. Solution

```
#include <iostream>
using namespace std;
const int US_CITIES = 20;
const int CANADIAN_CITIES = 20;

int main() {
  int i, j, min_j;
  double min;

  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++) {
    min = distances[i][0];
    min_j = 0;
    for (j = 1; j <= CANADIAN_CITIES - 1; j++) {
      if (distances[i][j] < min) {
        min = distances[i][j];
        min_j = j;
      }
    }
```

```
    cout << "Closest Canadian city to " << us_names[i] << " is " << canadian_names[min_j] << endl;
  }
  return 0;
}
```

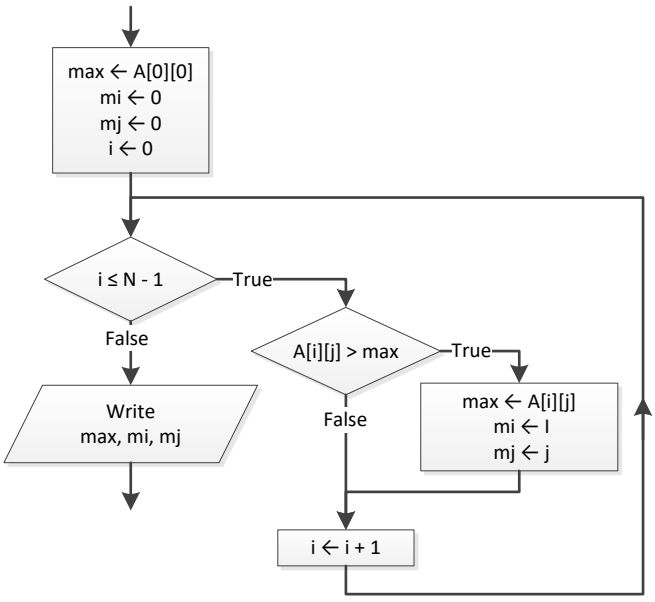
9. Solution




```
#include <iostream>
using namespace std;
const int MOUNTAINS = 30;

int main() {
  int i, index_of_max, index_of_min;
  double max, min;

  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];
  }

  max = heights[0];
```

```
  index_of_max = 0;
  min = heights[0];
  index_of_min = 0;
  for (i = 1; i <= MOUNTAINS - 1; i++) {
    if (heights[i] > max) {
      max = heights[i];
      index_of_max = i;
    }
    if (heights[i] < min) {
      min = heights[i];
      index_of_min = i;
    }
  }

  cout << heights[index_of_max] << ", " << names[index_of_max] << ", " << countries[index_of_max];
  cout << endl;
  cout << heights[index_of_min] << ", " << names[index_of_min] << ", " << countries[index_of_min];
  return 0;
}
```

10. Solution



11. Solution

```
#include <iostream>
using namespace std;
const int TEAMS = 26;
const int GAMES = 15;

int main() {
  int i, j, m_i, max;

  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;
      }
    }
  }

  max = points[0];
  m_i = 0;
  for (i = 1; i <= TEAMS - 1; i++) {
    if (points[i] > max) {
      max = points[i];
      m_i = i;
    }
  }

  cout << names[m_i];
  return 0;
}
```

12. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
const int OBJECTS = 10;
const int FALLS = 20;

int main() {
  int i, j;
  double maximum, minimum;

  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];
    }
  }
```

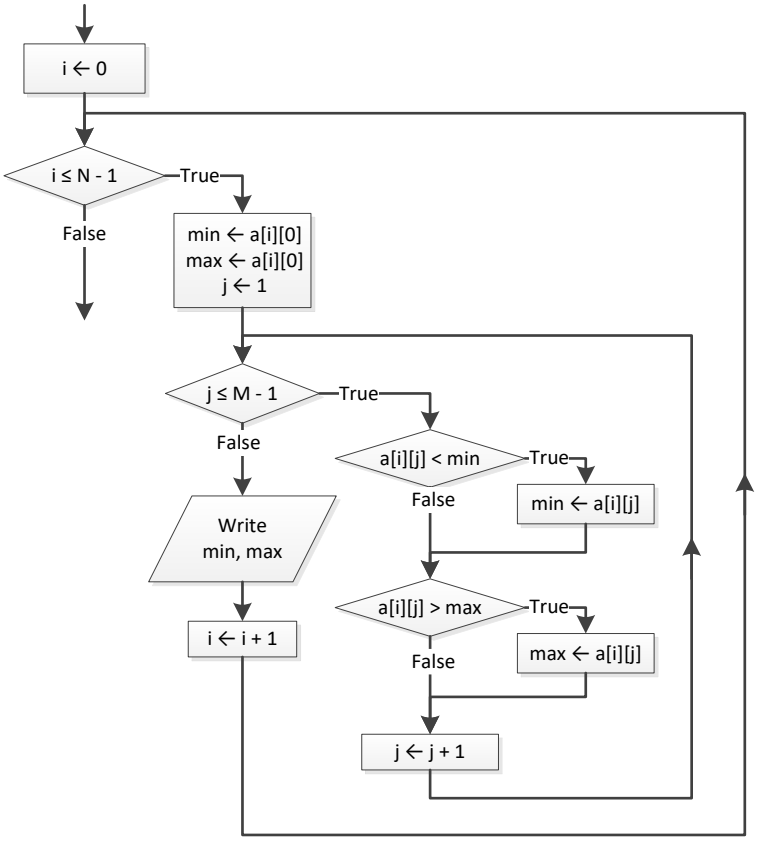
```
  double g[OBJECTS][FALLS];
  for (i = 0; i <= OBJECTS - 1; i++) {
    for (j = 0; j <= FALLS - 1; j++) {
      g[i][j] = 2 * heights[i][j] / pow(times[i][j], 2);
    }
  }

  double min[OBJECTS];
  double max[OBJECTS];
  for (i = 0; i <= OBJECTS - 1; i++) {
    min[i] = g[i][0];
    max[i] = g[i][0];
    for (j = 1; j <= FALLS - 1; j++) {
      if (g[i][j] < min[i]) {
        min[i] = g[i][j];
      }
      if (g[i][j] > max[i]) {
        max[i] = g[i][j];
      }
    }
  }

  for (i = 0; i <= OBJECTS - 1; i++) {
    cout << min[i] << ", " << max[i] << endl;
  }

  maximum = max[0];
  minimum = min[0];
  for (i = 1; i <= OBJECTS - 1; i++) {
    if (max[i] > maximum) {
      maximum = max[i];
    }
    if (min[i] < minimum) {
      minimum = min[i];
    }
  }

  cout << minimum << ", " << maximum;
  return 0;
}
```

13. Solution

```
#include <iostream>
using namespace std;
const int STATIONS = 10;
const int DAYS = 365;

int main() {
  int i, j, m_i;
  double min;

  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;
  }

  min = average[0];
  m_i = 0;
  for (i = 1; i <= STATIONS - 1; i++) {
    if (average[i] < min) {
      min = average[i];
      m_i = i;
    }
  }
  cout << names[m_i];
  return 0;
}
```

14. Solution



15. 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 == false) break;
  }

  cout << "Gold: " << names[0] << endl;
  cout << "Silver: " << names[1] << endl;
  cout << "Bronze: " << names[2];
  return 0;
}
```

16. 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];
```

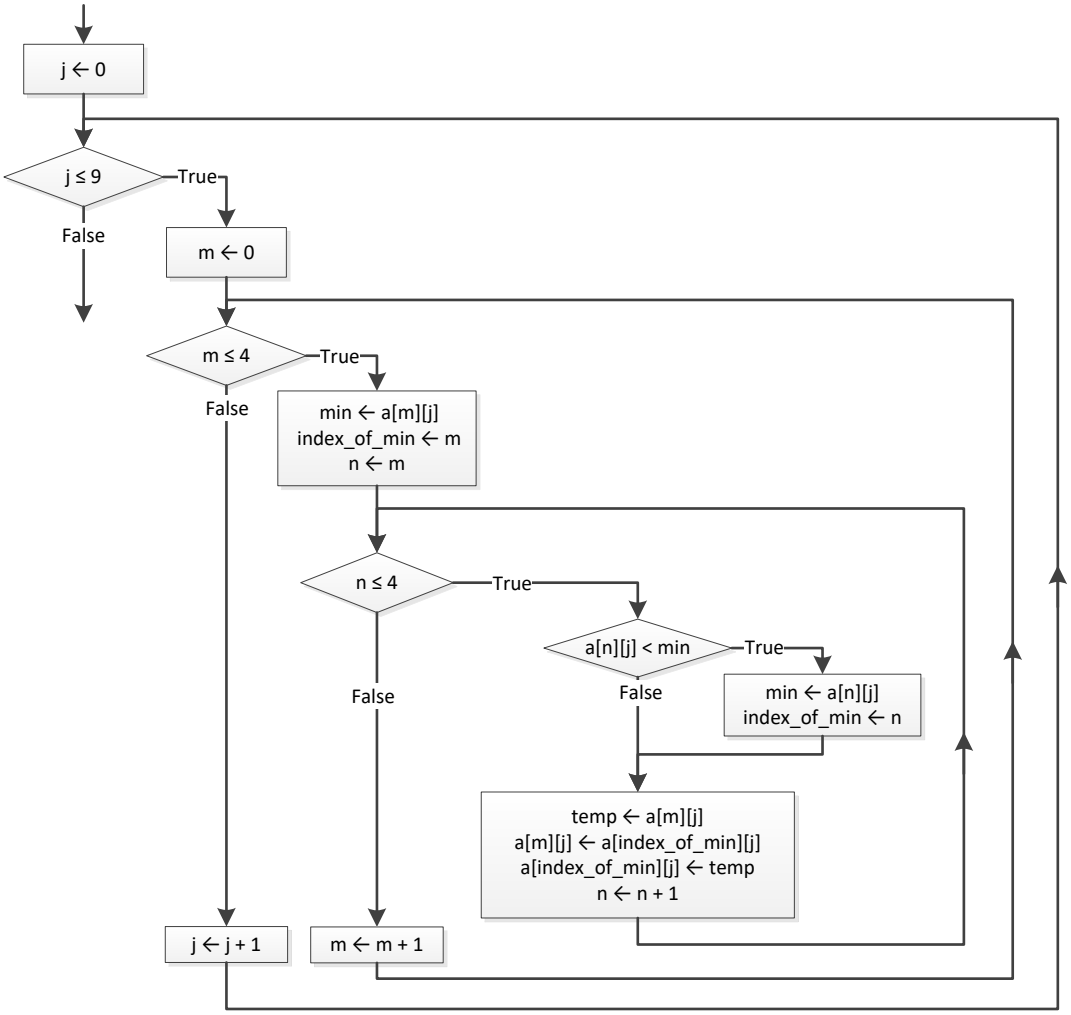
```
  for (i = 0; i <= PEOPLE - 1; i++) {
    cout << "Enter name for person No. " << (i + 1) << ": ";
    cin >> names[i];
    cout << "Enter height for person No. " << (i + 1) << ": ";
    cin >> heights[i];
  }

  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;
}
```

17. Solution

```
#include <iostream>
using namespace std;
const int ARTISTS = 12;
const int JUDGES = 10;

int main() {
  int i, j, m, max, min, 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 sum[ARTISTS];
  for (i = 0; i <= ARTISTS - 1; i++) {
    sum[i] = 0;
    for (j = 1; j <= JUDGES - 1; j++) {
      sum[i] += score[i][j];
    }
  }

  for (i = 0; i <= ARTISTS - 1; i++) {
    min = score[i][0];
    max = score[i][0];
    for (j = 1; j <= JUDGES - 1; j++) {
      if (score[i][j] < min) {
        min = score[i][j];
      }
      if (score[i][j] > max) {
        max = score[i][j];
      }
    }
    sum[i] = sum[i] - min - max;
    cout << sum[i] << endl;
  }

  for (m = 1; m <= ARTISTS - 1; m++) {
    for (n = ARTISTS - 1; n >= m; n--) {
      if (sum[n] > sum[n - 1]) {
        temp = sum[n];
        sum[n] = sum[n - 1];
        sum[n - 1] = temp;

        temp_str = artist_names[n];
        artist_names[n] = artist_names[n - 1];
        artist_names[n - 1] = temp_str;
      }
      else if (sum[n] == sum[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] << ", " << sum[i] << endl;
  }
  return 0;
}
```

18. Solution



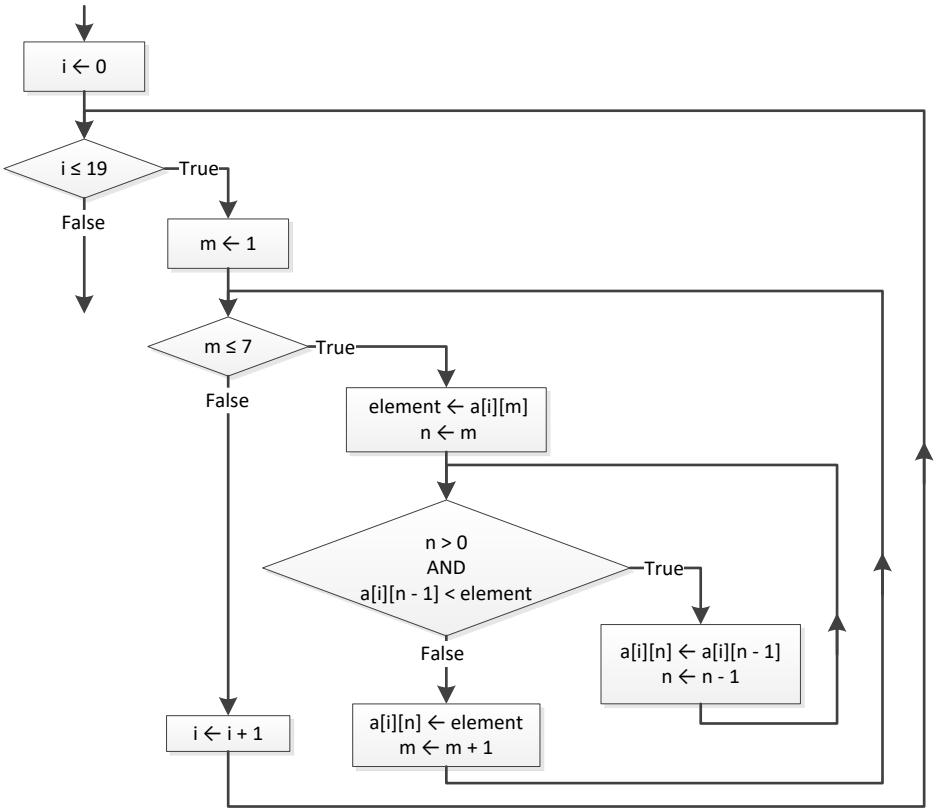
19. Solution

```
#include <iostream>
using namespace std;
const int PEOPLE = 10;
const int PUZZLES = 8;

int main() {
  int i, index_of_min, j, m, n;
  double min, 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++) {
    min = times[i][m];
    index_of_min = m;
    for (n = m; n <= PUZZLES - 1; n++) {
      if (times[i][n] < min) {
        min = 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++) {
  min = average[m];
  index_of_min = m;
  for (n = m; n <= PEOPLE - 1; n++) {
    if (average[n] < min) {
      min = 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];
  return 0;
}
```

20. Solution



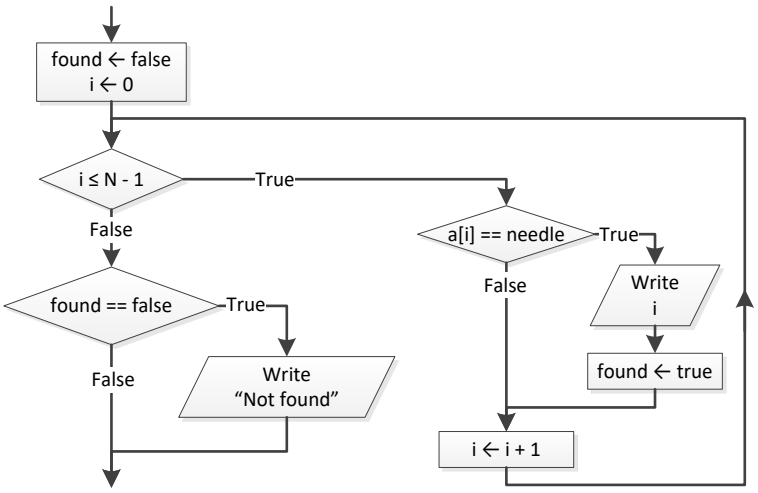
21. Solution

```
#include <iostream>
using namespace std;
const int CITIES = 5;
const int HOURS = 48;

int main() {
  int i, j, m, m_i, m_j, n;
  double max, element_1;
  string element_2;

  string names[CITIES];
  double CO2[CITIES][HOURS];
  for (i = 0; i <= CITIES - 1; i++) {
    cin >> names[i];
    for (j = 0; j <= HOURS - 1; j++) {
      cin >> CO2[i][j];
    }
  }
```

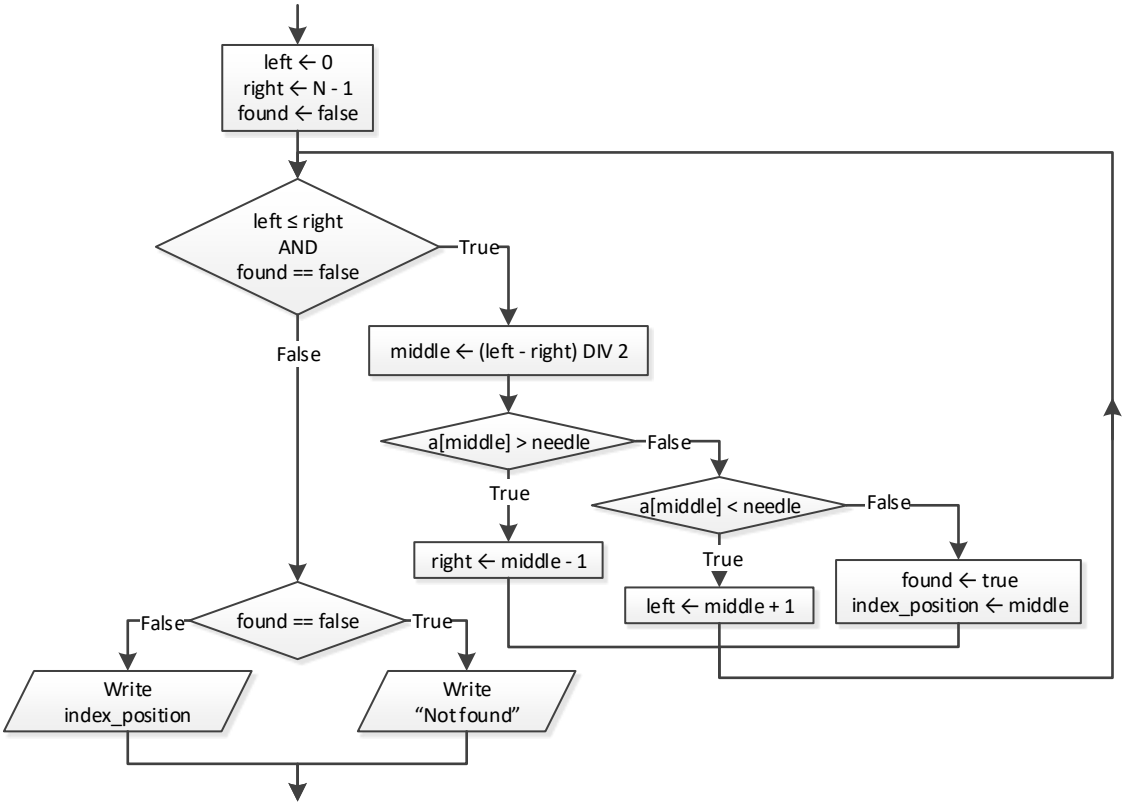
```
double average_per_hour[CITIES];
for (i = 0; i <= CITIES - 1; i++) {
  average_per_hour[i] = 0;
  for (j = 0; j <= HOURS - 1; j++) {
    average_per_hour[i] += CO2[i][j];
  }
  average_per_hour[i] /= HOURS;
}

for (i = 0; i <= CITIES - 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 <= CITIES - 1; i++) {
    average_per_city[j] += CO2[i][j];
  }
  average_per_city[j] /= CITIES;
}

for (j = 0; j <= HOURS - 1; j++) {
  cout << average_per_city[j] << endl;
}

max = average_per_city[0];
m_j = 0;
for (j = 1; j <= HOURS - 1; j++) {
  if (average_per_city[j] > max) {
    max = average_per_city[j];
    m_j = j;
  }
}
cout << m_j << endl;

max = CO2[0][0];
m_i = 0;
m_j = 0;
for (i = 0; i <= CITIES - 1; i++) {
  for (j = 0; j <= HOURS - 1; j++) {
    if (CO2[i][j] > max) {
      max = CO2[i][j];
      m_i = i;
      m_j = j;
    }
  }
}
cout << m_j << ", " << names[m_i] << endl;

for (m = 1; m <= CITIES - 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;
}
```

22. Solution



23. Solution



24. Solution

```
#include <iostream>
using namespace std;
const int TEAMS = 10;
const int GAMES = 16;

int main() {
  int i, j, sum;
  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 (cin.fail() == true || goals_scored[i][j] < 0) {
        cout << "Error! Enter goals scored: ";
        cin.clear();
        cin.ignore(100, '\n');
```

```
        cin >> goals_scored[i][j];
      }

      cout << "Enter goals let in: ";
      cin >> goals_let_in[i][j];
      while (cin.fail() == true || goals_let_in[i][j] < 0) {
        cout << "Error! Enter goals let in: ";
        cin.clear();
        cin.ignore(100, '\n');
        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 {
    sum = 0;
    for (j = 0; j <= GAMES - 1; j++) {
      if (goals_scored[i][j] > goals_let_in[i][j]) {
        sum += 3;
      }
      else if (goals_scored[i][j] == goals_let_in[i][j]) {
        sum += 1;
      }
    }
    cout << sum << endl;
  }
  return 0;
}
```

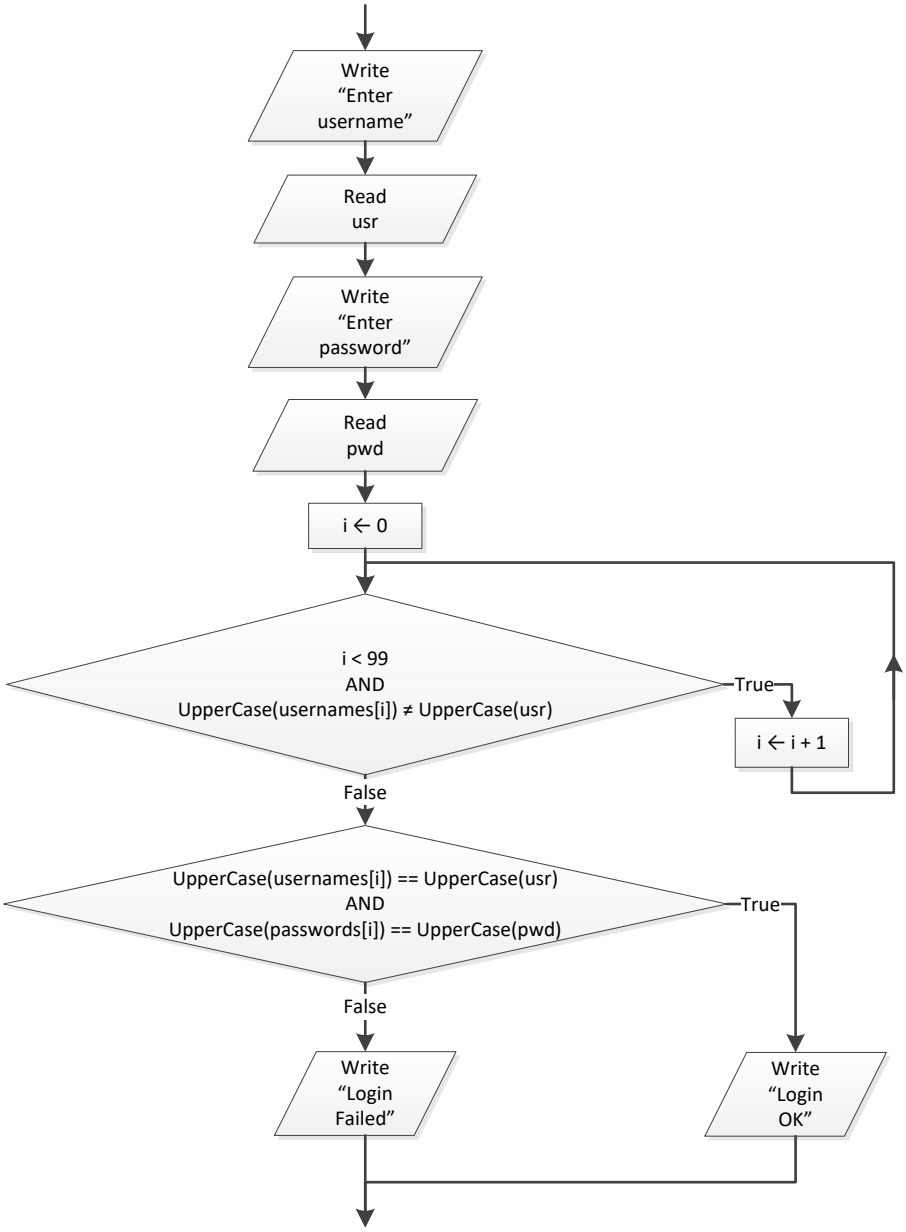
25. Solution

```
#include <iostream>
using namespace std;
const int CLASS1 = 20;
const int CLASS2 = 25;

int main() {
  int i, left, m, middle, n, right;
  string element, needle;
  bool found;

  cout << "Class A" << endl;
  string names1[CLASS1];
```

```
for (i = 0; i <= CLASS1 - 1; i++) {
  cout << "Enter name: ";
  cin >> names1[i];
}
cout << "Class B" << endl;
string names2[CLASS2];
for (i = 0; i <= CLASS2 - 1; i++) {
  cout << "Enter name: ";
  cin >> names2[i];
}

//Insertion sort algorithm
for (m = 1; m <= CLASS1 - 1; m++) {
  element = names1[m];
  n = m;
  while (n > 0 && names1[n - 1] > element) {
    names1[n] = names1[n - 1];
    n--;
  }
  names1[n] = element;
}
for (m = 1; m <= CLASS2 - 1; m++) {
  element = names2[m];
  n = m;
  while (n > 0 && names2[n - 1] > element) {
    names2[n] = names2[n - 1];
    n--;
  }
  names2[n] = element;
}

cout << "\nClass A" << endl;
for (i = 0; i <= CLASS1 - 1; i++) {
  cout << names1[i] << endl;
}
cout << "\nClass B" << 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 == false) {
  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 == true) {
    cout << "Student found in class No 1" << endl;
  }
  else {
    left = 0;
    right = CLASS2 - 1;
    while (left <= right && found == false) {
      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 == true) {
      cout << "Student found in class No 2" << endl;
    }
    else {
      cout << "Student not found in either class" << endl;
    }
  }
  return 0;
}
```

26. Solution




```
cout << "Enter username: ";
cin >> usr;
cout << "Enter password: ";
cin >> pwd;

i = 0;
while (i < 99 && to_upper_copy(usernames[i]) != to_upper_copy(usr)) {
  i++;
}
```

```
if (to_upper_copy(usernames[i]) == to_upper_copy(usr) &&
    to_upper_copy(passwords[i]) == to_upper_copy(pwd)) {

  cout << "Login OK!" << endl;
}
else {
  cout << "Login Failed!" << endl;
}
```

27. Solution

```
cout << "Enter a value to search: ";
cin >> value;  //variable "value" must be declared as long int

found = false;

//Check if entered value is a valid nine-digit SSN
if (cin.fail() != true && value >= 100000000 && value <= 999999999) {
  i = 0;
  while (i < 999 && SSNs[i] != value) {
    i++;
  }

  if (SSNs[i] == value) {
    found = true;
    cout << names[i] << endl;
  }
}
else {
  cin.clear();
  cin >> value_str;    //Do not ignore data input using the cin.ignore(100, 'n') statement.
                       //Send data input to value_str variable instead.

  for (i = 0; i <= 999; i++) {
    if (names[i] == value) {
      cout << names[i] << endl;
      found = true;
    }
  }
}

if (found == false) {
  cout << "This value does not exist" << endl;
}
```

28. Solution

```
#include <iostream>
using namespace std;
const int STUDENTS = 12;
const int LESSONS = 6;

int main() {
```

```
  int i, j;
  bool found;

  int grades[STUDENTS][LESSONS];
  for (i = 0; i <= STUDENTS - 1; i++) {
    for (j = 0; j <= LESSONS - 1; j++) {
      cin >> grades[i][j];
    }
  }

  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;
    }
  }

  if (found == true) {
    cout << "There is at least one student that has an average value below 70" << endl;
  }
  return 0;
}
```

Chapter 38

38.4 Review Questions: True/False

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

Chapter 39

39.5 Review Questions: True/False

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

39.6 Review Exercises

1. Solution

```
int find_max(int a, int b) {
  int max;
  if (a > b) {
    max = a;
  }
  else {
    max = b;
  }
  return max;
}
```

2. Solution

Step	Statement	Main Code		Function 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	24 is displayed				

3. Solution

Step	Statement	Main Code		Function sss()		
		s	i	a	sum	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	sum = 0			2	**0**	?
11	k = 1			2	0	**1**
12	k <= a			True		
13	sum += k			2	**1**	1
14	k++			2	1	**2**
15	k <= a			True		
16	sum += k			2	**3**	2
17	k++			2	3	**3**
18	k <= a			False		

19	return sum	**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	sum = 0			4	**0**	?
29	k = 1			4	0	**1**
30	k <= a			True		
31	sum += k			4	**1**	1
32	k++			4	1	**2**
33	k <= a			True		
34	sum += k			4	**3**	2
35	k++			4	3	**3**
36	k <= a			True		
37	sum += k			4	**6**	4
38	k++			4	6	**4**
39	k <= a			True		
40	sum += k			4	**10**	4
41	k++			4	10	**5**
42	k <= a			False		
43	return sum	**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	16 is displayed				

4. Solution

Step	Statement	Main Code				Function 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	2 1 4 is displayed					
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	3 3 9 is displayed					
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	4 5 13 is displayed					
23	a += 2	12	4	**7**	13		
24	m++	12	**5**	7	13		
25	while (a < 6)	False					

5. 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;
}
```

6. Solution

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

  min = a;
  if (b < min) {
    min = b;
  }
  return min;
}

int main() {
  double temp1, temp2, x1, x2, x3, x4;

  cout << "Enter four numbers: ";
  cin >> x1;
  cin >> x2;
  cin >> x3;
  cin >> 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));
  return 0;
}
```

7. 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);
  return 0;
}
```

8. 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 (cin.fail() == true || weight < 0) {
    cout << "Error! Enter your weight (in pounds): ";
    cin.clear();
    cin.ignore(100, '\n');
    cin >> weight;
  }
```

```
  cout << "Enter your age: " << endl;
  cin >> age;
  while (cin.fail() == true || age < 18) {
    cout << "Error! Enter your age: ";
    cin.clear();
    cin.ignore(100, '\n');
    cin >> age;
  }

  cout << "Enter your height (in inches): " << endl;
  cin >> height;
  while (cin.fail() == true || height < 0) {
    cout << "Error! Enter your height (in inches): " << endl;
    cin.clear();
    cin.ignore(100, '\n');
    cin >> height;
  }

  cout << bmi(weight, height) << endl;
  return 0;
}
```

Chapter 40

40.5 Review Questions: True/False

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

40. 6 Review Exercises

1. Solution

Step	Statement	Main Code		Function 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"	The message “3 is odd” is displayed		
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"	The message “7 is odd” is displayed		
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"	The message “9 is odd” is displayed		
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"	The message “2 is even” is displayed		
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"	The message “4 is even” is displayed		
31	i++	**6**	4	
32	i <= 5	False		

2. *Solution*

Step	Statement	Main Code		Function 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	0 is displayed			
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	93 is displayed			
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	96 is displayed			
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	99 is displayed			
25	x = 4 * y	**132**	33		
26	y++	132	**34**		
27	while (x % y < 30)	False			

3. Solution

Step	Statement	Main Code		Function 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	20 is displayed				
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`	56 is displayed				
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`	120 is displayed				
52	`i++`	**4**	4			
53	`i <= 3`	False				

4. Solution

```
void maximum(double a, double b, double c, double d, double e) {
  double max;

  max = a;
```

```
  if (b > max) {
    max = b;
  }
  if (c > max) {
    max = c;
  }
  if (d > max) {
    max = d;
  }
  if (e > max) {
    max = e;
  }
  cout << max << endl;
}
```

5. 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;
```

```
}
```

6. 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;

  do {
    display_menu();

    cin >> choice;

    if (choice == 3) {
      cout << "Bye!" << endl;
    }
    else {
      cout << "Enter distance: " << endl;
      cin >> distance;
      if (choice == 1) {
        meters_to_miles(distance);
      }
      else {
        miles_to_meters(distance);
      }
    }
  } while (choice != 3);
  return 0;
}
```

7. 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);
  return 0;
}
```

Chapter 41

41.10 Review Questions: True/False

1. true
2. true
3. true
4. false
5. true
6. false
7. false
8. false
9. true
10. false
11. true
12. true
13. true
14. false
15. true
16. true
17. false
18. false
19. false
20. false
21. true
22. true
23. false
24. true
25. true
26. true
27. false
28. fasle

41.11 Review Exercises

1. Solution

The value 5 is displayed

2. Solution

The value 14 is displayed

3. Solution

The value 14 is displayed

4. Solution

Step	Statement	Main Code				Function swap()		
		a	m	k	x	x	y	temp
1	`cin >> k`	?	?	**12**	?			
2	`m = 1`	?	**1**	12	?			
3	`a = 1`	**1**	1	12	?			
4	`while (a < 8)`	True						
5	`if (k % m != 0)`	False						
6	`x = a + m + (int)(a - m)`	1	1	12	**2**			
7	`cout << m << " " << a << " " << x`	1 1 2 is displayed						
8	`a += 2`	**3**	1	12	2			

9	m++	3	**2**	12	2			
10	swap (a, m)					**3**	**2**	?
11	temp = x					3	2	**3**
12	x = y					**2**	2	3
13	y = temp					2	**3**	3
14	while (a < 8)	**2**	**3**	12	2			
		True						
15	if (k % m != 0)	False						
16	x = a + m + (int)(a - m)	2	3	12	**4**			
17	cout << m << " " << a << " " << x	3 2 4 is displayed						
18	a += 2	**4**	3	12	4			
19	m++	4	**4**	12	4			
20	swap (a, m)					**4**	**4**	?
21	temp = x					4	4	**4**
22	x = y					**4**	4	4
23	y = temp					4	**4**	4
24	while (a < 8)	**4**	**4**	12	4			
		True						
25	if (k % m != 0)	False						
26	x = a + m + (int)(a - m)	4	4	12	**8**			
27	cout << m << " " << a << " " << x	4 4 8 is displayed						
28	a += 2	**6**	4	12	8			
29	m++	6	**5**	12	8			
30	swap (a, m)					**6**	**5**	?
31	temp = x					6	5	**6**
32	x = y					**5**	5	6
33	y = temp					5	**6**	5
34	while (a < 8)	**5**	**6**	12	8			
		True						
35	if (k % m != 0)	False						
36	x = a + m + (int)(a - m)	5	6	12	**10**			
37	cout << m << " " << a << " " << x	6 5 10 is displayed						
38	a += 2	**7**	6	12	10			
39	m++	7	**7**	12	10			
40	swap (a, m)					**7**	**7**	?

41	temp = x					7	7	**7**
42	x = y					**7**	7	7
43	y = temp					7	**7**	7
44	while (a < 8)	**7**	**7**	12	10			
		True						
45	if (k % m != 0)	True						
46	x = a % m	7	7	12	**0**			
47	swap (m, a)					**7**	**7**	?
48	temp = x					7	7	**7**
49	x = y					**7**	7	7
50	y = temp					7	**7**	7
51	cout << m << " " << a << " " << x	**7**	**7**	12	0			
		7 7 0 is displayed						
52	a += 2	**9**	7	12	0			
53	m++	9	**8**	12	0			
54	swap (a, m)					**9**	**8**	?
55	temp = x					9	8	**9**
56	x = y					**8**	8	9
57	y = temp					8	**9**	9
58	while (a < 8)	**8**	**9**	12	0			
		False						

5. Solution

"hellohellohello" is displayed

6. Solution

The value 15 is displayed

7. Solution

11 4 is displayed

8. Solution

```
#include <iostream>
using namespace std;
const int STUDENTS = 10;
const int LESSONS = 5;

//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 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];
    }
  }
}

#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;
}
```

9. Solution

```
#include <iostream>
#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;
}

int part2(string message) {
  int last_pos = message.length() - 1;
```

```
  return last_pos;
}

string part3(string message, int last_pos) {
  string letter, message_clean;
  int i;

  message_clean = "";
  for (i = 0; i <= last_pos; i++) {
    letter = message.at(i);
    if (letter != " " && letter != "," && letter != "." && letter != "?") {
      message_clean += letter;
    }
  }
  return message_clean;
}

bool part4(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; //or you can write  j = part2(message_clean);
  palindrome = true;
  for (i = 0; i <= middle_pos; i++) {
    left_letter = message_clean.at(i);
    right_letter = message_clean.at(j);
    if (left_letter != right_letter) {
      palindrome = false;
      break;
    }
    j--;
  }
  return palindrome;
}

bool part5(string message) {
  int last_pos;
  string message_clean;
  bool palindrome;

  last_pos = part2(message);
  message_clean = part3(message, last_pos);
  palindrome = part4(message_clean);
  return palindrome;
}

int main() {
  string message;
  bool palindrome;

  message = part1();
  palindrome = part5(message);
```

```
  if (palindrome == true) {
    cout << "The message is palindrome" << endl;
  }
  return 0;
}
```

10. Solution

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

  cin >> a;
  cin >> b;
  cin >> c;
  cin >> d;

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

  cout << max;
  return 0;
}
```

11. Solution

```
void f1(double a, double b, double c, double &sum, double &average) {
  sum = a + b + c;
  average = sum / 3;
}
```

12. 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;
```

```
}
```

13. Solution

```
#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;

string get_input()  {
  string answer;

  do {
    cout << "Enter Yes or No: ";
    cin >> answer;
    answer = to_upper_copy(answer);
  } while (answer != "YES" && answer != "NO");
  return answer;
}

double find_area(double base, double height) {
  return base * height;
}

int main() {
  double base, height;

  do {
    cout << "Enter the base of the parallelogram: ";
    cin >> base;
    cout << "Enter the height of the parallelogram: ";
    cin >> height;

    cout << "Area = " << find_area(base, height) << endl;

    cout << "Would you like to repeat? " << endl;
  } while (get_input() == "YES") ;
  return 0;
}
```

14. 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, sum = 0;
  for (i = 0; i <= STUDENTS - 1; i++) {
    sum += grades[i];
  }
  return sum / (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;
}
```

15. Solution

```
#include <iostream>
```

```
using namespace std;
const int JUDGES = 10;

int *get_array()  {
  static 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 &min, int &max) {
  int i;

  min = score[0];
  max = score[0];
  for (i = 1; i <= JUDGES - 1; i++) {
    if (score[i] > max) {
      max = score[i];
    }
    if (score[i] < min) {
      min = score[i];
    }
  }
}

int main() {
  string name;
  int i, sum, points, min = 0, max = 0;

  cout << "Enter artist's name: ";
  cin >> name;
  int * score = get_array();
  find_min_max(score, min, max);

  sum = 0;
  for (i = 0; i <= JUDGES - 1; i++) {
    sum += score[i];
  }

  points = sum - min - max;
  cout << "Artist " << name << " got " << points << " points";
  return 0;
}
```

16. 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 sum;
  int i;

  sum = 0;
  for (i = 1; i <= 64; i++) {
    sum += woc(i);
  }
  cout << sum;
  return 0;
}
```

17. Solution

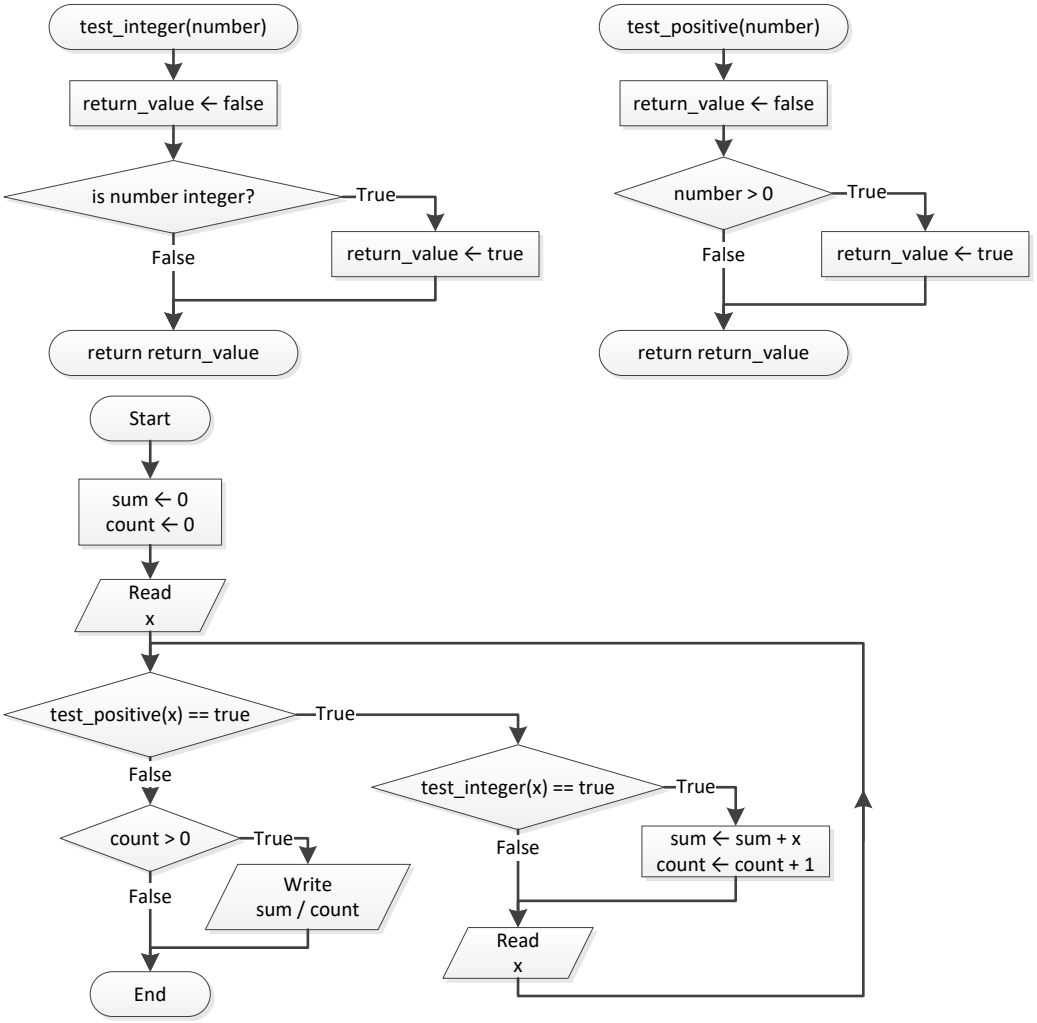
```
#include <iostream>
#include <cmath>
using namespace std;
double fact(int value) {
  double return_value;

  if (value == 1) {
    return_value = 1;
  }
  else {
    return_value = value * fact(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) / fact(i) - pow(x, i - 2) / fact(i - 2);
  }

  return return_value;
}

int main() {
```

```
  cout << my_cos(M_PI / 4);
  return 0;
}
```

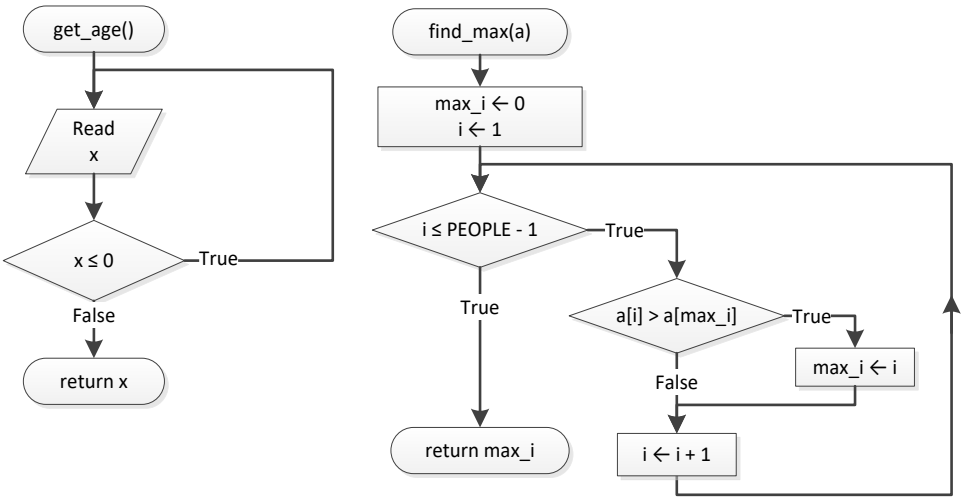
Chapter 42

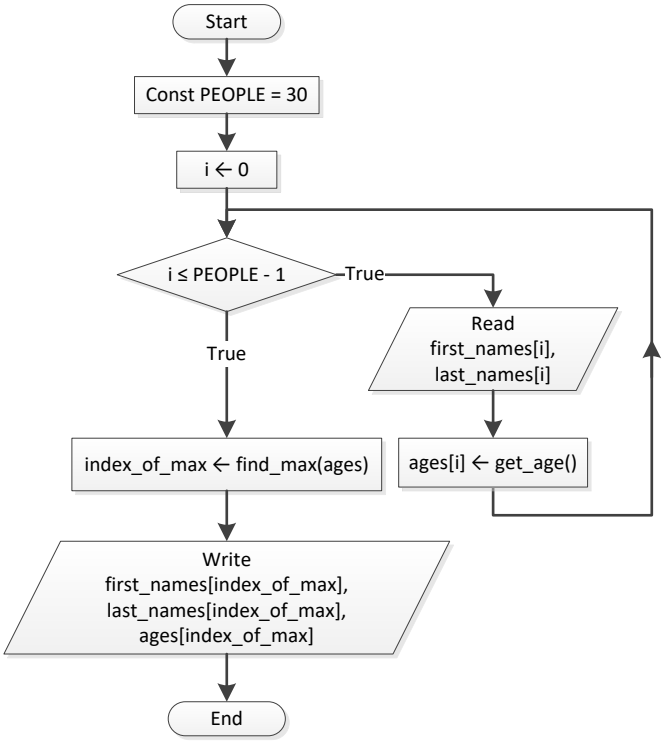
42.4 Review Exercises

1. Solution

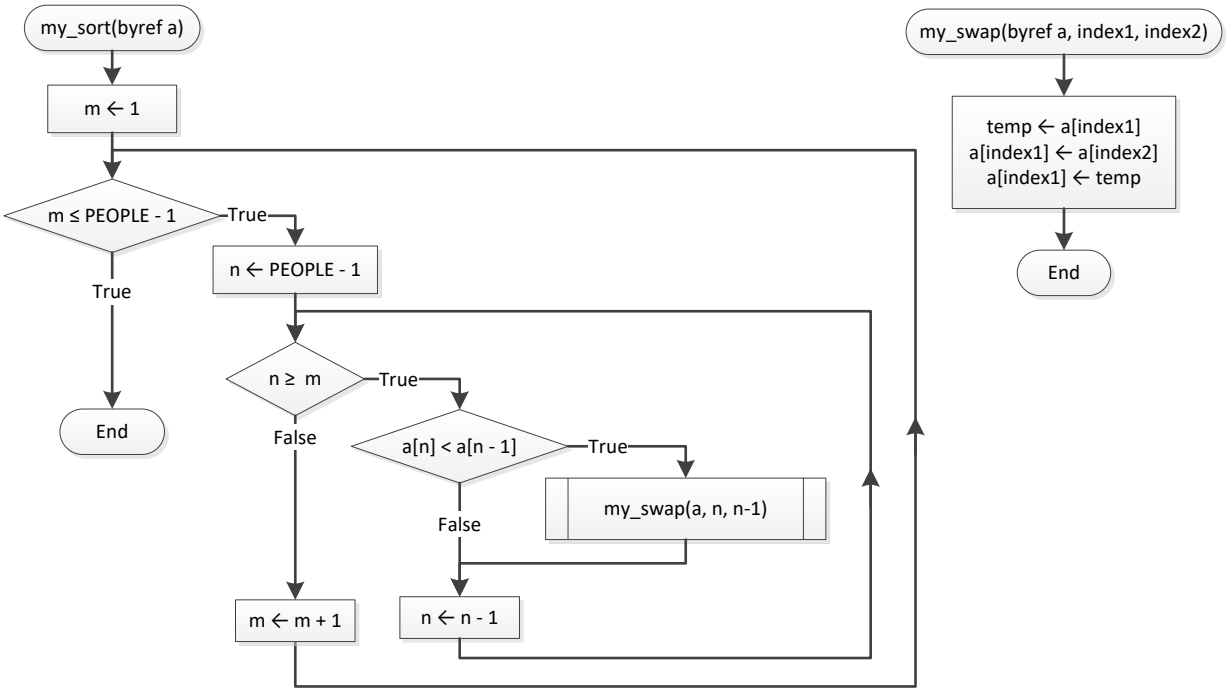


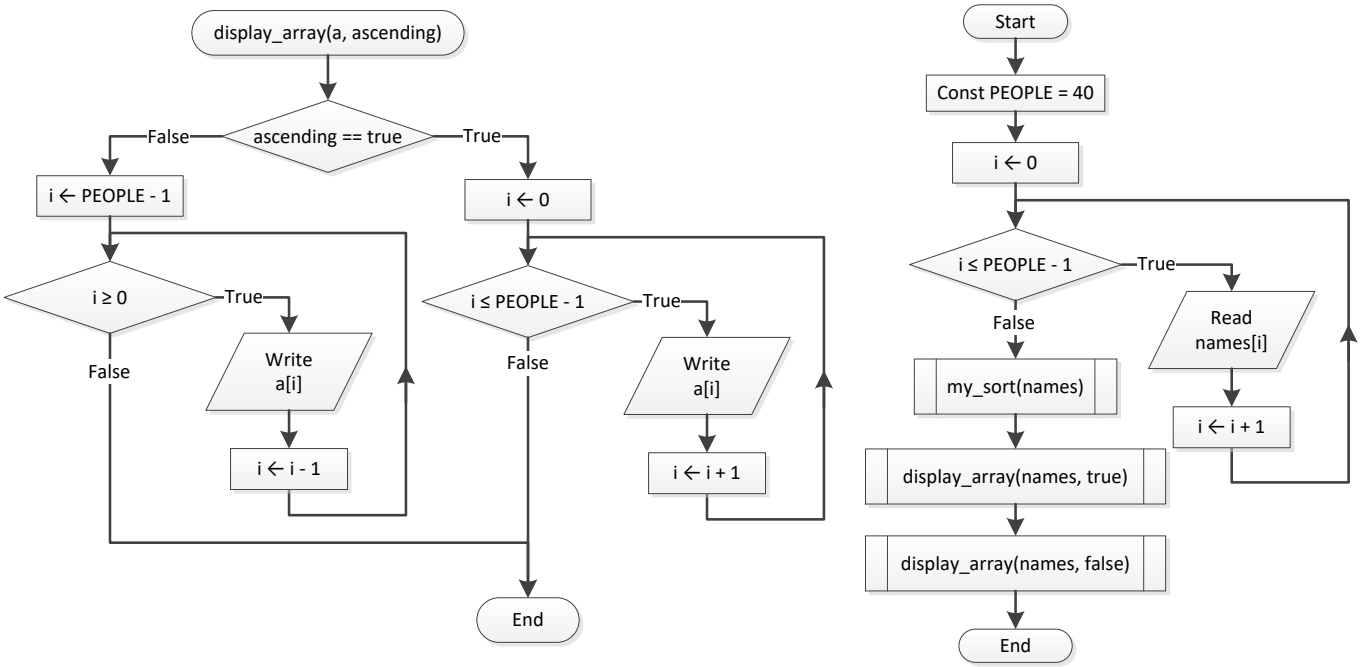
2. Solution


get_age()
Read
x
x ≤ 0
True
False
return x
find_max(a)
max_i ← 0
i ← 1
i ≤ PEOPLE - 1
True
True
a[i] > a[max_i]
True
max_i ← i
False
i ← i + 1
return max_i

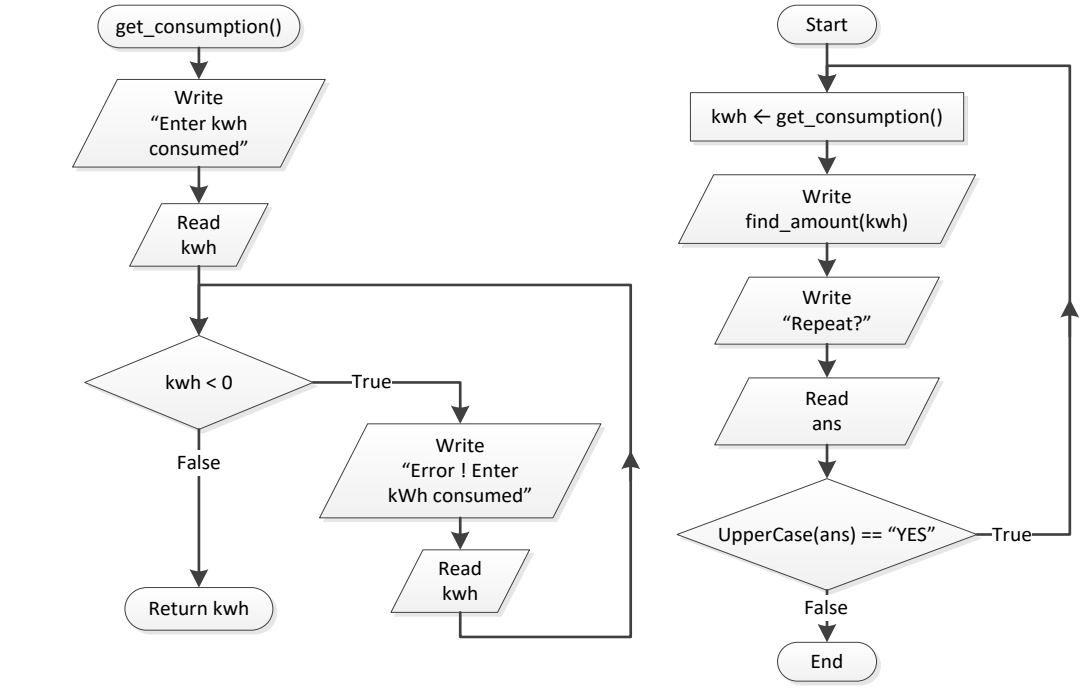

Start
Const PEOPLE = 30
i ← 0
i ≤ PEOPLE - 1
True
True
Read
first_names[i],
last_names[i]
index_of_max ← find_max(ages)
ages[i] ← get_age()
Write
first_names[index_of_max],
last_names[index_of_max],
ages[index_of_max]
End

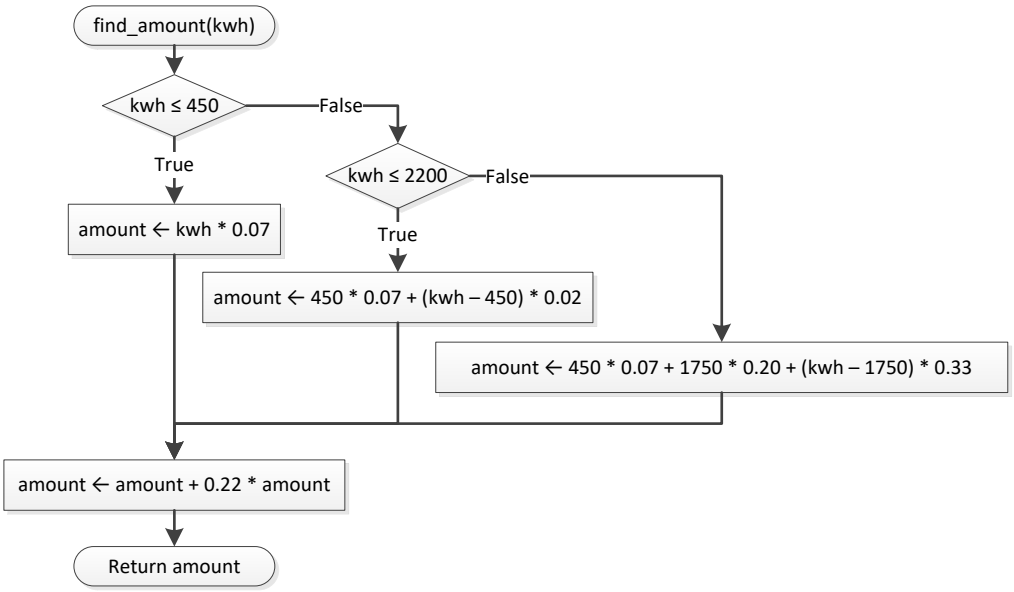
3. Solution


my_sort(byref a)
m ← 1
m ≤ PEOPLE - 1
True
n ← PEOPLE - 1
True
End
n ≥ m
True
False
a[n] < a[n - 1]
True
my_swap(a, n, n-1)
False
m ← m + 1
n ← n - 1
my_swap(byref a, index1, index2)
temp ← a[index1]
a[index1] ← a[index2]
a[index1] ← temp
End


display_array(a, ascending)
ascending == true
False
True
i ← PEOPLE - 1
i ≥ 0
True
Write a[i]
i ← i - 1
False
i ← 0
i ≤ PEOPLE - 1
True
Write a[i]
i ← i + 1
False
End
Start
Const PEOPLE = 40
i ← 0
i ≤ PEOPLE - 1
True
Read names[i]
i ← i + 1
False
my_sort(names)
display_array(names, true)
display_array(names, false)
End

4. Solution





5. Solution

```
#include <iostream>
using namespace std;
const int STUDENTS = 20;
const int LESSONS = 10;

//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_arrays(string names[], int grades[][LESSONS])  {
```

```
  int i, j;

  for (i = 0; i <= STUDENTS - 1; i++) {
    cin >> names[i];
    for (j = 0; j <= LESSONS - 1; j++) {
      cin >> grades[i][j];
    }
  }
}

double * find_average(int grades[][LESSONS]) {
  int i, j;
  static 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;
  }
  return average;
}

void display(string names[], double average[]) {
  int i;

  for (i = 0; i <= STUDENTS - 1; i++) {
    if (average[i] > 89) {
      cout << names[i] << ", " << average[i] << endl;
    }
  }
}

int main() {
  string names[STUDENTS];
  int grades[STUDENTS][LESSONS];

  get_arrays(names, grades);
  double * av = find_average(grades);
  display(names, av);
  return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
double fib(int n) {
  double return_val;

  if (n == 0 || n == 1) {
    return_val = n;
  }
```

```
  else {
    return_val = fib(n - 1) + fib(n - 2);
  }

  return return_val;
}

int main() {
  int n;
  string ans;

  do {
    cin >> n;
    while (n < 0) {
      cout << "Error" << endl;
      cin >> n;
    }

    cout << fib(n) << endl;
    cout << "Again? " << endl;
    cin >> ans;
  } while (ans == "Y");
  return 0;
}
```

Chapter 43

43.3 Review Exercises

1. Solution

```
#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;
}
```

2. 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) == true) {
        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, sum, i;
```

```
  cout << "Enter day: ";
  cin >> day;
  cout << "Enter month: ";
  cin >> month;
  cout << "Enter year: ";
  cin >> year;
  while (check_date(day, month, year) == false) {
    cout << "Error!" << endl;
    cout << "Enter day: ";
    cin >> day;
    cout << "Enter month: ";
    cin >> month;
    cout << "Enter year: ";
    cin >> year;
  }

  sum = 0;
  for (i = 1; i <= month - 1; i++) {
    sum += num_of_days(year, i);
  }
  sum += day;

  cout << sum;
  return 0;
}
```

3. Solution

```
#include <iostream>
#include <ctime>
#include <cstdlib>
using namespace std;
int dice() {
  return 1 + rand() % 6;
}

int main() {
  int dice1, dice2, i, player, sum, sum_player1 = 0, sum_player2 = 0;
  string key, name1, name2;

  srand(time(NULL));

  cout << "Player1 - Enter name: ";
  cin >> name1;
  cout << "Player2 - Enter name: ";
  cin >> name2;

  for (player = 1; player <= 2; player++) {
    sum = 0;
    for (i = 1; i <= 10; i++) {
      if (player == 1) {
        cout << name1 << ", hit enter to roll the dice!" << endl;
      }
      else {
```

```
        cout << name2 << ", hit enter to roll the dice!" << endl;
      }
      getline(cin, key);

      dice1 = dice();
      dice2 = dice();
      cout << dice1 << "   " << dice2 << endl;
      sum += dice1 + dice2;
    }
    if (player == 1) {
      sum_player1 = sum;
    }
    else {
      sum_player2 = sum;
    }
  }

  if (sum_player1 == sum_player2) {
    cout << "Tie!" << endl;
  }
  else if (sum_player1 > sum_player2) {
    cout << name1 << " wins" << endl;
  }
  else {
    cout << name2 << " wins" << endl;
  }
  return 0;
}
```

4. 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 type, int rental_days) {
  double charge;

  if (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 (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, sum;

  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();
  }
```

```
  sum = 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;
     sum += 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: " << sum / (1 + TAX_RATE);
  return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
const int CHANNELS = 10;
const int DAYS = 7;
const string day_names[7] = {"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 sum ,i;

  sum = 0;
  for (i = 0; i <= 4; i++) {
    sum += a[i];
  }
  return sum / 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 == true) {
      cout << names[i] << endl;
    }
  }
  return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
const int CITIZENS = 30;

void input_data(long int 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 int SSNs[], string answers[]) {
  int m, n, index_of_min;
```

```
  long int min, temp;
  string temp_str;

  for (m = 0; m <= CITIZENS - 1; m++) {
    min = SSNs[m];
    index_of_min = m;
    for (n = m; n <= CITIZENS - 1; n++) {
      if (SSNs[n] < min) {
        min = 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 int SSNs[], long int SSN) {
  int left, right, middle, index_position = 0, return_value;
  bool found;

  left = 0;
  right = CITIZENS - 1;
  found = false;
  while (left <= right && found == false) {
    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 == false) {
    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 int SSNs[CITIZENS];
  long int 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;
}
```

7. 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 == false) {
      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, sum;
```

```
  input_data(names, results);
  display_result(names, results);

  index = find_team(names);
  while (index != -1) {
    sum = 0;
    for (j = 0; j <= GAMES - 1; j++) {
      if (results[index][j] == "W") {
        sum += 3;
      }
      else if (results[index][j] == "T") {
        sum += 1;
      }
    }
    cout << "Points: " << sum << endl;
    index = find_team(names);
  }
  return 0;
}
```

8. Solution

```
#include <iostream>
using namespace std;
//space is a valid character!
string alphabet = " abcdefghijklmnopqrstuvwxyz";

string my_encrypt(string message, int encryption_key) {
  string return_value;
  int i, index, new_index;
  string letter, new_letter;

  return_value = "";
  for (i = 0; i <= message.length() - 1; i++) {
    letter = message.at(i);
    index = alphabet.find(letter);
    new_index = (index + encryption_key) % 27; //26 letters + 1 space
    new_letter = alphabet.at(new_index);
    return_value += new_letter;
  }
  return return_value;
}

string my_decrypt(string message, int decryption_key) {
  string return_value;
  int i, index, new_index;
  string letter, new_letter;

  return_value = "";
  for (i = 0; i <= message.length() - 1; i++) {
    letter = message.at(i);
    index = alphabet.find(letter);
    new_index = (index + 27 - decryption_key) % 27; //26 letters + 1 space
```

```
    new_letter = alphabet.at(new_index);
    return_value += new_letter;
  }
  return return_value;
}

void display_menu() {
  cout << endl;
  cout << "1. Encrypt a message" << endl;
  cout << "2. Decrypt a message" << endl;
  cout << "3. Exit" << endl;
}

int main() {
  int choice, encryption_key, decryption_key;
  string message;

  do {
    display_menu();
    cout << "Enter a choice: ";
    cin >> choice;

    //When there is a cin statement before a getline() statement
    //you need to write the following statement between them
    //otherwise the next getline() statement won't work
    cin.ignore(100, '\n');

    if (choice == 1) {
      cout << "Enter a message to encrypt: ";
      getline(cin, message);
      cout << "Enter an encryption key: ";
      cin >> encryption_key;
      cout << "Your encrypted message is: " << my_encrypt(message, encryption_key) << endl;
    }
    else if (choice == 2) {
      cout << "Enter a message to decrypt: ";
      getline(cin, message);
      cout << "Enter an decryption key: ";
      cin >> decryption_key;
      cout << "Your decrypted message is: " << my_decrypt(message, decryption_key) << endl;
    }
  } while (choice != 3);
  return 0;
}
```

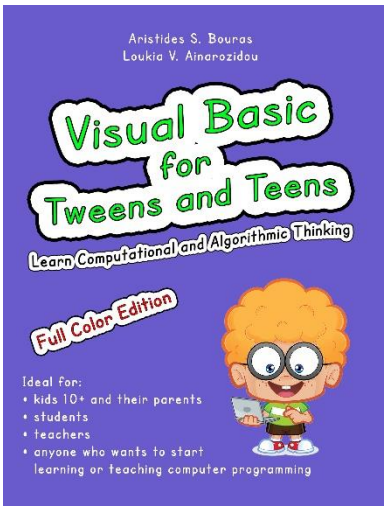
Some Final Words from the Authors

We hope you really enjoyed reading this book. We made every possible effort to make it comprehensible even by people that probably have no previous experience in programming.

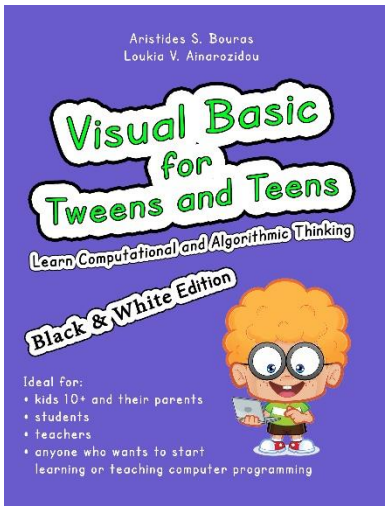
So if you liked this book, please visit the web store where you bought it and show us your gratitude by writing a good review and giving us as many stars as possible. By doing this, you will encourage us to continue writing and of course you'll help other readers to reach us.

And remember: Learning is a process within an endless loop structure. It begins at birth and continues throughout your lifetime!

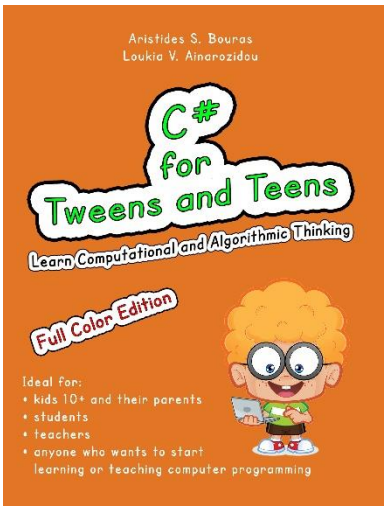
More of our Books



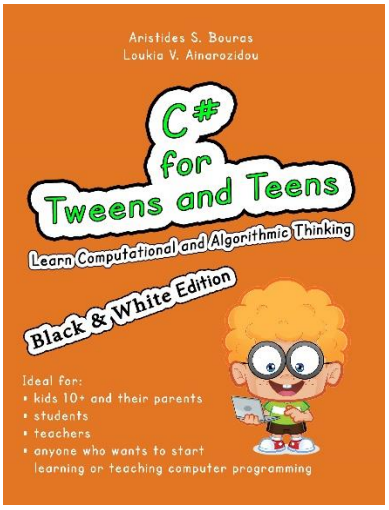
ISBN-10: 1982083670
ISBN-13: 978-1982083670



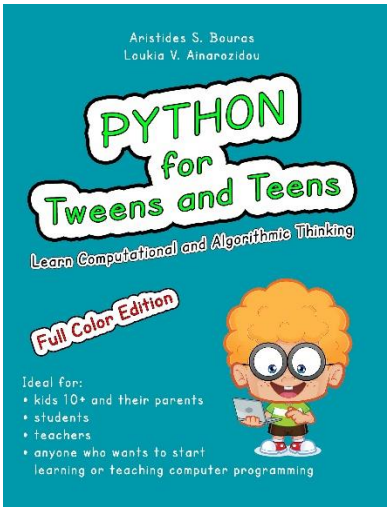
ISBN-10: 1982083697
ISBN-13: 978-1982083694



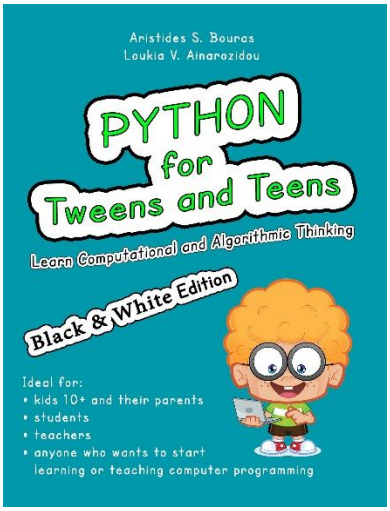
ISBN-10: 1973727684
ISBN-13: 978-1973727682



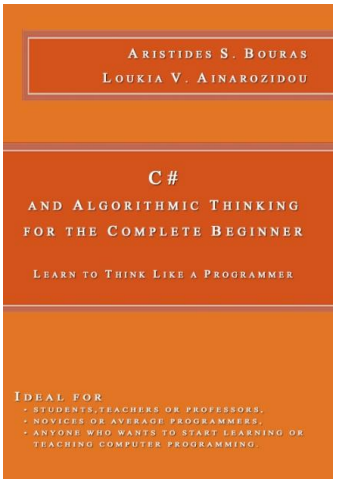
ISBN-10: 1973727765
ISBN-13: 978-1973727767



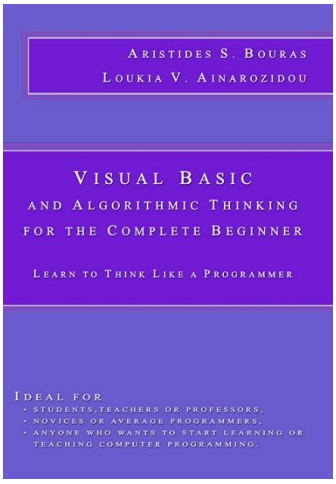
ISBN-10: 1543127940
ISBN-13: 978-1543127942



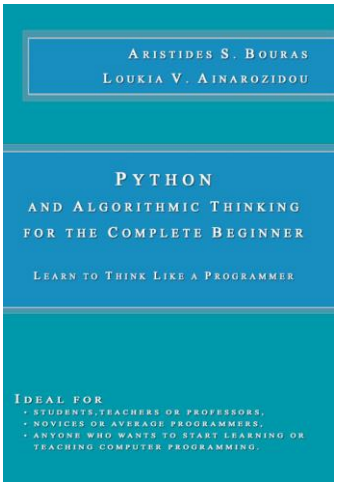
ISBN-10: 1546611215
ISBN-13: 978-1546611219



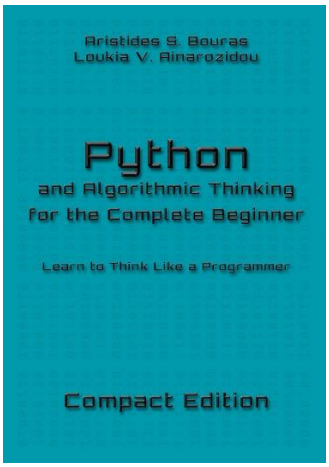
ISBN-10: 1508952485
ISBN-13: 978-1508952480



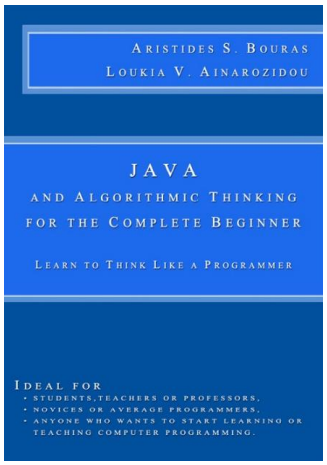
ISBN-10: 1511798963
ISBN-13: 978-1511798969



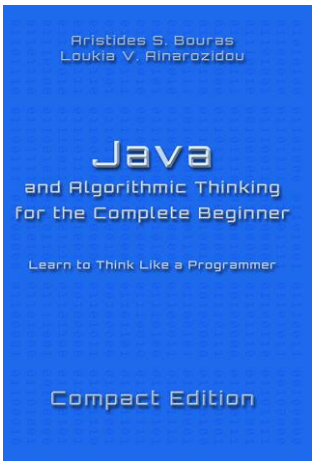
ISBN-10: 1514802163
ISBN-13: 978-1514802168



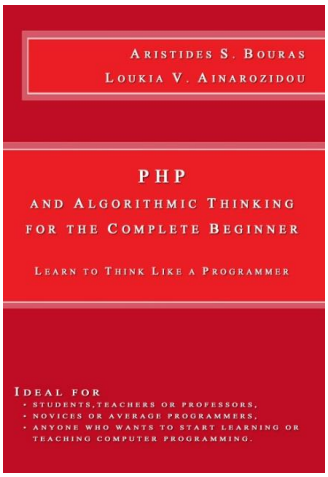
ISBN-10: 1983276111
ISBN-13: 978-1983276118



ISBN-10: 1506179398
ISBN-13: 978-1506179391



ISBN-10: 1723731587
ISBN-13: 978-1723731587



ISBN-10: 1503015912
ISBN-13: 978-1503015913