Python
AND ALGORITHMIC THINKING
FOR THE COMPLETE BEGINNER

The Answers

Aristides S. Bouras
Loukia V. Ainarozidou

RCode: 180725

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 "PYTHON AND ALGORITHMIC THINKING FOR THE COMPLETE BEGINNER". Every effort has been taken to make this book compatible with all releases of Python 3.x, and it is almost certain to be compatible with any future releases of Python.

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

5.8 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

5.9 Answers of Review Questions: Multiple Choice

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

5.10 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

7.6 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

7.7 Answers of Review Questions: Multiple Choice

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

7.8 Answers of Review Exercises

1. ii, iv, v, ix, x
2. i. String, ii. Boolean, iii. String, iv. String, v. Real, vi. Integer
3. i. d, ii. f, iii. c, iv. e
4. i. 27, 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. Division 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	`a = int(input())`	**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	`print(a, ",", b, ",", c, ",", d)`	40, 1, 1, 40 is displayed			

For the input value of 4

Step	Statement	a	b	c	d
1	`a = int(input())`	**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	`print(a, ",", b, ",", c, ",", d)`	49, 10, 3, 1470 is displayed			

For the input value of 1

Step	Statement	a	b	c	d
1	`a = int(input())`	**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	`print(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	`a = float(input())`	**3**	?	?	?	?
2	`b = float(input())`	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 -= 1`	3	4	30.5	8.25	**14.25**
8	`d -= c + d % c`	3	4	30.5	**-30.5**	14.25
9	`print(c, ",", d, ",", e)`	30.5, -30.5, 14.25 is displayed				

For the input values of 4, 4

Step	Statement	a	b	c	d	e
1	`a = float(input())`	**4**	?	?	?	?
2	`b = float(input())`	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 -= 1`	4	4	38	11	**18**
8	`d -= c + d % c`	4	4	38	**-38**	18
9	`print(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	`a = float(input())`	**5**	?	?	?	?
2	`b = float(input())`	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 -= 1`	5	5	-17	17	**-8**
8	`d -= c + a % c`	5	5	-17	**29**	-8
9	`print(c, ",", d, ",", e)`	-17, 29, -8 is displayed				

For the input values of 4, 8

Step	Statement	a	b	c	d	e
1	`a = float(input())`	**4**	?	?	?	?
2	`b = float(input())`	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 -= 1`	4	8	-13	13	**-2**
8	`d -= c + a % c`	4	8	-13	**22**	-2
9	`print(c, ",", d, ",", e)`	-13, 22, -2 is displayed				

3. Solution

For the input value of 0.50

Step	Statement	a	b	c
1	`b = float(input())`	?	**0.50**	?
2	`c = 5`	?	0.50	**5**

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

For the input value of 3

Step	Statement	a	b	c
1	`b = float(input())`	?	3	?
2	`c = 5`	?	3	**5**
3	`c = c * b`	?	3	**15**
4	`a = 10 * c % 10`	**0**	3	15
5	`print(a)`	Value 0 is displayed		

For the input value of 15

Step	Statement	a	b	c
1	`b = float(input())`	?	15	?
2	`c = 5`	?	15	**5**
3	`c = c * b`	?	15	**75**
4	`a = 10 * c % 10`	**0**	15	75
5	`print(a)`	Value 0 is displayed		

Chapter 10

10.2 Answers of Review Exercises

1. Solution

```
base = float(input("Enter base: "))
height = float(input("Enter height: "))

area = 0.5 * base * height

print(area)
```

2. Solution

```
angle1 = float(input("Enter 1st angle: "))
angle2 = float(input("Enter 2nd angle: "))

angle3 = 180 - angle1 - angle2

print(angle3)
```

3. Solution

```
g1 = int(input("Enter 1st grade: "))
g2 = int(input("Enter 2nd grade: "))
g3 = int(input("Enter 3rd grade: "))
g4 = int(input("Enter 4th grade: "))

average = (g1 + g2 + g3 + g4) / 4

print(average)
```

4. Solution

```
PI = 3.14159

r = float(input("Enter radius: "))

perimeter = 2 * PI * r

print(perimeter)
```

5. Solution

```
charge = float(input("Enter charge for a meal: "))

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

total = charge + tip + tax
```

```
print(total)
```

6. Solution

```
a = float(input("Enter acceleration in m/sec2: "))
t = float(input("Enter time traveled in sec: "))

s = 0.5 * a * t * t

print(s)
```

7. Solution

```
f = float(input("Enter temperature in Fahrenheit: "))

c = 5 / 9 * (f - 32)

print(c)
```

8. Solution

```
w = int(input("Enter weight in pounds: "))
h = int(input("Enter height in inches: "))

bmi = w * 703 / (h * h)

print(bmi)
```

9. Solution

```
s_total = float(input("Enter subtotal: "))
g_rate = float(input("Enter gratuity rate: "))

tip = s_total * g_rate / 100

total = s_total + tip

print("Tip is", tip, "and Total is", total)
```

10. Solution

```
VAT = 0.20

btax_price1 = float(input("Enter before-tax price 1: "))
btax_price2 = float(input("Enter before-tax price 2: "))
btax_price3 = float(input("Enter before-tax price 3: "))

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
```

```
print(avg)
```

11. Solution

```
VAT = 0.20

atax_price = float(input("Enter after-tax price: "))

btax_price = atax_price / (1 + VAT)

print(btax_price)
```

12. Solution

```
i_price = float(input("Enter price: "))
discount = float(input("Enter discount: "))

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

print(f_price, saved)
```

13. Solution

```
VAT = 0.20

i_kWh = int(input("Enter kWh at the beginning of the month: "))
f_kWh = int(input("Enter kWh at the end of the month: "))

kWh_consumed = f_kWh - i_kWh

cost = kWh_consumed * 0.06
cost += cost * VAT

print(kWh_consumed, cost)
```

14. Solution

```
day = int(input("Enter day: "))
month = int(input("Enter month: "))

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

print(days_left)
```

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	`a = float(input())`	**9**	?	?
2	`a += 6 / math.sqrt(a) * 2 + 20`	**33**	?	**?**
3	`b = round(a) % 4`	33	**1**	**?**
4	`c = b % 3`	33	1	**1**
5	`print(a, ",", b, ",", c)`	33, 1, 1 is displayed		

For the input value of 4

Step	Statement	a	b	c
1	`a = float(input())`	**4**	?	?
2	`a += 6 / math.sqrt(a) * 2 + 20`	**30**	?	**?**
3	`b = round(a) % 4`	30	**2**	**?**
4	`c = b % 3`	30	2	**2**
5	`print(a, ",", b, ",", c)`	30, 2, 2 is displayed		

2. Solution

For the input value of -2

Step	Statement	a	b	c
1	`a = int(input())`	**-2**	?	?
2	`b = abs(a) % 4 + a ** 4`	-2	**18**	**?**
3	`c = b % 5`	-2	18	**3**

4	`print(b, ",", c)`	18, 3 is displayed

For the input value of -3

Step	Statement	a	b	c
1	`a = int(input())`	**-3**	?	?
2	`b = abs(a) % 4 + a ** 4`	-3	**84**	**?**
3	`c = b % 5`	-3	84	**4**
4	`print(b, ",", c)`	84, 4 is displayed		

3. Solution

```
import math

radians = float(input("Enter angle in radians: "))

degrees = radians * 180 / math.pi

print(degrees)
```

4. Solution

```
import math

a = float(input("Enter side A of a right-angled triangle: "))
b = float(input("Enter side B of a right-angled triangle: "))

hypotenuse = math.sqrt(a ** 2 + b ** 2)

print(hypotenuse)
```

5. Solution

```
import math

th = float(input("Enter angle (in degrees) of a right-angled triangle: "))
adjacent = float(input("Enter length of adjacent side: "))

opposite = math.tan(th * math.pi / 180) * adjacent

print(opposite)
```

Chapter 12

12.2 Answers of Review Exercises

1. Solution

i. a, e, g, h

ii. c, f

2. Solution

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

ii. `y = (3 * x ** 2 - x ** 3 / 4) ** (1 / 5)`

iii. `y = math.sqrt(x ** 4 - 2 * x ** 3 - 7 * x ** 2 + x) / (4 * (7 * x ** 4 - 3 / 4 * x ** 3) * (7 * x ** 2 + x)) ** (1 / 3)`

iv. `y = x / (x - 3 * (x - 1)) + x * (x - 1) ** (1 / 5) / ((x ** 3 - 2) * (x - 1) ** 3)`

v. `y = (math.sin(math.pi / 3) - math.cos(math.pi / 2 * w)) ** 2`

vi. `y = (math.sin(math.pi / 2 * x) + math.cos(3 * math.pi / 2 * w)) ** 3 / (math.tan(2 * math.pi / 3 * w) - math.sin(math.pi / 2 * x)) ** 0.5`

3. Solution

```
import math

x = float(input("Enter value for x: "))

y = math.sqrt(x) * (x ** 3 + x ** 2)

print(y)
```

4. Solution

```
x = float(input("Enter value for x: "))

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

print(y)
```

5. Solution

```
import math

x = float(input("Enter value for x: "))
w = float(input("Enter value for w: "))

y = x ** (x + 1) / (math.tan(2 * w / 3 + 5) - math.tan(x / 2 + 1)) ** 3

print(y)
```

6. Solution

```
x = float(input("Enter value for x: "))
w = float(input("Enter value for w: "))

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

print(y)
```

7. Solution

```
import math

x = float(input("Enter value for x: "))
w = float(input("Enter value for w: "))

y = x ** x / (math.sin(2 * w / 3 + 5) - x) ** 2 + (math.sin(3 * x) + w) ** (x + 1) / math.sqrt(7 * w) ** (3 / 2)

print(y)
```

8. Solution

```
import math

a = float(input("Enter length A: "))
b = float(input("Enter length B: "))
c = float(input("Enter length C: "))

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

print(area)
```

Chapter 13

13.2 Answers of Review Exercises

1. Solution

```
n = int(input("Enter an integer: "))

last_digit = n % 10
result = last_digit * 8

print(result)
```

2. Solution

```
number = int(input("Enter a five-digit integer: "))


digit5 = number % 10
r = number // 10

digit4 = r % 10
r = r // 10

digit3 = r % 10
r = r // 10

digit2 = r % 10
digit1 = r // 10

reversed_number = digit5 * 10000 + digit4 * 1000 + digit3 * 100 + digit2 * 10 + digit1
print(reversed_number)
```

However, using the `divmod()` function it can become:

```
number = int(input("Enter a five-digit integer: "))

r, digit5 = divmod(number, 10)
r, digit4 = divmod(r, 10)
r, digit3 = divmod(r, 10)
digit1, digit2 = divmod(r, 10)

reversed_number = digit5 * 10000 + digit4 * 1000 + digit3 * 100 + digit2 * 10 + digit1
print(reversed_number)
```

3. Solution

```
n = int(input("Enter an integer: "))

result = n % 2

print(result)
```

4. Solution

```
n = int(input("Enter an integer: "))

result = 1 - n % 2

print(result)
```

5. Solution

```
number = int(input("Enter a period of time in seconds: "))

weeks = number // 604800       # 60 * 60 * 24 * 7 = 604800
r = number % 604800

days = r // 86400              # 60 * 60 * 24 = 86400
r = r % 86400

hours = r // 3600
r = r % 3600

minutes = r // 60
seconds = r % 60

print(weeks, "weeks", days, "days", hours, "hours", minutes, "minutes and", seconds, "seconds")
```

However, using the `divmod()` function it can become:

```
number = int(input("Enter a period of time in seconds: "))

weeks, r = divmod(number, 604800)      # 60 * 60 * 24 * 7 = 604800
days, r = divmod(r, 86400)             # 60 * 60 * 24 = 86400
hours, r = divmod(r, 3600)
minutes, seconds = divmod(r, 60)

print(weeks, "weeks", days, "days", hours, "hours", minutes, "minutes and", seconds, "seconds")
```

6. Solution

```
amount = int(input("Enter amount to withdraw: "))

usd20, r = divmod(amount, 20)
usd10, r = divmod(r, 10)
usd5, usd1 = divmod(r, 5)

print(usd20, "notes of $20", usd10, "notes of $10", usd5, "notes of $5 and", usd1, "notes of $1")
```

7. Solution

```
steps = int(input("Enter number of steps: "))

distance = steps * 25

miles, r = divmod(distance, 63360)
```

```
yards, r = divmod(r, 36)
feet, inches = divmod(r, 12)

print(miles, "miles", yards, "yards", feet, "feet and", inches, "inches")
```

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

```
first_name = input("First name: ")
middle_name = input("Middle name: ")
last_name = input("Last name: ")
title = input("Title: ")

print(title, first_name, middle_name, last_name)
print(first_name, middle_name, last_name)
print(last_name, ",", first_name)
print(last_name, ",", first_name, middle_name)
print(last_name, ",", first_name, middle_name, ",", title)
print(first_name, last_name)
```

2. Solution

```
import random
import string

alphabet = string.ascii_lowercase

random_word = alphabet[random.randrange(26)].upper() +   \
              alphabet[random.randrange(26)] +           \
              alphabet[random.randrange(26)] +           \
              alphabet[random.randrange(26)] +           \
              alphabet[random.randrange(26)]
```

```
print(random_word)
```

3. Solution

```
import random
name = input("Enter name: ").lower()

secret_password = name[random.randrange(len(name))] +    \
                  name[random.randrange(len(name))] +    \
                  name[random.randrange(len(name))] +    \
                  str(random.randrange(1000, 10000))

print(secret_password)
```

Chapter 15

15.9 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

15.10 Answers of Review Questions: Multiple Choice

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

15.11 Answers of Review Exercises

1. Solution

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

2. Solution

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

3. Solution

Boolean Expression1 (BE1)	Boolean Expression2 (BE2)	BE1 or BE2	BE1 and BE2	not(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 or c > b and c > 1	a > 3 and c > b or c > 1
4	**-6**	**2**	True	True
-3	**2**	**-4**	False	False
2	**5**	**5**	False	True

5. Solution

Expression	Value
`(x + y) ** 3`	8
`(x + y) / (x ** 2 - 14)`	1
`x - 1 == y + 5`	True
`x > 2 and y == 1`	False
`x == 1 or y == -2 and not(flag == False)`	True
`not(x >= 3) and (x % 2 > 1)`	False

6. Solution

i. False
ii. True

7. Solution

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

8. Solution

i. `x != 4 or y == 3`
ii. `x + 4 > 0`
iii. `not(x <= 5) and y != 4`
iv. `x == False`
v. `not(x < 4 and z <= 4)`
vi. `x == 2 or x < -5`

9. Solution

i. `not(x < 4 or y == 10)`

ii. `not( x - 2 < 9)`

iii. `not(not(x < 2) and y == 4)`

iv. `not(x == False and y != 3)`

v. First approach: `not(not(x < 2 or y < 2))`
Second approach: `x < 2 or y < 2`

vi. `not( x == -2 or 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

```
x = float(input())

y = - 5
if x * y / 2 > 20:
  y -= 1
  x += 4 * x ** 2

print(x, y)
```

2. Solution

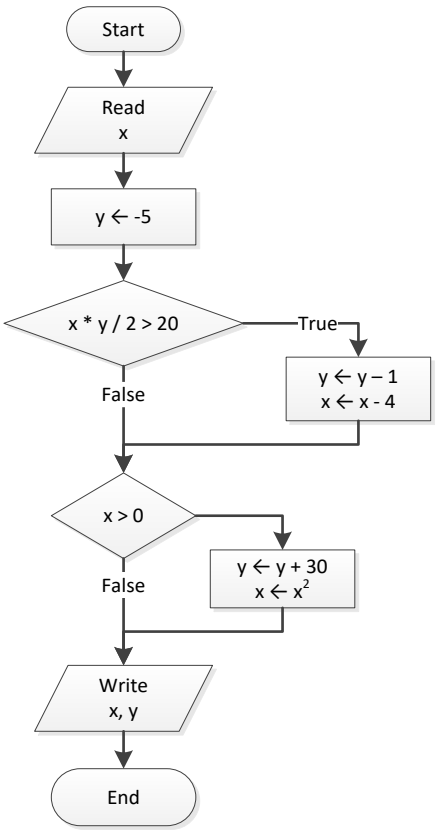
For the input value of 10

Step	Statement	x	y
1	`x = float(input())`	**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 = x ** 2`	**100**	25
7	`print(x, ",", y)`	100, 25 is displayed	

For the input value of -10

Step	Statement	x	y
1	`x = float(input())`	**-10**	?
2	`y = - 5`	-10	**-5**
3	`if x * y / 2 > 20:`	True	

4	`y -= 1`	-10	**-6**
5	`x -= 4`	**-14**	-6
6	`if x > 0:`	False	
7	`print(x, ",", y)`	-14, -6 is displayed	



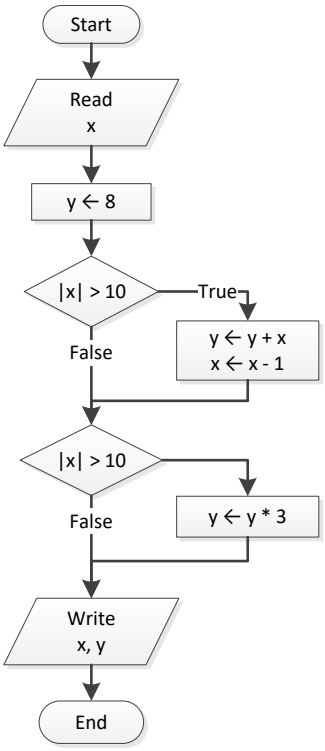
3. *Solution*

For the input value of -11

Step	Statement	x	y
1	`x = int(input())`	**-11**	?
2	`y = 8`	-11	**8**
3	`if abs(x) > 10:`	True	
4	`y += x`	-11	**-3**
5	`x -= 1`	**-12**	-3
6	`if abs(x) > 10:`	True	
7	`y *= 3`	-12	**-9**
8	`print(x, ",", y)`	-12, -9 is displayed	

For the input value of 11

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



4. Solution

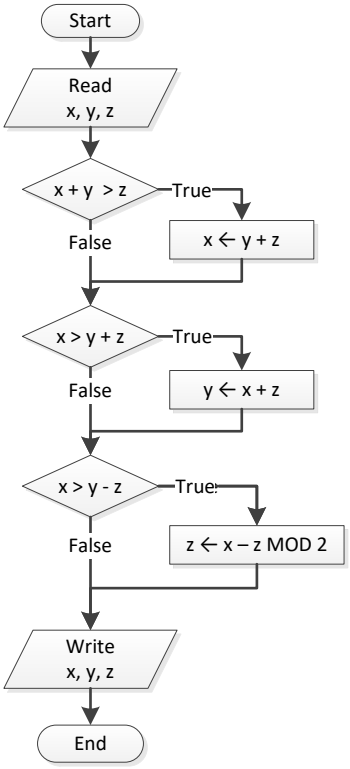
For input values of 1, 2 and 3

Step	Statement	x	y	z
1	`x = int(input())`	**1**	?	?
2	`y = int(input())`	1	**2**	?
3	`z = int(input())`	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	`print(x, ",", y, ",", z)`	1, 2, 0 is displayed

For input values of 4, 2 and 1

Step	Statement	x	y	z
1	`x = int(input())`	**4**	?	?
2	`y = int(input())`	4	**2**	?
3	`z = int(input())`	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	`print(x, ",", y, ",", z)`	3, 2, 2 is displayed		



5. Solution

```
x = float(input("Enter a number: "))
if x > 0:
    print("Positive")
```

6. Solution

```
x = float(input("Enter a number: "))
y = float(input("Enter a second number"))
```

```
if x > 0 and y > 0:
    print("Positive")
```

7. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

x = input("Enter a number: ")

if re.match(IS_NUMERIC, x):
    print("Numeric")
```

8. Solution

```
s = input("Enter a string: ")

if s == s.upper():
    print("Uppercase")
```

9. Solution

```
s = input("Enter a string: ")

if len(s) > 20:
    print("Many characters")
```

10. Solution

```
n1 = float(input("Enter 1st number: "))
n2 = float(input("Enter 2nd number: "))
n3 = float(input("Enter 3rd number: "))
n4 = float(input("Enter 4th number: "))

if n1 < 0 or n2 < 0 or n3 < 0 or n4 < 0:
    print("Among the given numbers, there is a negative one!")
```

11. Solution

```
a = float(input("Enter 1st number: "))
b = float(input("Enter 2nd number: "))

if a > b:
    c = a           # Or you can do the following:
    a = b           # a, b = b, a
    b = c           #

print(a, b)
```

12. Solution

```
t1 = float(input("Enter 1st temperature: "))
t2 = float(input("Enter 2nd temperature: "))
t3 = float(input("Enter 3rd temperature: "))

average = (t1 + t2 + t3) / 3

if average > 60:
    print("Heat Wave")
```

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

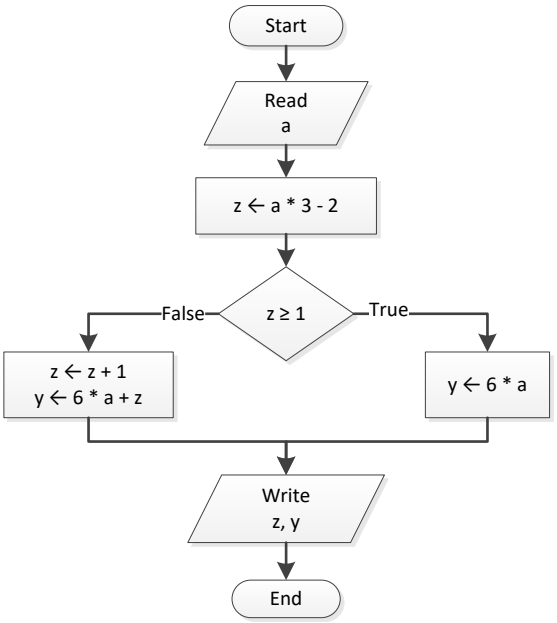
1. Solution

For input value of 3

Step	Statement	a	y	z
1	`a = float(input())`	**3**	?	?
2	`z = a * 3 - 2`	3	?	**7**
3	`if z >= 1:`		True	
4	`y = 6 * a`	3	**18**	7
5	`print(z, ",", y)`	7, 18 is displayed		

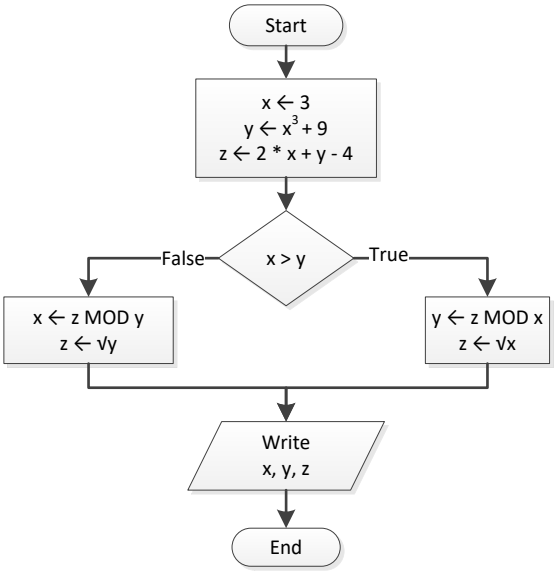
For input value of 0.5

Step	Statement	a	y	z
1	`a = float(input())`	**0.5**	?	?
2	`z = a * 3 - 2`	0.5	?	**-0.5**
3	`if z >= 1:`		False	
4	`z += 1`	0.5	?	**0.5**
5	`y = 6 * a + z`	0.5	**3.5**	0.5
6	`print(z, ",", y)`	0.5, 3.5 is displayed		



2. Solution

Step	Statement	x	y	z
1	`x = 3`	**3**	?	?
2	`y = 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 = math.sqrt(y)`	2	36	**6**
7	`print(x, ",", y, ",", z)`	2, 36, 6 is displayed		



3. Solution

```
x = float(input())
w = x * 3 - 15
z = (w + 7) * (x + 4) - 10
if w > x and z > x:
    x += 1
    y = x / 2 + 4
else:
    y = x / 4 + 2
print(y)
```

For input value of 10

Step	Statement	x	y	w	z
1	`x = float(input())`	**10**	?	?	?
2	`w = x * 3 - 15`	10	?	**15**	?
3	`z = (w + 7) * (x + 4) - 10`	10	?	15	**298**
4	`if w > x and z > x:`	True			
5	`x += 1`	**11**	?	15	298
6	`y = x / 2 + 4`	11	**9.5**	15	298
7	`print(y)`	9.5 is displayed			

For input value of 2

Step	Statement	x	y	w	z
1	`x = float(input())`	**2**	?	?	?
2	`w = x * 3 - 15`	2	?	**-9**	?
3	`z = (w + 7) * (x + 4) - 10`	2	?	-9	**-22**
4	`if w > x and z > x:`	False			
5	`y = x / 4 + 2`	2	**2.5**	-9	-22
6	`print(y)`	2.5 is displayed			

4. Solution

```
x = int(input())
if x % 6 == 0:
    print(x, "is a multiple of 6")
else:
    print(x, "is not a multiple of 6")
```

5. Solution

```
x = int(input())
if x % 6 == 0 or x % 7 == 0:
    print(x, "is a multiple of 6 or a multiple of 7")
else:
    print(x, "is neither a multiple of 6 nor a multiple of 7")
```

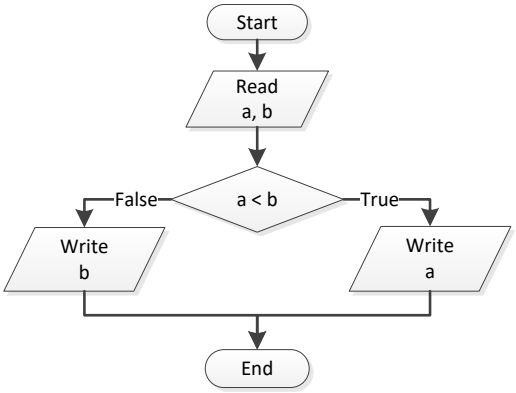
6. Solution

```
x = int(input())

y = x % 4
if y == 0:
    print(x, "is a multiple of 4")
else:
    print(x, "is not a multiple of 4")

print("The structure is:", x, "=", (x // 4), "x 4 +", y)
```

7. Solution

```
x = int(input())

if 1000 <= x <= 9999:
    print(x, "is a four-digit integer")
else:
    print(x, "is not a four-digit integer")
```

8. Solution




```
a = float(input())
b = float(input())

if a < b:
    print(a)
else:
    print(b)
```

9. Solution

```
a = float(input())
b = float(input())
c = float(input())

if a < b + c and b < a + c and c < a + b:
    print("Given numbers can be lengths of the three sides of a triangle")
else:
```

```
    print("Given numbers cannot be lengths of the three sides of a triangle")
```

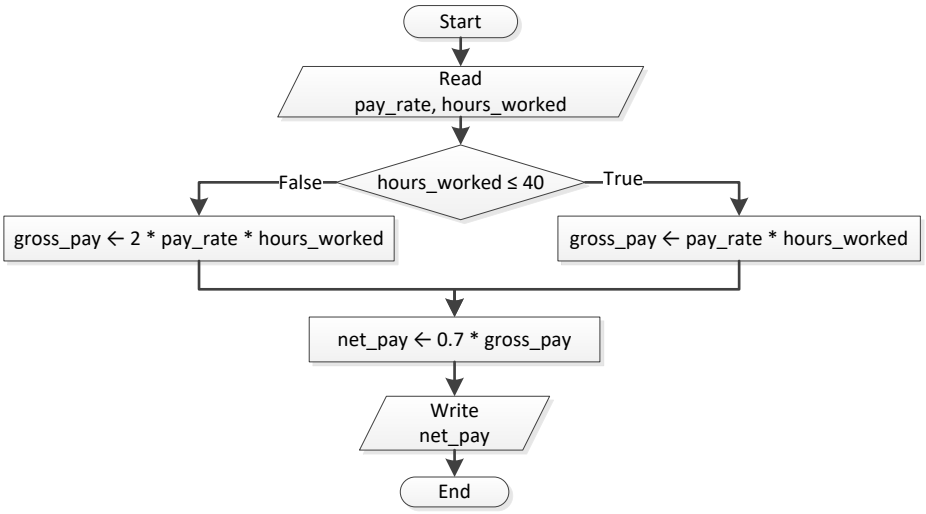
10. Solution

```
a = float(input())
b = float(input())
c = float(input())

if a ** 2 == b ** 2 + c ** 2 or b ** 2 == a ** 2 + c ** 2 or c ** 2 == a ** 2 + b ** 2:
    print("Given numbers can be lengths of the three sides of a right triangle")
else:
    print("Given numbers cannot be lengths of the three sides of a right triangle")
```

11. Solution

```
a = float(input("Enter 1st jump in meters: "))
b = float(input("Enter 2nd jump in meters: "))
c = float(input("Enter 3rd jump in meters: "))

average = (a + b + c) / 3

if average < 8:
    print("Disqualified")
else:
    print("Qualified")
```

12. Solution




```
pay_rate = float(input())
hours_worked = int(input())

if hours_worked <= 40:
    gross_pay = pay_rate * hours_worked
else:
    gross_pay = 2 * pay_rate * hours_worked

net_pay = 0.7 * gross_pay
```

```
print("Net Pay:", net_pay)
```

13. Solution

```
miles = int(input("Enter miles traveled: "))

r = miles % 12000

if r > 6000:
    miles_left = 12000 - r
    print("Your car needs a major service in", miles_left, "miles")
else:
    miles_left = 6000 - r
    print("Your car needs a minor service in", miles_left, "miles")
```

14. Solution

```
t = float(input("Enter the time the two cars traveled: "))
a1 = float(input("Enter the accelaration for car A: "))
a2 = float(input("Enter the accelaration for car B: "))

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

if s1 > s2:
    print("Car A is first")
else:
    print("Car B is first")
```

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	`q = int(input())`	**5**	?
2	`if 0 < q <= 50:`	True	
3	`b = 1`	5	**1**
4	`print(b)`	1 is displayed	

For input value of 150

Step	Statement	q	b
1	`q = int(input())`	**150**	?
2	`if 0 < q <= 50:`	False	
3	`elif 50 < q <= 100:`	False	
4	`elif 100 < q <= 200:`	True	
5	`b = 3`	150	**3**
6	`print(b)`	3 is displayed	

For input value of 250

Step	Statement	q	b
1	`q = int(input())`	**250**	?
2	`if 0 < q <= 50:`	False	
3	`elif 50 < q <= 100:`	False	
4	`elif 100 < q <= 200:`	False	
5	`b = 4`	250	**4**
6	`print(b)`	4 is displayed	

For input value of -1

Step	Statement	q	b
1	`q = int(input())`	**-1**	?
2	`if 0 < q <= 50:`	False	

3	`elif 50 < q <= 100:`	False	
4	`elif 100 < q <= 200:`	False	
5	`b = 4`	-1	**4**
6	`print(b)`	4 is displayed	

2. *Solution*

For input value of 5

Step	Statement	amount	discount	payment
1	`amount = float(input())`	**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	`print(discount, ",", payment)`	0, 5 is displayed.		

For input value of 150

Step	Statement	amount	discount	payment
1	`amount = float(input())`	**150**	?	?
2	`discount = 0`	150	**0**	?
3	`if amount < 20:`	False		
4	`elif 20 <= amount < 60:`	False		
5	`elif 60 <= amount < 100:`	False		
6	`elif amount >= 100:`	True		
7	`discount = 15`	150	**15**	?
8	`payment = amount - amount * discount / 100`	150	15	**5**
9	`print(discount, ",", payment)`	15, 127.5 is displayed.		

For input value of -1

Step	Statement	amount	discount	payment
1	`amount = float(input())`	**-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	`print(discount, ",", payment)`	0, -1 is displayed.		

3. Solution

For input value of 1

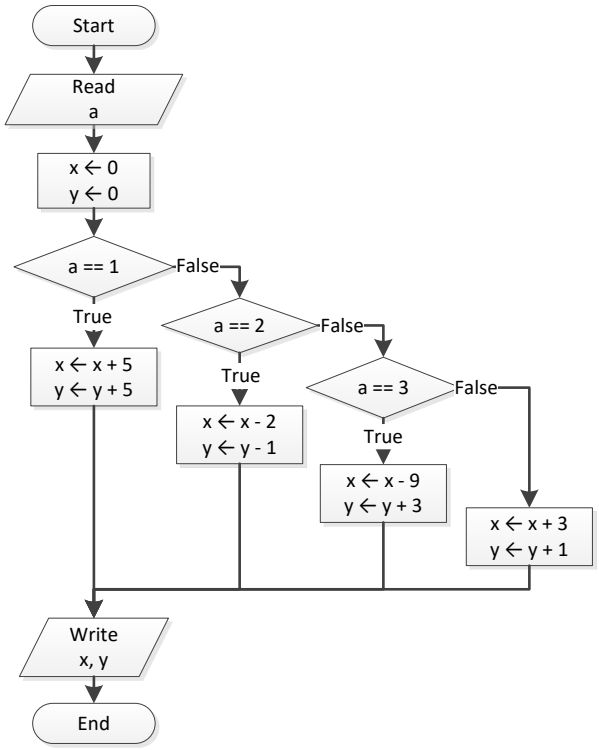
Step	Statement	a	x	y
1	`a = int(input())`	**1**	?	?
2	`x = 0`	1	**0**	?
3	`y = 0`	1	0	**0**
4	`if a == 1:`		True	
5	`x = x + 5`	1	**5**	0
6	`y = y + 5`	1	5	**5**
7	`print(x, ",", y)`	5, 5 is displayed		

For input value of 3

Step	Statement	a	x	y
1	`a = int(input())`	**3**	?	?
2	`x = 0`	3	**0**	?
3	`y = 0`	3	0	**0**
4	`if a == 1:`		False	
5	`elif a == 2:`		False	
6	`elif a == 3:`		True	
7	`x = x - 9`	3	-9	0
8	`y = y + 3`	3	-9	3
9	`print(x, ",", y)`	-9, 3 is displayed		

For input value of 250

Step	Statement	a	x	y
1	`a = int(input())`	**250**	?	?
2	`x = 0`	250	**0**	?
3	`y = 0`	250	0	**0**
4	`if a == 1:`		False	
5	`elif a == 2:`		False	
6	`elif a == 3:`		False	
7	`x = x + 3`	250	3	0
8	`y += 1`	250	3	1
9	`print(x, ",", y)`	3, 1 is displayed		



4. *Solution*

For input values of 10, 2, 5

Step	Statement	a	x	y
1	`a = int(input())`	**10**	?	?
2	`x = int(input())`	10	**2**	?
3	`y = float(input())`	10	2	**5**
4	`if a == 10:`		True	
5	`x = x % 2`	10	**0**	5
6	`y = y ** 2`	10	0	**25**
7	`print(x, ",", y)`	0, 25 is displayed		

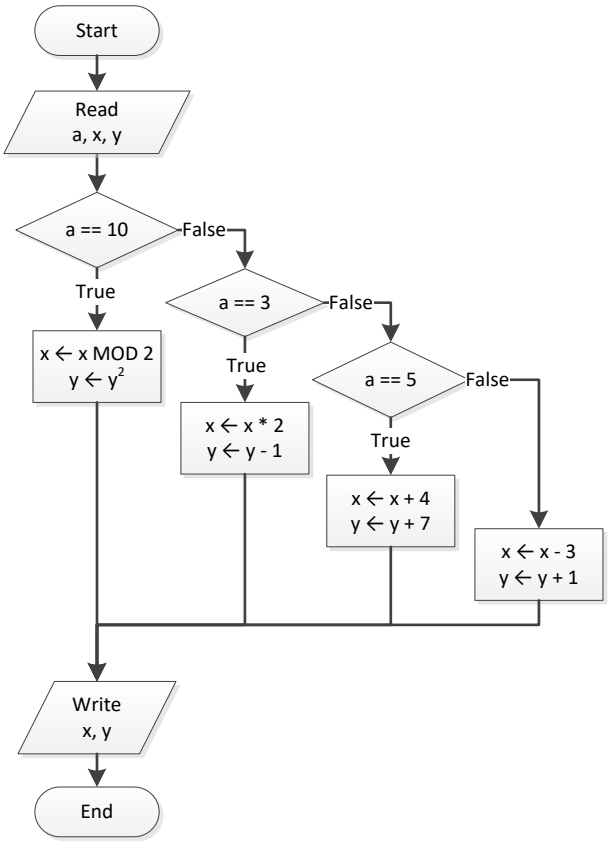
For input values of 5, 2, 3

Step	Statement	a	x	y
1	`a = int(input())`	**5**	?	?
2	`x = int(input())`	5	**2**	?
3	`y = float(input())`	5	2	**3**
4	`if a == 10:`		False	
5	`elif a == 3:`		False	
6	`elif a == 5:`		True	
7	`x = x + 4`	5	**6**	3

8	`y += 7`	5	6	**10**
9	`print(x, ",", y)`	6, 10 is displayed		

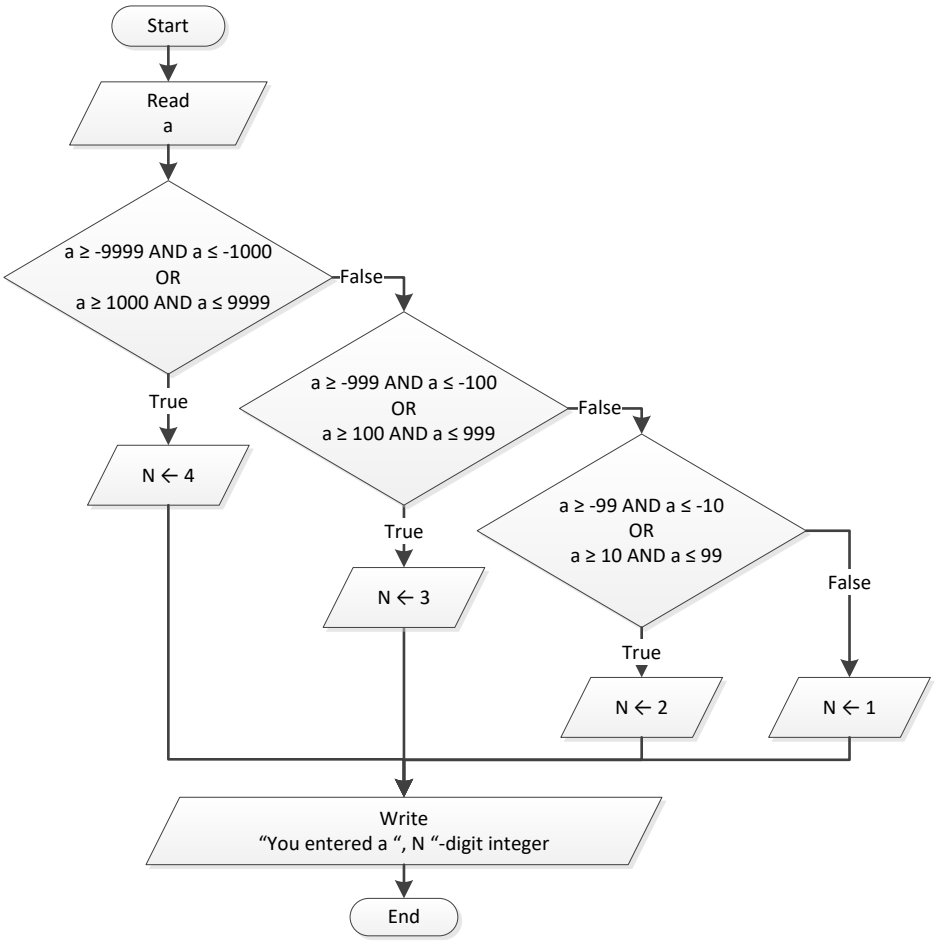
For input values of 4, 6, 2

Step	Statement	a	x	y
1	`a = int(input())`	**4**	?	?
2	`x = int(input())`	4	**6**	?
3	`y = float(input())`	4	6	**2**
4	`if a == 10:`		False	
5	`elif a == 3:`		False	
6	`elif a == 5:`		False	
7	`x -= 3`	4	**3**	2
8	`y += 1`	4	3	**3**
9	`print(x, ",", y)`	3, 3 is displayed		



5. Solution

First Approach




```
a = int(input())

if -9999 <= a <= -1000 or 1000 <= a <= 9999:
    n = 4
elif -999 <= a <= -100 or 100 <= a <= 999:
    n = 3
elif -99 <= a <= -10 or 10 <= a <= 99:
    n = 2
else:
    n = 1

print("You entered a ", n, "-digit integer", sep = "")
```

Second Approach

```
a = int(input())
a_string = str(abs(a))
print("You entered a ", len(a_string), "-digit integer", sep = "")
```

6. Solution

```
print("1. Convert USD to Euro (EUR)")
print("2. Convert USD to British Pound Sterling (GBP)")
print("3. Convert USD to Japanese Yen (JPY)")
print("4. Convert USD to Canadian Dollar (CAD)")
ch = int(input("Enter a choice: "))

usd = float(input("Enter an amount in US dollars: "))

if ch == 1:
    eur = usd / 0.72
    print("$", usd, " = ", eur, " EUR", sep = "")
elif ch == 2:
    gbp = usd / 0.60
    print("$", usd, " = ", gbp, " GBP", sep = "")
elif ch == 3:
    jpy = usd / 102.15
    print("$", usd, " = ", jpy, " JPY", sep = "")
else:
    cad = usd / 1.10
    print("$", usd, " = ", cad, " CAD", sep = "")
```

7. Solution

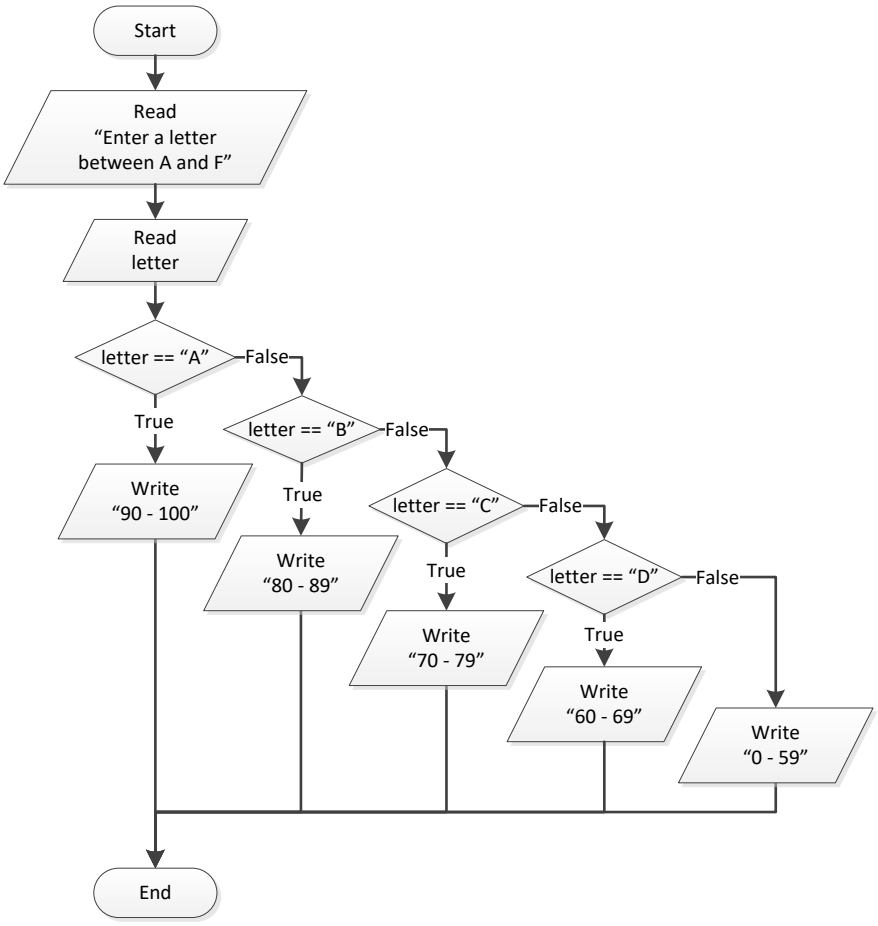
```
m = int(input("Enter the number of a month between 1 and 12: "))

if m <= 2 or m == 12:
    print("Winter")
elif m <= 5:
    print("Spring")
elif m <= 8:
    print("Summer")
else:
    print("Fall (Autumn)")
```

8. Solution

```
n = float(input("Enter a number between 1.0 and 4.9: "))

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

if x == 1:
    print("One", end = "")
elif x == 2:
    print("Two", end = "")
elif x == 3:
    print("Three", end = "")
elif x == 4:
    print("Four", end = "")

print(" point ", end = "")
```

```
if y == 1:
    print("one")
elif y == 2:
    print("two")
elif y == 3:
    print("three")
elif y == 4:
    print("four")
elif y == 5:
    print("five")
elif y == 6:
    print("six")
elif y == 7:
    print("seven")
elif y == 8:
    print("eight")
elif y == 9:
    print("nine")
elif y == 0:
    print("zero")
```

9. Solution

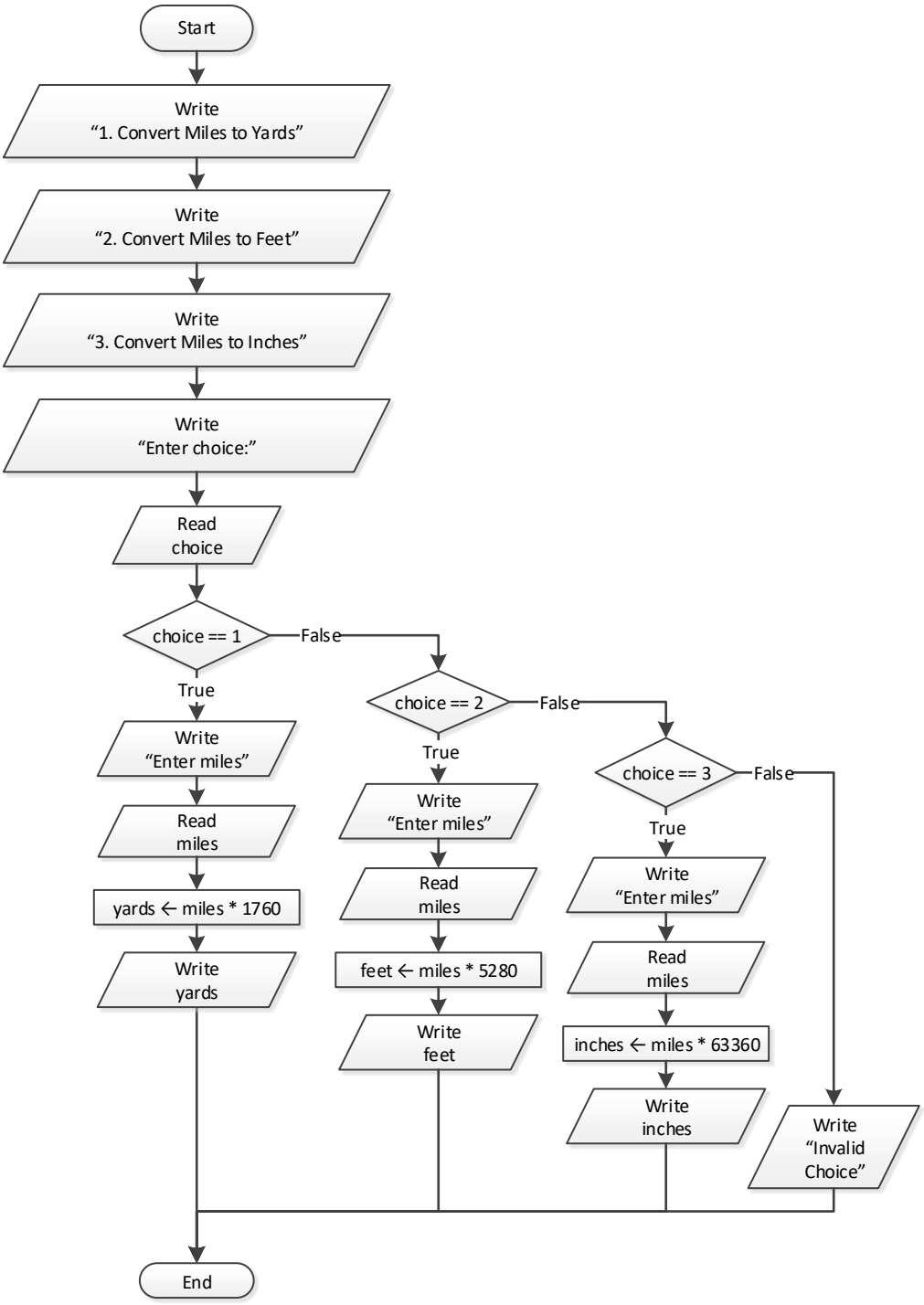


```
letter = input("Enter a letter between A and F: ")

if letter == "A":
    print("90 - 100")
elif letter == "B":
    print("80 - 89")
elif letter == "C":
    print("70 - 79")
elif letter == "D":
    print("60 - 69")
else:
    print("0 - 59")
```

10. Solution

```
name = input("Enter the name of a month: ")

if name == "JANUARY":
    print(1)
elif name == "FEBRUARY":
    print(2)
elif name == "MARCH":
    print(3)
elif name == "APRIL":
    print(4)
elif name == "MAY":
    print(5)
elif name == "JUNE":
    print(6)
elif name == "JULY":
    print(7)
elif name == "AUGUST":
    print(8)
elif name == "SEPTEMBER":
    print(9)
elif name == "OCTOBER":
    print(10)
elif name == "NOVEMBER":
    print(11)
elif name == "DECEMBER":
    print(12)
else:
    print("Error")
```

11. Solution




```
print("1. Convert Miles to Yards")
print("2. Convert Miles to Feet")
print("3. Convert Miles to Inches")

choice = int(input("Enter a choice: "))
```

```
if choice == 1:
    miles = float(input("Enter miles: "))
    yards = miles * 1760
    print(miles, "miles =", yards, "yards")
elif choice == 2:
    miles = float(input("Enter miles: "))
    feet = miles * 5280
    print(miles, "miles =", feet, "feet")
elif choice == 3:
    miles = float(input("Enter miles: "))
    inches = miles * 63360
    print(miles, "miles =", inches, "inches")
else:
    print("Invalid choice!")
```

12. Solution

```
roman = input("Enter a Roman number between I and X: ")

if roman == "I":
    print(1)
elif roman == "II":
    print(2)
elif roman == "III":
    print(3)
elif roman == "IV":
    print(4)
elif roman == "V":
    print(5)
elif roman == "VI":
    print(6)
elif roman == "VII":
    print(7)
elif roman == "VIII":
    print(8)
elif roman == "IX":
    print(9)
elif roman == "X":
    print(10)
else:
    print("Error")
```

13. Solution

```
total = int(input("Enter the total number of CDs purchased in a month: "))

if total == 1:
    print("You are awarded 3 points")
elif total == 2:
    print("You are awarded 10 points")
elif total == 3:
    print("You are awarded 20 points")
```

```
else:
    print("You are awarded 45 points")
```

14. Solution

```
import random

name = input("Enter your name: ")

i = random.randrange(3)

if i == 0:
    print("Good morning", name)
elif i == 1:
    print("Good evening", name)
elif i == 2:
    print("Good night", name)
```

15. Solution

```
num = input()

if num == "ZERO":
    print(0)
elif num == "ONE":
    print(1)
elif num == "TWO":
    print(2)
elif num == "THREE":
    print(3)
elif num == "FOUR":
    print(4)
elif num == "FIVE":
    print(5)
elif num == "SIX":
    print(6)
elif num == "SEVEN":
    print(7)
elif num == "EIGHT":
    print(8)
elif num == "NINE":
    print(9)
else:
    print("I don't know this number!")
```

16. Solution

```
b = int(input("Enter Beaufort number: "))

if b == 0:
    print("Calm")
elif b == 1:
    print("Light Air")
```

```
elif b == 2:
    print("Light breeze")
elif b == 3:
    print("Gentle breeze")
elif b == 4:
    print("Moderate breeze")
elif b == 5:
    print("Fresh breeze")
elif b == 6:
    print("Strong breeze")
elif b == 7:
    print("Moderate gale")
elif b == 8:
    print("Gale")
elif b == 9:
    print("Strong gale")
elif b == 10:
    print("Storm")
elif b == 11:
    print("Violent storm")
elif b == 12:
    print("Hurricane force")
else:
    print("Invalid Beaufort number!")
```

Chapter 19

19.3 Answers of Review Questions: True/False

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

19.4 Answers of Review Exercises

1. Solution

For input values of 20, 1

Step	Statement	x	y
1	`x = int(input())`	**20**	?
2	`y = int(input())`	20	**1**
3	`if x < 30:`	True	
4	`if y == 1:`	True	
5	`x = x % 3`	**2**	1
6	`y = 5`	2	**5**
7	`print(x, ",", y)`	2, 5 is displayed	

For input values of 20, 3

Step	Statement	x	y
1	`x = int(input())`	**20**	?
2	`y = int(input())`	20	**3**
3	`if x < 30:`	True	
4	`if y == 1:`	False	
5	`elif y == 2:`	False	
6	`elif y == 3:`	True	
7	`x = x + 5`	**25**	3
8	`y += 3`	25	**6**
9	`print(x, ",", y)`	25, 6 is displayed	

For input values of 12, 8

Step	Statement	x	y
1	`x = int(input())`	**12**	?
2	`y = int(input())`	12	**8**
3	`if x < 30:`	True	
4	`if y == 1:`	False	
5	`elif y == 2:`	False	

6	`elif y == 3:`	False	
7	`x -= 2`	**10**	8
8	`y += 1`	10	**9**
9	`print(x, ",", y)`	10, 9 is displayed	

For input values of 50, 0

Step	Statement	x	y
1	`x = int(input())`	**50**	?
2	`y = int(input())`	50	**0**
3	`if x < 30:`	False	
4	`y += 1`	50	**1**
5	`print(x, ",", y)`	50, 1 is displayed	

2. *Solution*

For input values of 60, 25

Step	Statement	x	y
1	`x = int(input())`	**60**	?
2	`y = int(input())`	60	25
3	`if (x + y) / 2 <= 20:`	False	
4	`if y < 15:`	False	
5	`elif y < 23:`	False	
6	`x = 2 * x + 5`	**125**	25
7	`y += 1`	125	**26**
8	`print(x, ",", y)`	125, 26 is displayed	

For input values of 50, 8

Step	Statement	x	y
1	`x = int(input())`	**50**	?
2	`y = int(input())`	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	`print(x, ",", y)`	2, 2 is displayed	

For input values of 20, 15

Step	Statement	x	y
1	`x = int(input())`	**20**	?
2	`y = int(input())`	20	15

3	`if (x + y) / 2 <= 20:`	True	
4	`if y < 10:`	False	
5	`elif y < 20:`	True	
6	`x = x * 5`	**100**	15
7	`y += 2`	100	**17**
8	`print(x, ",", y)`	100, 17 is displayed	

For input values of 10, 30

Step	Statement	x	y
1	`x = int(input())`	**10**	?
2	`y = int(input())`	10	30
3	`if (x + y) / 2 <= 20:`	True	
4	`if y < 10:`	False	
5	`elif y < 20:`	False	
6	`x = x - 2`	**8**	30
7	`y += 3`	8	**33**
8	`print(x, ",", y)`	8, 33 is displayed	

3. Solution

```
print("Enter the three sides of a triangle: ")
a = float(input())
b = float(input())
c = float(input())

if a >= b + c or b >= a + c or c >= a + b:
    print("Given numbers cannot be lengths of the three sides of a triangle")
else:
    if a == b and b == c:
        print("Equilateral")
    elif a ** 2 == b ** 2 + c ** 2 or b ** 2 == a ** 2 + c ** 2 or c ** 2 == a ** 2 + b ** 2:
        print("Right (or right-angled)")
    else:
        print("not special")
```

4. Solution

```
pin = int(input("Enter your four-digit PIN : "))
if pin != 1234:
    pin = int(input("Wrong PIN. Enter your four-digit PIN : "))
    if pin != 1234:
        pin = int(input("Wrong PIN. Enter your four-digit PIN : "))

if pin != 1234:
    print("PIN locked!")
else:
```

```
    amount = int(input("Enter the amount of money (an integer value) that you want to withdraw: "))
    usd10 = amount // 10
    r = amount % 10
    usd5 = r // 5
    usd1 = r % 5
    print(usd10, "notes of $10", usd5, "notes of $5", "and", usd1, "notes of $1")
```

5. Solution

First Approach

```
t = float(input("Enter temperature (in Fahrenheit): "))
w = float(input("Enter wind speed (in miles/hour): "))

if t > 75:
    if w > 12:
        print("The day is hot and windy")
    else:
        print("The day is hot and not windy")
else:
    if w > 12:
        print("The day is cold and windy")
    else:
        print("The day is cold and not windy")
```

Second Approach

```
t = float(input("Enter temperature (in Fahrenheit): "))
w = float(input("Enter wind speed (in miles/hour): "))

if t > 75:
    message1 = "hot"
else:
    message1 = "cold"

if w > 12:
    message2 = "windy"
else:
    message2 = "not windy"

print("The day is", message1, "and", message2)
```

Chapter 20

20.10 Answers of Review Questions: True/False

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

20.11 Answers of Review Questions: Multiple Choice

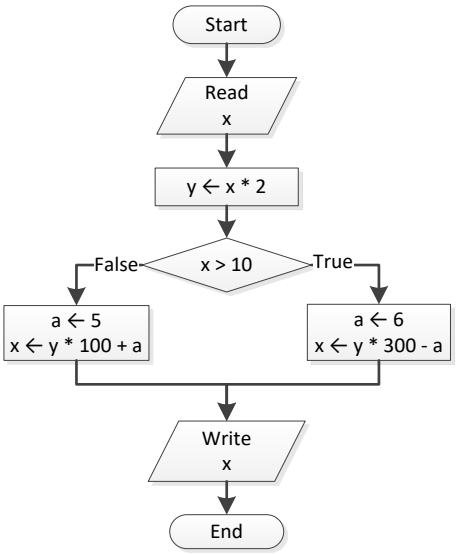
1. c
2. b
3. a
4. a
5. c
6. c
7. c
8. a

20.12 Answers of Review Exercises

1. Solution

```
y = int(input())
x = int(input())

if y > 0:
    a = x * 4 * y + 1
else:
    a = x * 2 * y + 6

print(y)
print(a)
```

2. Solution



3. Solution

```
a = float(input())

if a >= 10:
    print("Error!")
else:
    if a < 1:
        y = 5 + a
    elif a < 5:
        y = 23 / a
    else:
        y = 5 * a
    print(y)
```

4. Solution

```
day = int(input())
month = int(input())
name = input()

if day == 16 and month == 2 and name == "Loukia":
    print("Happy Birthday!!!")
else:
    print("No match!")
```

5. Solution

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

```
a = float(input())
b = float(input())
c = float(input())

if a > 10:
    if c < 2000:
        d = (a + b + c) / 12
        print("The result is:", d)
    else:
        print("Error!")
else:
    print("Error!")
```

6. Solution

```
a = float(input())
b = float(input())
c = float(input())

if a > 10 and b < 2000 and c != 10:
    d = (a + b + c) / 12
    print("The result is:", d)
```

```
if a <= 10:
    print("Error!")
```

7. Solution

```
a = int(input())
b = int(input())

y = 3
if a > 0:
    y = y * a
    print("Hello Zeus")

print(y, b)
```

8. Solution

```
a = float(input())
b = float(input())

y = 0
if a > 0:
    y = y + 7
else:
    print("Hello Zeus")
    print(abs(a))
print(y)
```

9. Solution

```
os = input("What is your tablet's OS? ")

if os == "iOS":
    print("Apple")
elif os == "android":
    print("Google")
elif os == "Windows":
    print("Microsoft")
```

10. Solution

```
a = int(input())
x = float(input())
y = float(input())

if a in (3, 5, 15):
    x = x / 4
    y = y ** 5
elif a >= 7 and a <= 12:
    x = x * 3
    y += 1
elif a > 52:
    x = x % 4
```

```
    y += 9
else:
    x -= 9
    y += 1

print(x, y)
```

11. Solution

```
print("1. Red")
print("2. Green")
print("3. Blue")
print("4. White")
print("5. Black")
print("6. Gray")
color = int(input("Select a color: "))

print("Your color in hexadecimal is: ")

if color == 1:
    print("FF0000")
else:
    if color == 2:
        print("00FF00")
    else:
        if color == 3:
            print("0000FF")
        else:
            if color == 4:
                print("FFFFFF")
            else:
                if color == 5:
                    print("000000")
                else:
                    if color == 6:
                        print("7F7F7F")
                    else:
                        print("Unknown color!")
```

12. Solution

```
a = int(input())
if a > 1000:
    print("Big Positive")
else:
    if a > 0:
        print("Positive")
    else:
        if a < -1000:
            print("Big Negative")
        else:
            if a < 0:
                print("Negative")
```

```
            else:
                print("Zero")
```

```
a = int(input())

if a > 1000:
    print("Big Positive")
elif a > 0:
    print("Positive")
elif a < -1000:
    print("Big Negative")
elif a < 0:
    print("Negative")
else:
    print("Zero")
```

13. Solution

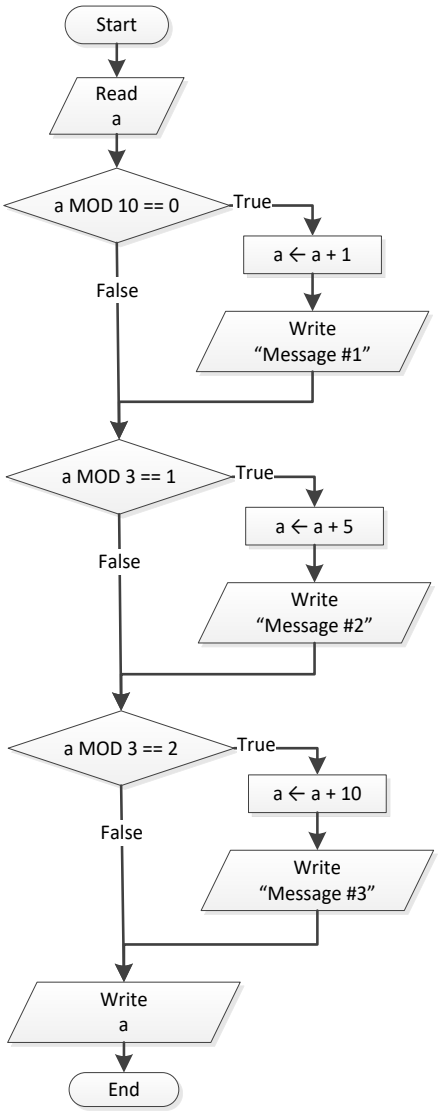
```
a = float(input())

if a < 1:
    y = 5 + a
    print(y)
elif a < 5:
    y = 23 / a
    print(y)
elif a < 10:
    y = 5 * a
    print(y)
else:
    print("Error!")
```

```
a = float(input())

if a < 1:
    y = 5 + a
    print(y)
else:
    if a < 5:
        y = 23 / a
        print(y)
    else:
        if a < 10:
            y = 5 * a
            print(y)
        else:
            print("Error!")
```

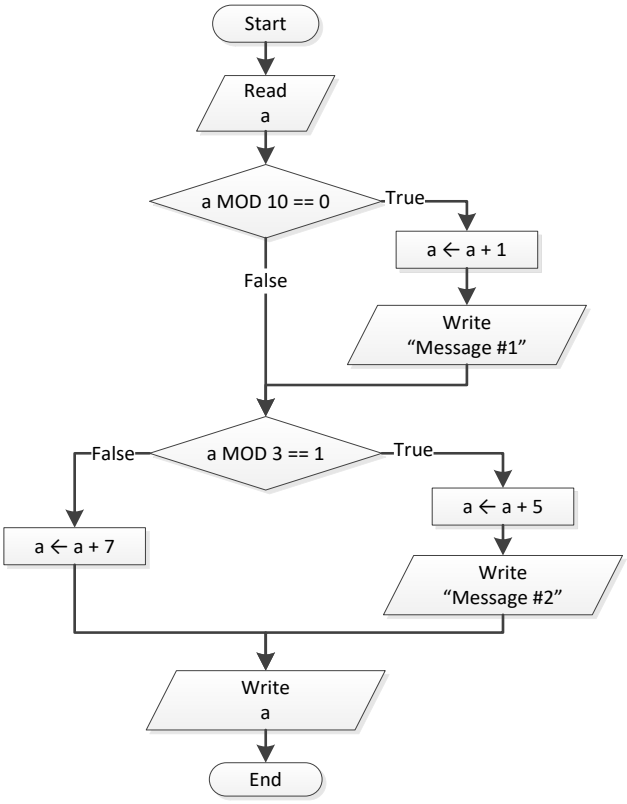
Chapter 21

21.4 Answers of Review Exercises

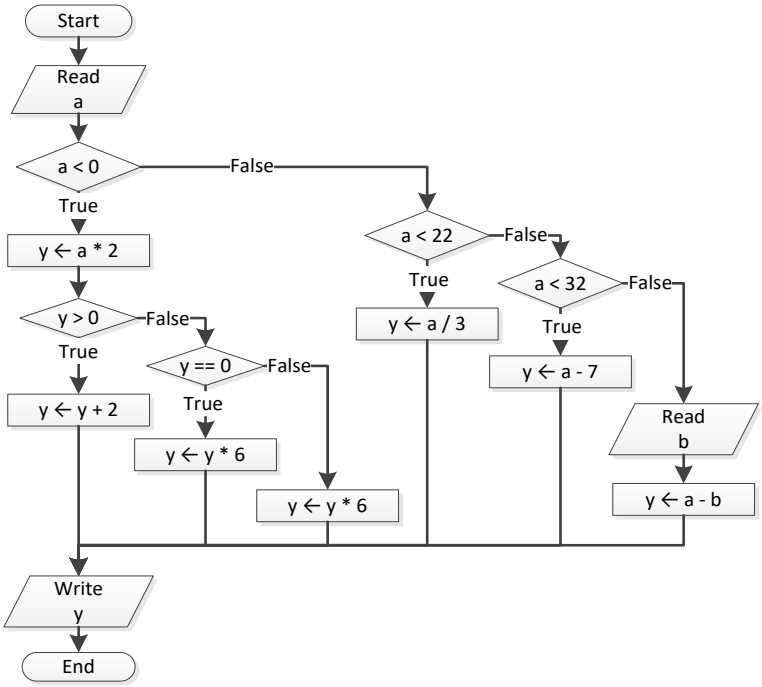
1. Solution



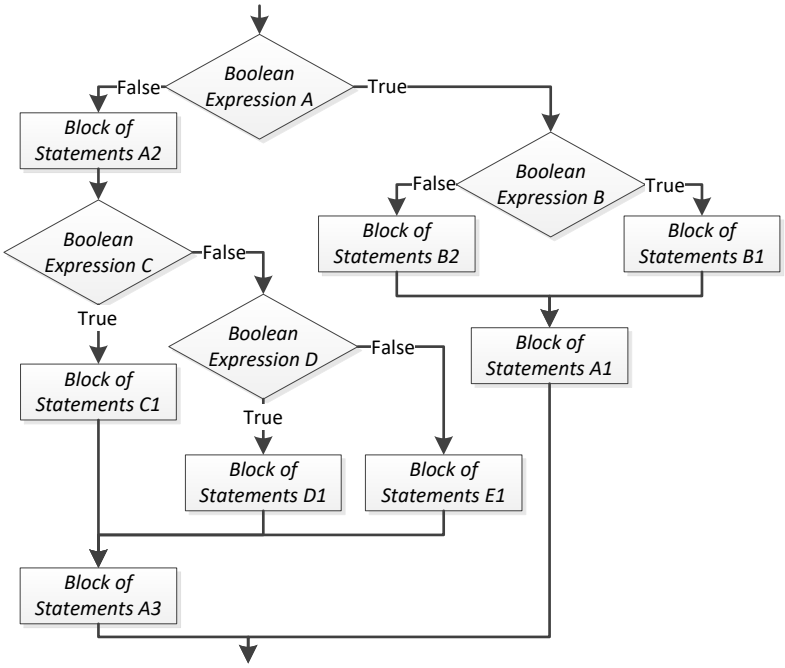
2. Solution



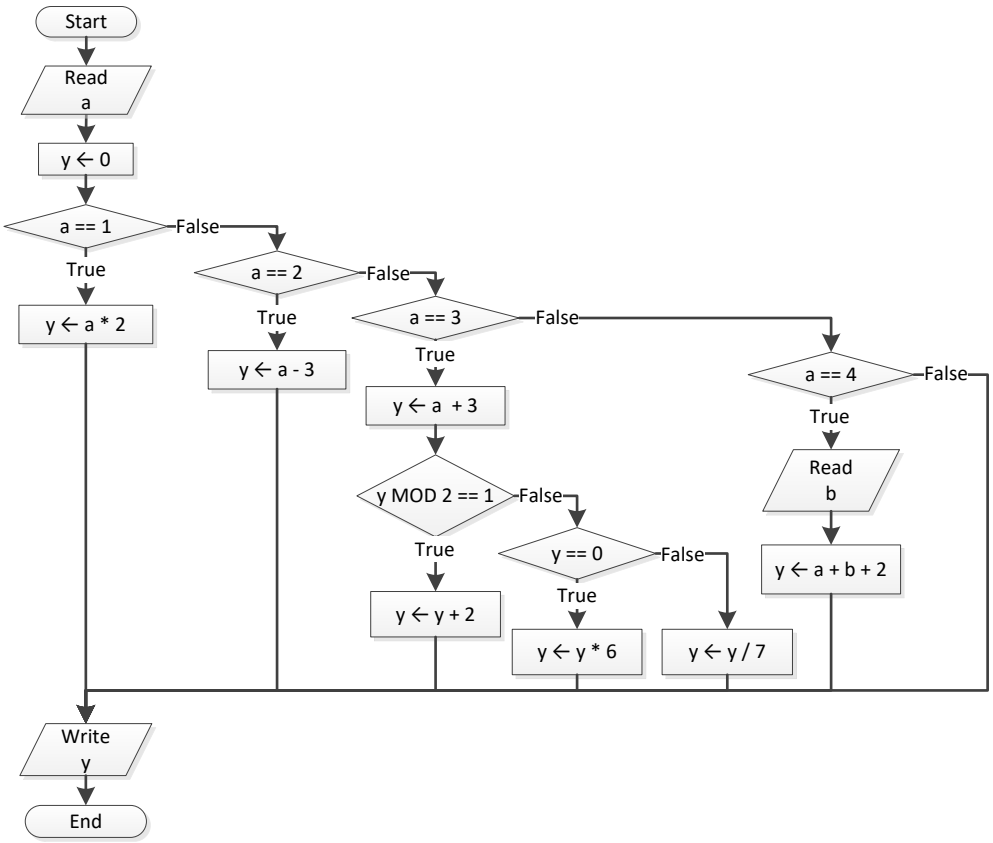
3. Solution



4. Solution



5. Solution



6. Solution

```
x = float(input())
y = float(input())

if x != 100 or y <= 10:
    z = float(input())
    if z <= x + y:
        x -= 3
        y = x + 4
print(x, y)
```

7. Solution

```
x = int(input())

if x == 1:
    print("Good Morning")
    print("How Do you do?")
    print("Is everything okay?")
elif x == 2:
    print("Good Evening")
    print("How Do you do?")
    print("Is everything okay?")
elif x == 3:
    print("Good Afternoon")
    print("Is everything okay?")
else:
    print("Good Night")
```

8. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

inp = input()
if re.match(IS_NUMERIC, inp):
    x = int(inp)
    if x % 10 == 0:
        print("Last digit equal to 0")
    elif x % 10 == 1:
        print("Last digit equal to 1")
    else:
        print("None")
else:
    if inp == "Exit":
        print("Bye")
    else:
        print("Invalid Number")
```

9. Solution

```
a = float(input())
```

```
b = float(input())

y = a * b

if y > 0:
    y -= 1
    y /= 2
else:
    y +=10
    if y > 0:
        y /= 2
    else:
        y *= 2
```

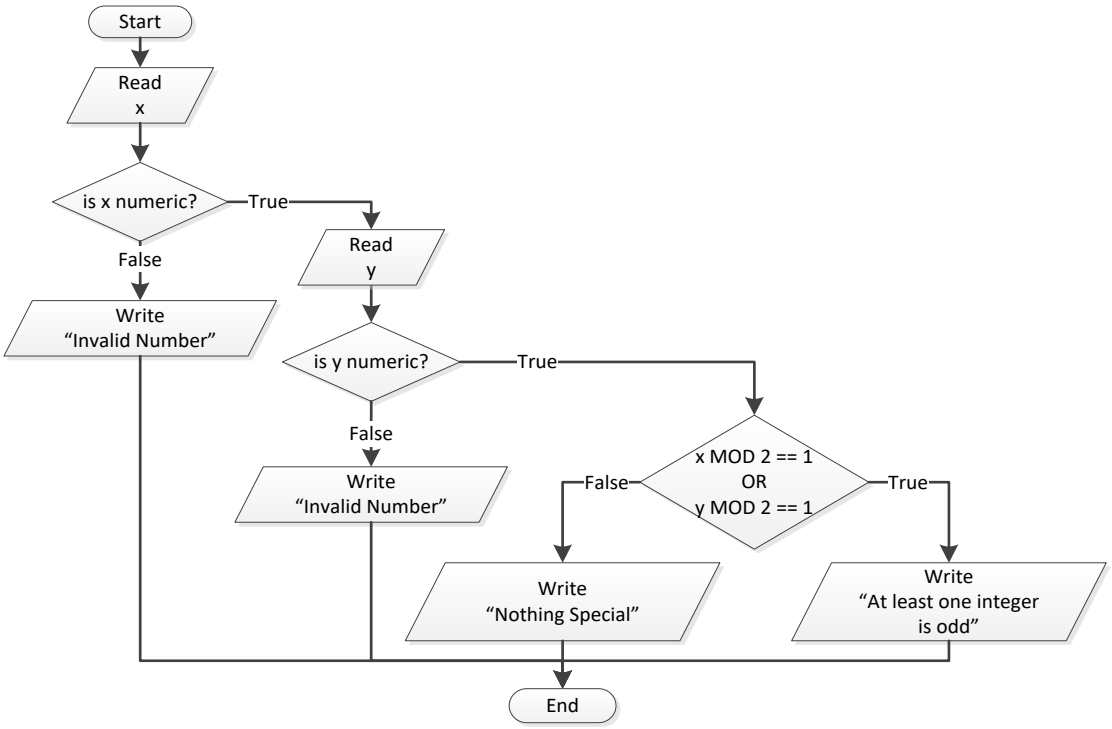
10. Solution

```
a = float(input())
b = float(input())
c = float(input())

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
print(a, b, c)
```

Chapter 22

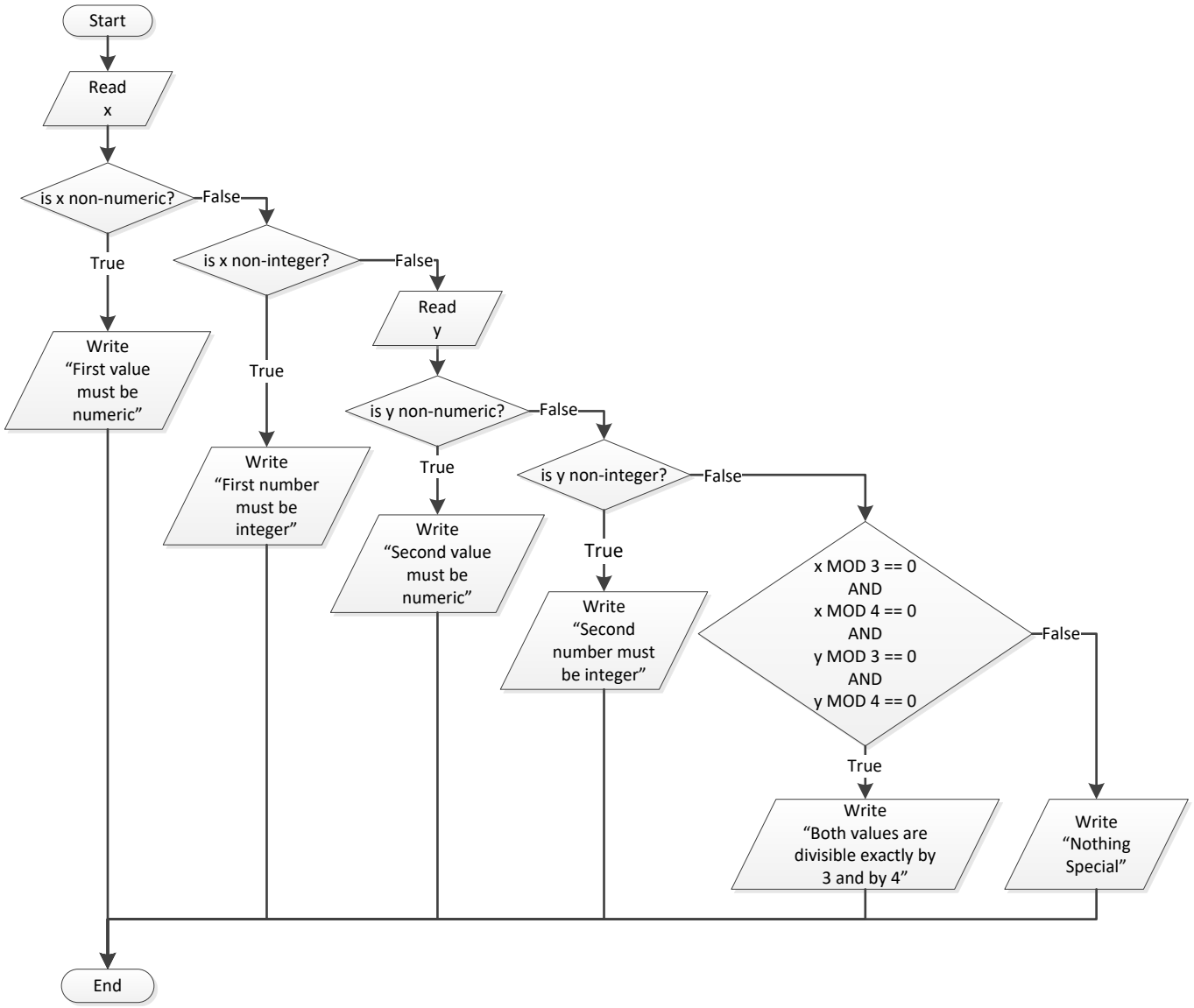
22.6 Answers of Review Exercises

1. Solution




```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

inp = input()

if re.match(IS_NUMERIC, inp):
    x = int(inp)
    inp = input()
    if re.match(IS_NUMERIC, inp):
        y = int(inp)
        if x % 2 == 1 or y % 2 == 1:
            print("At least one integer is odd")
        else:
            print("nothing Special")
    else:
        print("Invalid Number")
else:
    print("Invalid Number")
```

2. Solution




```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

inp = input()

if not re.match(IS_NUMERIC, inp):
    print("First value must be numeric")
else:
    x = float(inp)
    if x != int(x):
        print("First number must be integer")
    else:
        inp = input()
        if not re.match(IS_NUMERIC, inp):
            print("Second value must be numeric")
```

```
        else:
            y = float(inp)
            if y != int(y):
                print("Second number must be integer")
            else:
                if x % 3 == 0 and x % 4 == 0 and y % 3 == 0 and y % 4 == 0:
                    print("Both values are divisible exactly by 3 and by 4")
                else:
                    print("nothing Special")
```

3. Solution

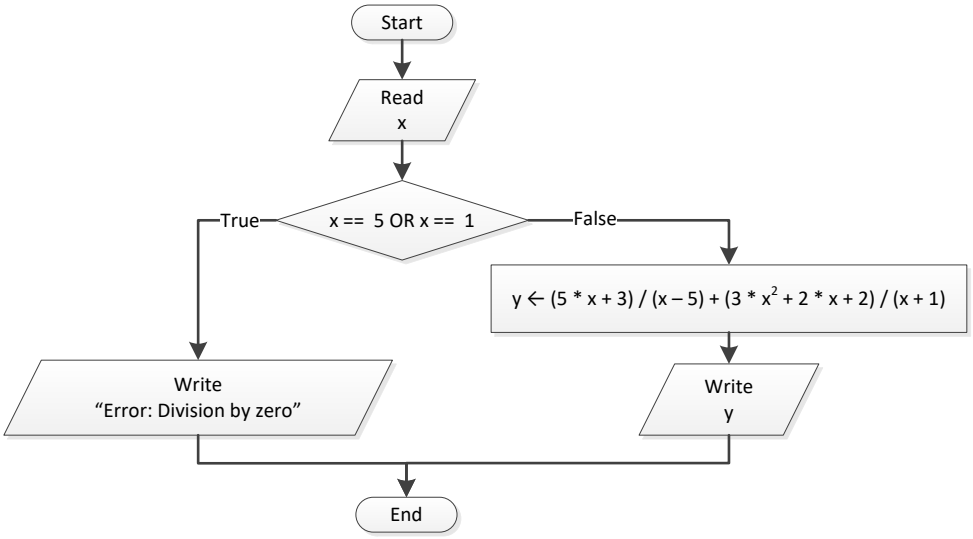
```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

print("1. Convert Kelvin to Fahrenheit")
print("2. Convert Fahrenheit to Kelvin")
print("3. Convert Fahrenheit to Celsius")
print("4. Convert Celsius to Fahrenheit")

choice = int(input("Enter a choice: "))
inp = input("Enter a temperature: ")

if choice < 1 or choice > 4:
    print("Wrong choice")
elif not re.match(IS_NUMERIC, inp):
    print("Wrong temperature")
else:
    t = float(inp)
    if choice == 1:
        print(1.8 * t - 459.67)
    elif choice == 2:
        print((t + 459.57) / 1.8)
    elif choice == 3:
        print(5 / 9 * (t - 32))
    elif choice == 4:
        print(9 / 5 * t + 32)
```

4. Solution

```
a = int(input("Enter 1st integer: "))
op = input("Enter type of operation: ")
b = int(input("Enter 2nd integer: "))

if op == "+":
    print(a, b)
elif op == "-":
    print(a - b)
elif op == "*":
    print(a * b)
elif op == "/":
    if b == 0:
        print("Error: Division by zero")
```

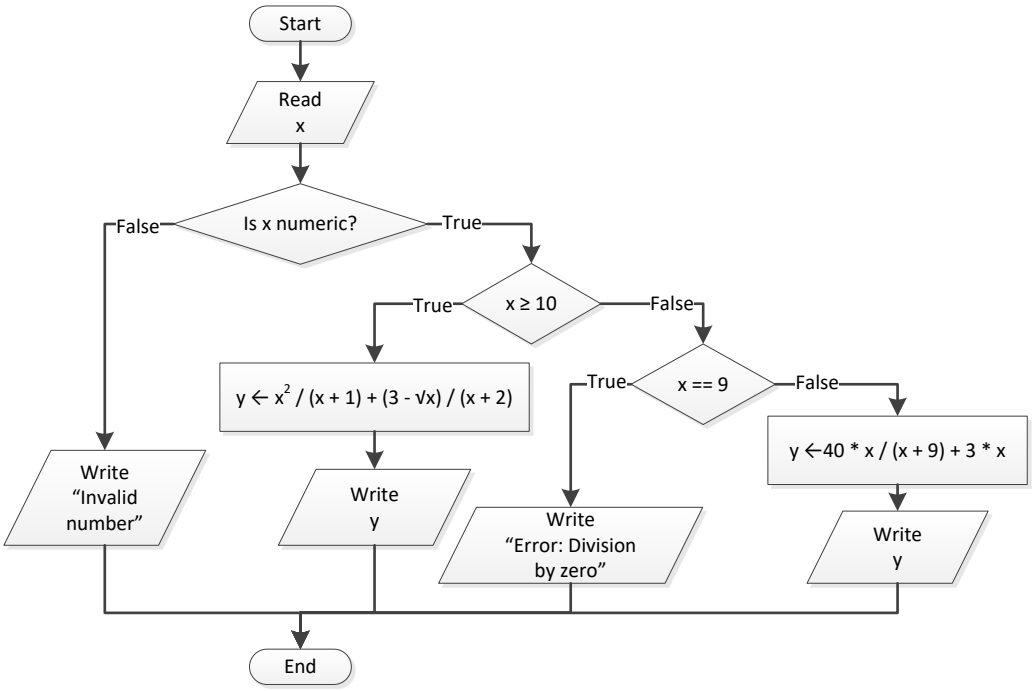
```
    else:
        print(a / b)
elif op == "DIV":
    if b == 0:
        print("Error: Division by zero")
    else:
        print(a // b)
elif op == "MOD":
    if b == 0:
        print("Error: Division by zero")
    else:
        print(a % b)
elif op == "POWER":
    print(a ** b)
```

5. Solution




```
x = float(input())

if x == 5 or x == 1:
    print("Error: Division by zero")
else:
    y = (5 * x + 3) / (x - 5) + (3 * x ** 2 + 2 * x + 2) / (x + 1)
    print(y)
```

6. Solution




```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

import math

inp = input()

if re.match(IS_NUMERIC, inp):
    x = float(inp)
    if x >= 10:
        y = x ** 2 / (x + 1) + (3 - math.sqrt(x)) / (x + 2)
        print(y)
    else:
        if x == 9:
            print("Error: Division by zero")
        else:
            y = 40 * x / (x + 9) + 3 * x
            print(y)
else:
    print("Invalid number")
```

7. Solution

```
import math

x = float(input())

if x <= -15 or x > 25:
    y = x - 1
```

```
    print(y)
elif x <= -10:
    y = x / math.sqrt(x + 30) + (8 + x) ** 2 / (x + 1)
    print(y)
elif x <= 0:
    y = abs(40 * x) / (x - 8)
    print(y)
else:
    if x == 9:
        print("Error: Division by zero")
    elif x < 9:
        print("Error: Invalid square root")
    else:
        y = 3 * x / math.sqrt(x - 9)
        print(y)
```

8. Solution

```
age1 = int(input("Enter age for person No1:"))
age2 = int(input("Enter age for person No2:"))
age3 = int(input("Enter age for person No3:"))

minimum = age1
if age2 < minimum:
    minimum = age2
if age3 < minimum:
    minimum = age3
maximum = age1
if age2 > maximum:
    maximum = age2
if age3 > maximum:
    maximum = age3

middle = age1 + age2 + age3 - minimum - maximum
print(middle)
```

9. Solution

```
a1 = int(input("Enter the age of the first person: "))
n1 = input("Enter the name of the first person: ")
a2 = int(input("Enter the age of the second person: "))
n2 = input("Enter the name of the second person: ")
a3 = int(input("Enter the age of the third person: "))
n3 = input("Enter the name of the third person: ")

minimum = a1
min_name = n1
if a2 > minimum:
    minimum = a2
    min_name = n2
if a3 > minimum:
    minimum = a3
    min_name = n3
```

```
maximum = a1
max_name = n1
if a2 > maximum:
    maximum = a2
    max_name = n2
if a3 > maximum:
    maximum = a3
    max_name = n3

middle = a1 + a2 + a3 - minimum - maximum

if abs(maximum - middle) < abs(minimum - middle):
    print(max_name)
else:
    print(min_name)
```

10. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

inp = input("Enter a three-digit integer: ")

if not re.match(IS_NUMERIC, inp):
    print("Entered value contains non-numeric characters")
else:
    x = int(inp)
    if x < 100 or x > 999:
        print("Entered integer is not a three-digit integer")
    else:
        digit1, r = divmod(x, 100)
        digit2, digit3 = divmod(r, 10)

        total = digit1 ** 3 + digit2 ** 3 + digit3 ** 3

        if total == x:
            print("You entered an Armstrong number!")
        else:
            print("You entered a non-Armstrong number!")
```

11. Solution

```
d = int(input("Enter day 1 - 31: "))
m = int(input("Enter month 1 - 12: "))
y = int(input("Enter year: "))

if m == 2:
    if y % 4 == 0 and y % 100 != 0 or y % 400 == 0:
        print(29 - d)
    else:
        print(28 - d)
elif m == 4 or m == 6 or m == 9 or m == 11:
```

```
    print(30 - d)
else:
    print(31 - d)
```

12. Solution

```
word = input()

word1 = word[0].upper() +       \
        word[1].lower() +       \
        word[2].upper() +       \
        word[3].lower() +       \
        word[4].upper() +       \
        word[5].lower()

word2 = word[0].lower() +       \
        word[1].upper() +       \
        word[2].lower() +       \
        word[3].upper() +       \
        word[4].lower() +       \
        word[5].upper()

if word == word1 or word == word2:
    print("Word is okay!")
else:
    print("Word is not okay")
```

13. Solution

```
q = int(input("Enter quantity: "))

if q < 3:
    discount = 0
elif q < 6:
    discount = 10
elif q < 10:
    discount = 15
elif q < 14:
    discount = 20
elif q < 20:
    discount = 27
else:
    discount = 30

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

print("You got a discount of ", discount, "%", sep = "")
print("You must pay $", payment, sep = "")
```

14. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"
```

```
VAT = 0.19

inp = input("Enter total amount: ")

if not re.match(IS_NUMERIC, inp):
    print("Entered value contains non-numeric characters")
else:
    amount = float(inp)
    if amount < 0:
        print("Entered non-negative value")
    else:
        if amount < 50:
            discount = 0
        elif amount < 100:
            discount = 1
        elif amount < 250:
            discount = 2
        else:
            discount = 3

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

        print("You got a discount of ", discount, "%", sep ="")
        print("You must pay $", payment, sep = "")
```

15. Solution

```
a = int(input("Enter age: "))
if a < 18:
    print("Invalid age")
else:
    w = int(input("Enter weight in pounds: "))
    h = int(input("Enter height in inches: "))

    bmi = w * 703 / h ** 2

    if bmi < 15:
        print("Very severely underweight")
    elif bmi < 16:
        print("Severely underweight")
    elif bmi < 18.5:
        print("Underweight")
    elif bmi < 25:
        print("Normal")
    elif bmi < 30:
        print("Overweight")
    elif bmi < 35:
        print("Severely overweight")
    else:
        print("Very severely overweight")
```

16. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"
TAX_RATE = 0.10

inp = input("Enter water consumption (in cubic feet): ")

if not re.match(IS_NUMERIC, inp):
    print("Entered value contains non-numeric characters")
else:
    water = int(inp)
    if water < 0:
        print("Entered value is negative")
    else:
        if water <= 10:
            total = water * 3
        elif water <= 20:
            total = 10 * 3 + (water - 10) * 5
        elif 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
        print("Total amount to pay (taxes included):", total)
```

17. Solution

```
income = float(input("Enter taxable income: "))
children = int(input("Enter number of children: "))

if income <= 8000:
    tax = income * 0.10
elif income <= 30000:
    tax = 8000 * 0.10 + (income - 8000) * 0.15
elif 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

print("Tax:", tax)
```

18. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

inp = input("Enter wind speed (in miles/hour): ")
```

```
if not re.match(IS_NUMERIC, inp):
    print("Entered value contains non-numeric characters")
else:
    wind = float(inp)
    if wind < 0:
        print("Entered value is negative")
    else:
        if wind < 1:
            print("Beaufort: 0\n", "Calm", sep = "")
        elif wind < 4:
            print("Beaufort: 1\n", "Light air", sep = "")
        elif wind < 8:
            print("Beaufort: 2\n", "Light breeze", sep = "")
        elif wind < 13:
            print("Beaufort: 3\n", "Gentle breeze", sep = "")
        elif wind < 18:
            print("Beaufort: 4\n", "Moderate breeze", sep = "")
        elif wind < 25:
            print("Beaufort: 5\n", "Fresh breeze", sep = "")
        elif wind < 31:
            print("Beaufort: 6\n", "Strong breeze", sep = "")
        elif wind < 39:
            print("Beaufort: 7\n", "Moderate gale", sep = "")
        elif wind < 47:
            print("Beaufort: 8\n", "Gale", sep = "")
        elif wind < 55:
            print("Beaufort: 9\n", "Strong gale", sep = "")
        elif wind < 64:
            print("Beaufort: 10\n", "Storm", sep = "")
        elif wind < 74:
            print("Beaufort: 11\n", "Violent storm", sep = "")
        else:
            print("Beaufort: 12\n", "Hurricane force", sep = "")

        if wind < 13:
            print("It's Fishing Day!!!")
```

Chapter 23

23.3 Answers of Review Questions: True/False

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

Chapter 24

24.4 Answers of Review Questions: True/False

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

24.5 Answers of Review Questions: Multiple Choice

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

24.6 Answers of Review Exercises

1. Solution

```
i = 3
while True:
    i -= 1
    if i > 0: break
print("The end")
```

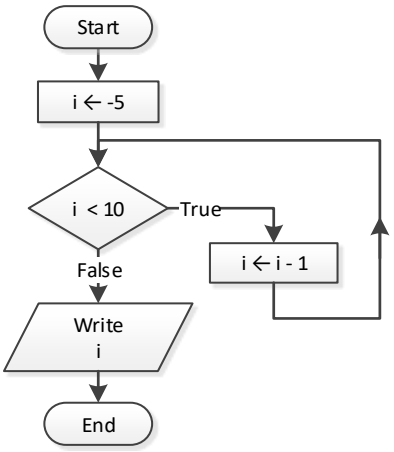
2. Solution

Step	Statement	i	x
1	`i = 3`	**3**	?
2	`x = 0`	3	**0**
3	`while i >= 0:`	True	
4	`i -= 1`	**2**	0
5	`x += i`	2	**2**
6	`while i >= 0:`	True	
7	`i -= 1`	**1**	2
8	`x += i`	1	**3**
9	`while i >= 0:`	True	

10	`i -= 1`	**0**	3
11	`x += i`	0	**3**
12	`while i >= 0:`	True	
13	`i -= 1`	**-1**	3
14	`x += i`	-1	**2**
15	`while i >= 0:`	False	
16	`print(x)`	2 is displayed	

It performs 4 iterations

3. Solution



Step	Statement	Notes	i
1	`i = -5`		**-5**
2	`while i < 10:`	True	
3	`i -= 1`		**-6**
4	`while i < 10:`	True	
5	`i -= 1`		**-7**
6	`while i < 10:`	True	
7	`i -= 1`		**-8**
8	…	**…**	**…**
9	…	**…**	**…**

It performs an infinite number of iterations

4. Solution

Step	Statement	a	b	c	d
1	`a = 2`	**2**	?	?	?
2	`while a <= 10:`	True			

3	`b = a + 1`	2	**3**	?	?
4	`c = b * 2`	2	3	**6**	?
5	`d = c - b + 1`	2	3	6	**4**
6	`if d == 4:`	True			
7	`print(b, ",", c)`	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	`if d == 4:`	False			
14	`elif d == 5:`	False			
15	`elif d == 8:`	True			
16	`print(a, ",", b)`	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	`if d == 4:`	False			
23	`elif d == 5:`	False			
24	`elif d == 8:`	False			
25	`print(a, ",", b, ",", d)`	10, 11, 12 is displayed			
26	`a += 4`	**14**	11	22	12
27	`while a <= 10:`	False			

5. Solution

Step	Statement	a	b	c	d	x
1	`a = 1`	**1**	?	?	?	?
2	`b = 1`	1	**1**	?	?	?
3	`c = 0`	1	1	**0**	?	?
4	`d = 0`	1	1	0	**0**	?
5	`while b < 2:`	True				
6	`x = a + b`	1	1	0	0	**2**
7	`if x % 2 != 0:`	False				
8	`d = d + 1`	1	1	0	**1**	2

9	`a = b`	**1**	1	0	1	2
10	`b = c`	1	**0**	0	1	2
11	`c = d`	1	0	**1**	1	2
12	`while b < 2:`	True				
13	`x = a + b`	1	0	1	1	**1**
14	`if x % 2 != 0:`	True				
15	`c = c + 1`	1	0	**2**	1	1
16	`a = b`	**0**	0	2	1	1
17	`b = c`	0	**2**	2	1	1
18	`c = d`	0	2	**1**	1	1
19	`while b < 2:`	False				

6. Solution

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

7. Solution

Step	Statement	x	y
1	`y = 5`	?	**5**
2	`x = 38`	**38**	5
3	`y *= 2`	38	**10**
4	`x += 1`	**39**	10
5	`print(y)`	10 is displayed	
6	`if y >= x: break`	False	
7	`y *= 2`	39	**20**
8	`x += 1`	**40**	20
9	`print(y)`	20 is displayed	
10	`if y >= x: break`	False	
11	`y *= 2`	40	**40**
12	`x += 1`	**41**	40
13	`print(y)`	40 is displayed	
14	`if y >= x: break`	False	
15	`y *= 2`	41	**80**

16	`x += 1`	**42**	80
17	`print(y)`	80 is displayed	
18	`if y >= x: break`	True	

8. Solution

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

9. Solution

Step	Statement	x	y
1	`y = 2`	?	**2**
2	`x = 0`	**0**	2
3	`y = y ** 2`	0	**4**
4	`if x < 256:`	True	
5	`x = x + y`	**4**	
6	`print(x, ",", y)`	4, 4 is displayed	

7	`if y >= 65535: break`	False	
8	`y = y ** 2`	4	**16**
9	`if x < 256:`	True	
10	`x = x + y`	**20**	16
11	`print(x, ",", y)`	20, 16 is displayed	
12	`if y >= 65535: break`	False	
13	`y = y ** 2`	20	**256**
14	`if x < 256:`	True	
15	`x = x + y`	**276**	256
16	`print(x, ",", y)`	276, 256 is displayed	
17	`if y >= 65535: break`	False	
18	`y = y ** 2`	276	**65536**
19	`if x < 256:`	False	
20	`print(x, ",", y)`	276, 65536 is displayed	
21	`if y >= 65535: break`	True	

10. Solution

Step	Statement	a	b	c	d	x
1	`a = 2`	**2**	?	?	?	?
2	`b = 4`	2	**4**	?	?	?
3	`c = 0`	2	4	**0**	?	?
4	`d = 0`	2	4	0	**0**	?
5	`x = a + b`	2	4	0	0	**6**
6	`if x % 2 != 0:`	False				
7	`elif 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	`if c >= 11: break`	False				
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	`if c >= 11: break`			False		
18	`x = a + b`	5	5	5	5	**10**
19	`if x % 2 != 0:`			False		
20	`elif 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	`if c >= 11: break`			False		
25	`x = a + b`	5	5	8	5	**10**
26	`if x % 2 != 0:`			False		
27	`elif d % 2 == 0:`			False		
28	`c = c + 3`	5	5	**11**	5	10
29	`a = b`	**5**	5	11	5	10
30	`b = d`	5	**5**	11	5	10
31	`if c >= 11: break`			True		

11. Solution

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

12. Solution

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

13. Solution

```
n = int(input())
total = 0

i = 1
while i <= n:
```

```
    a = float(input())
    total = total + a
    i += 1

print(total)
if n > 0:
    print(total / n)
```

14. Solution

```
n = int(input())
p = 1

i = 1
while i <= n:
    a = int(input())
    if a % 2 == 0:
        p = p * a
    i += 1
print(p)
```

15. Solution

```
total = 0

i = 1
while i <= 100:
    a = int(input())
    if a % 10 == 0:
        total = total + a
    i += 1
print(total)
```

16. Solution

```
total = 0

i = 1
while i <= 20:
    a = int(input())
    if 100 <= a <= 999:
        total = total + a
    i += 1
print(total)
```

17. Solution

```
p = 1

a = float(input())
while a != 0:
    p = p * a
    a = float(input())
```

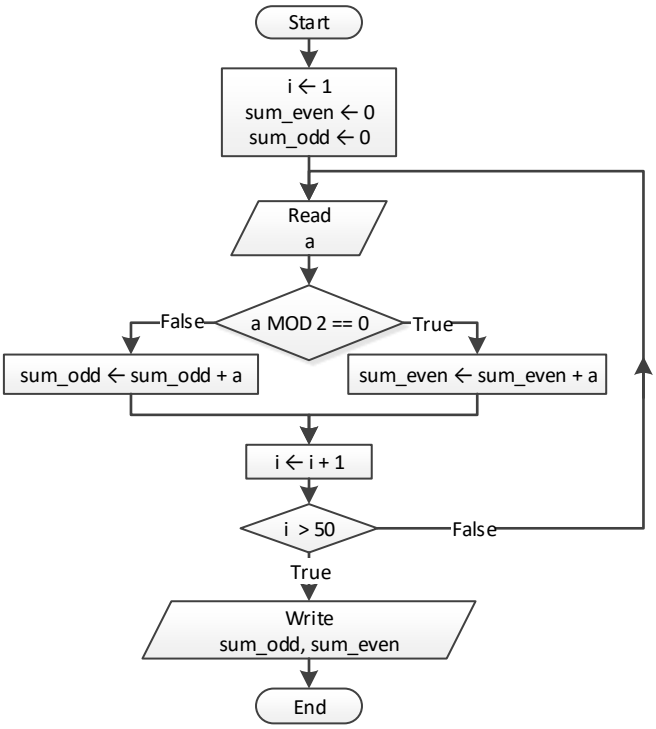
```
print(p)
```

Step	Statement	a	p
1	`p = 1`	?	**1**
2	`a = float(input())`	**3**	1
3	`while a != 0:`	True	
4	`p = p * a`	3	**3**
5	`a = float(input())`	**2**	3
6	`while a != 0:`	True	
7	`p = p * a`	2	**6**
8	`a = float(input())`	**9**	6
9	`while a != 0:`	True	
10	`p = p * a`	9	**54**
11	`a = float(input())`	**0**	54
12	`while a != 0:`	False	
13	`print(p)`	54 is displayed	

18. Solution

```
population = 30000

years = 0
while population <= 100000:
    population += population * 0.03
    years += 1

print(years)
```

19. Solution




```
i = 1
sum_even = 0
sum_odd = 0
while True:
    a = int(input())
    if a % 2 == 0:
        sum_even += a
    else:
        sum_odd += a
    i += 1
    if i > 50: break
print(sum_odd, sum_even)
```

20. Solution

```
n = int(input())
i = 1
p = 1
while True:
    a = int(input())
    if a < 0:
        p *= a
    i += 1
    if i > n: break
print(abs(p))
```

21. Solution

```
i = 1
p = 1
while True:
    a = int(input("Enter an integer: "))
    if 500 <= a <= 599:
        p *= a
    i += 1
    if i > 5: break

print(p)
```

22. Solution

```
population = 50000

years = 0
while population >= 20000:
    population -= population * 0.10
    years += 1

print(years)
```

Chapter 25

25.3 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

25.4 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

25.5 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	`if j < 5:`		True	
5	`b += 1`	0	**1**	0
6	`j = 2`	0	1	**2**
7	`if j < 5:`		True	
8	`b += 1`	0	**2**	2
9	`j = 4`	0	2	**4**
10	`if j < 5:`		True	
11	`b += 1`	0	**3**	4
12	`j = 6`	0	3	**6**
13	`if j < 5:`		False	
14	`a += j - 1`	**5**	3	6
15	`j = 8`	5	3	**8**
16	`if j < 5:`		False	
17	`a += j - 1`	**12**	3	8

18	`print(a, ",", b)`	12, 3 is displayed

2. Solution

For input value of 10

Step	Statement	a	b	j
1	`a = int(input())`	**10**	?	?
2	`b = a`	10	**10**	?
3	`j = a - 5`	10	10	**5**
4	`if j % 2 != 0:`		True	
5	`b = a + j + 5`	10	**20**	5
6	`j = 7`	10	20	**7**
7	`if j % 2 != 0:`		True	
8	`b = a + j + 5`	10	**22**	7
9	`j = 9`	10	22	**9**
10	`if j % 2 != 0:`		True	
11	`b = a + j + 5`	10	**24**	9
12	`print(b)`	24 is displayed		

For input value of 21

Step	Statement	a	b	j
1	`a = int(input())`	**21**	?	?
2	`b = a`	21	**21**	?
3	`j = a - 5`	21	21	**16**
4	`if j % 2 != 0:`		False	
5	`b = a - j`	21	**5**	16
6	`j = 18`	21	5	**18**
7	`if j % 2 != 0:`		False	
8	`b = a - j`	21	**3**	18
9	`j = 20`	21	3	**20**
10	`if j % 2 != 0:`		False	
11	`b = a - j`	21	**1**	20
12	`print(b)`	1 is displayed		

3. Solution

For input value of 12

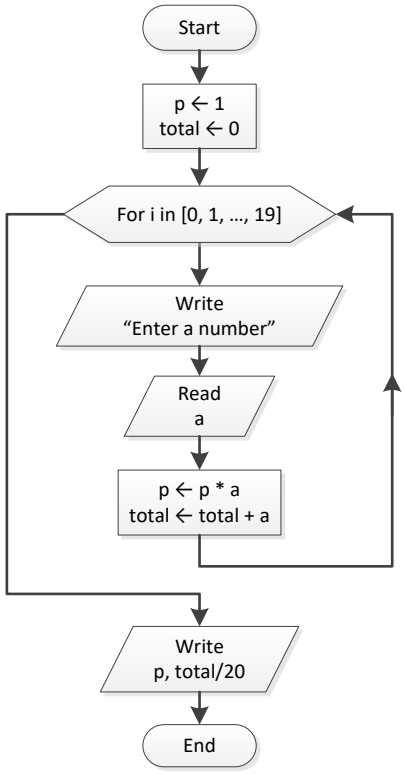
Step	Statement	a	x	y	j
1	`a = int(input())`	**12**	?	?	?

2	`j = 2`	12	?	?	**2**
3	`x = j * 3 + 3`	12	**9**	?	2
4	`y = j * 2 + 10`	12	9	**14**	2
5	`if y - x > 0 or x > 30:`	True			
6	`y *= 2`	12	9	**28**	2
7	`x += 4`	12	**13**	28	2
8	`print(x, ",", y)`	13, 28 is displayed			
9	`j = 5`	12	13	28	**5**
10	`x = j * 3 + 3`	12	**18**	28	5
11	`y = j * 2 + 10`	12	18	**20**	5
12	`if y - x > 0 or x > 30:`	True			
13	`y *= 2`	12	18	**40**	5
14	`x += 4`	12	**22**	40	5
15	`print(x, ",", y)`	22, 40 is displayed			
16	`j = 8`	12	22	40	**8**
17	`x = j * 3 + 3`	12	**27**	40	8
18	`y = j * 2 + 10`	12	27	**26**	8
19	`if y - x > 0 or x > 30:`	False			
20	`x += 4`	12	**31**	26	8
21	`print(x, ",", y)`	31, 26 is displayed			
22	`j = 11`	12	31	26	**11**
23	`x = j * 3 + 3`	12	**36**	26	11
24	`y = j * 2 + 10`	12	36	**32**	11
25	`if y - x > 0 or x > 30:`	True			
26	`y *= 2`	12	36	**64**	11
27	`x += 4`	12	**40**	64	11
28	`print(x, ",", y)`	40, 64 is displayed			

4. Solution

i. 9
ii. 20
iii. -7
iv. -1

5. Solution

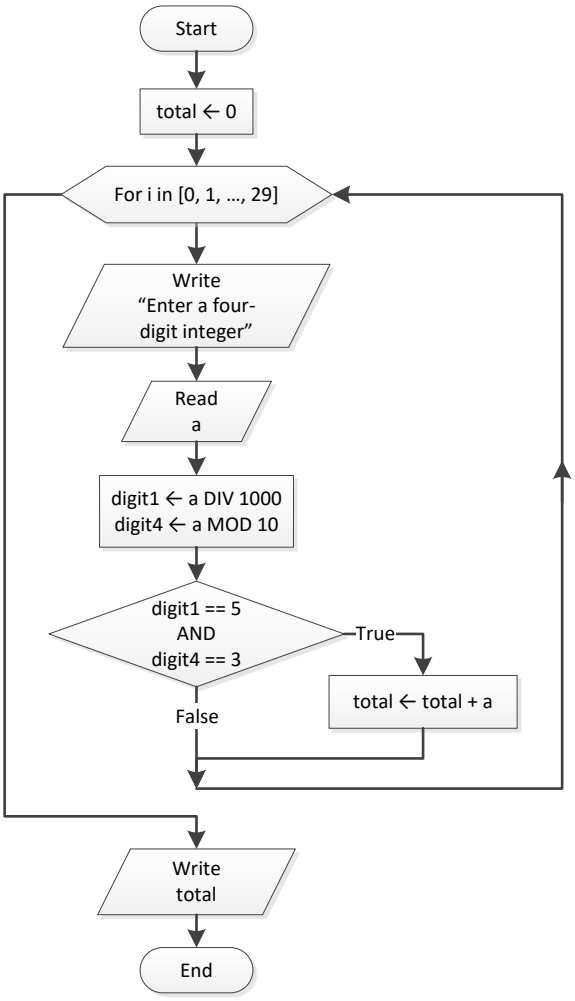



```
p = 1
total = 0
for i in range(20):
    a = float(input("Enter a number: "))
    p = p * a
    total = total + a
print(p, total / 20)
```

6. Solution

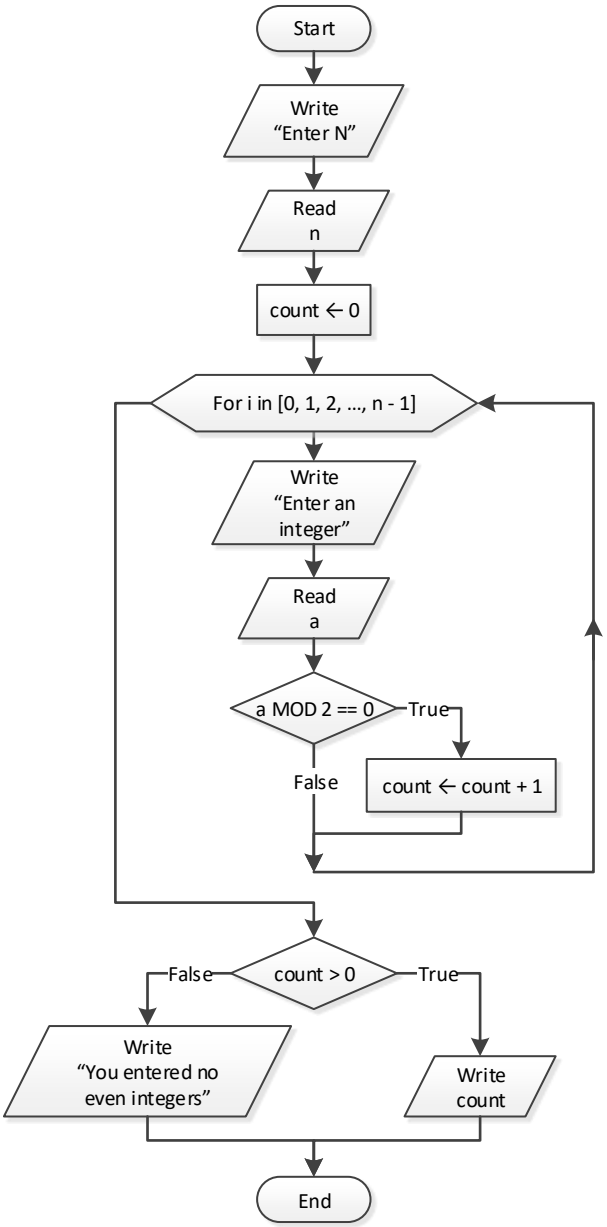
```
import math

deg = int(input("Enter degrees: "))
for i in range(0, deg * 10 + 5, 5):
    print(math.sin((i / 10) * math.pi / 180))
```

7. Solution



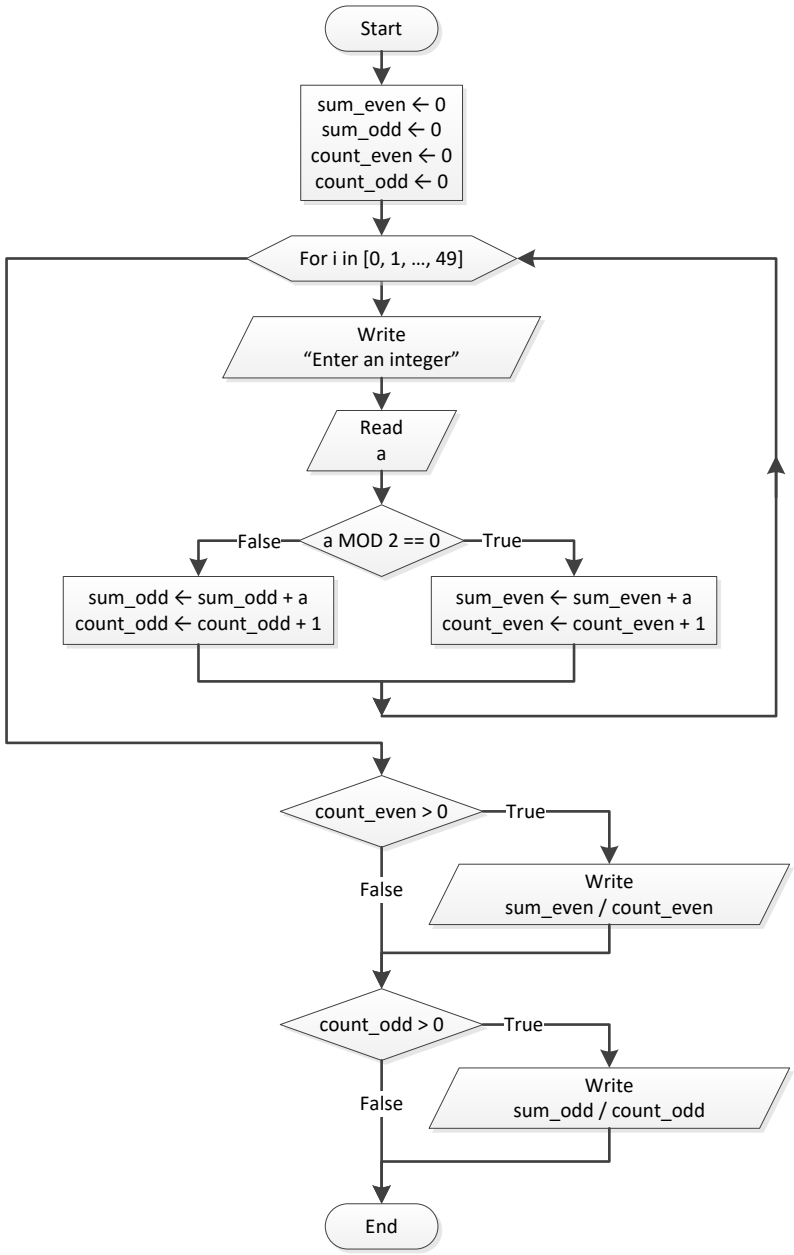

```
total = 0
for i in range(30):
    a = int(input("Enter a four-digit integer: "))
    digit1 = a // 1000
    digit4 = a % 10
    if digit1 == 5 and digit4 == 3:
        total += a

print(total)
```

8. Solution




```
n = int(input("Enter N: "))
count = 0
for i in range(n):
    a = int(input("Enter an integer: "))
    if a % 2 == 0:
        count += 1

if count > 0:
    print(count)
else:
    print("You entered no even integers")
```

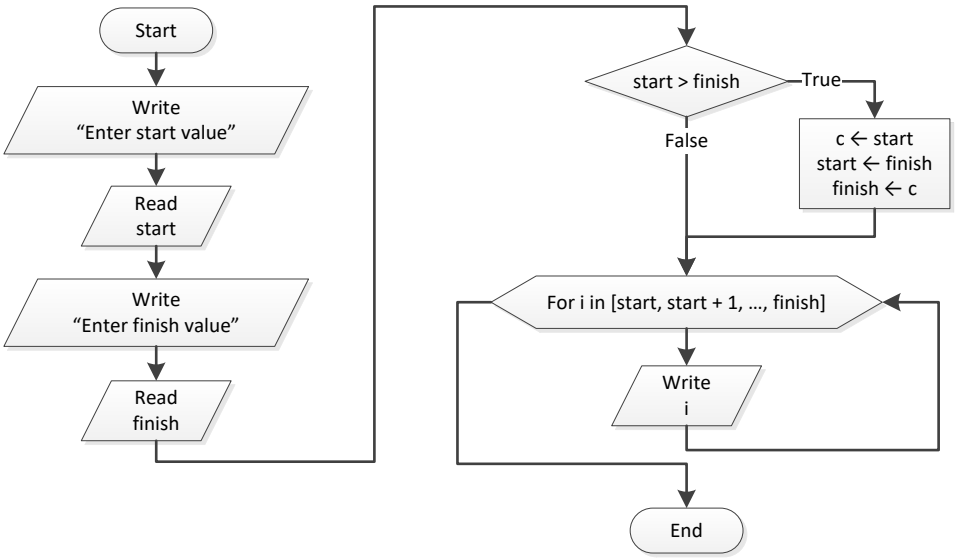
9. Solution




```
sum_even = 0
sum_odd = 0
count_even = 0
count_odd = 0
for i in range(50):
    a = int(input("Enter an integer: "))
    if a % 2 == 0:
        sum_even += a
        count_even += 1
    else:
        sum_odd += a
        count_odd += 1
```

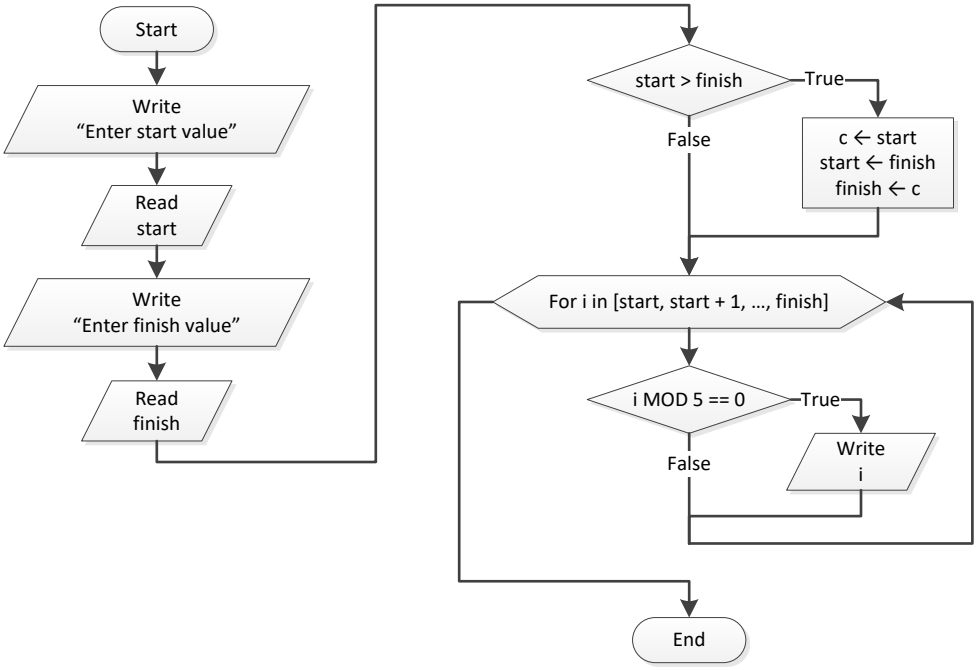
```
if count_even > 0:
    print(sum_even / count_even)

if count_odd > 0:
    print(sum_odd / count_odd)
```

10. Solution




```
start = int(input("Enter start value: "))
finish = int(input("Enter finish value: "))

if start > finish:          # Or you can do the following:
    c = start               # start, finish = finish, start
    start = finish
    finish = c

for i in range(start, finish + 1):
    print(i)
```

11. Solution




```
start = int(input("Enter start value: "))
finish = int(input("Enter finish value: "))

if start > finish:
    c = start
    start = finish
    finish = c

for i in range(start, finish + 1):
    if i % 5 == 0:
        print(i)
```

12. Solution

First Approach

```
base = float(input("Enter a value for base: "))
exp = int(input("Enter an integer for exponent: "))

p = 1
if exp >= 0:
    for i in range(exp):
        p *= base
else:
    for i in range(-exp):
        p *= 1 / base

print(p)
```

Second Approach

```
base = float(input("Enter a value for base: "))
exp = int(input("Enter an integer for exponent: "))

p = 1
for i in range(abs(exp)):
    p *= base

if exp < 0:
    p = 1 / p

print(p)
```

13. Solution

```
msg = input("Enter a message: ")

characters = len(msg)
count = 0
for i in range(characters):
    if msg[i] == " ":
        count += 1

words = count + 1

print("The message entered contains", words, "words")
```

14. Solution

```
msg = input("Enter a message: ")

characters = len(msg)
count = 0
for i in range(characters):
    if msg[i] == " ":
        count += 1

words = count + 1

print("The average number of letters in each word is", (characters - count) / words)
```

Chapter 26

26.3 Answers of Review Questions: True/False

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

26.4 Answers of Review Questions: Multiple Choice

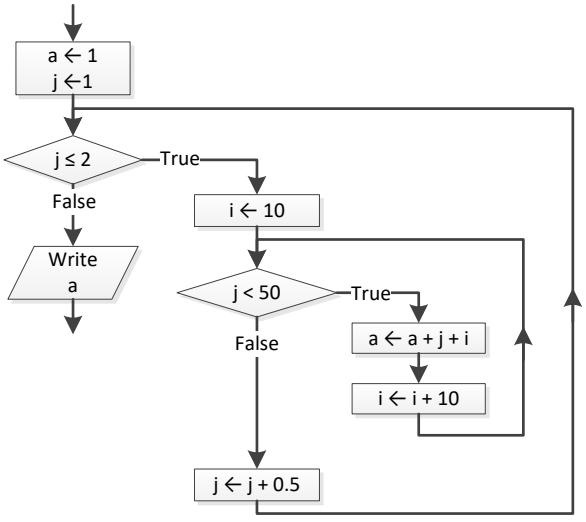
1. b
2. a
3. c
4. a
5. b

26.5 Answers of Review Exercises

1. Solution

i. 10
ii. 50
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	print(a)	100 is displayed		

3. Solution

Step	Statement	s	i	j
1	s = 0	**0**	?	?
2	i = 1	0	**1**	?
3	j = 3	0	1	**3**
4	s = s + i * j	**3**	1	3

5	`j = 2`	3	1	**2**
6	`s = s + i * j`	**5**	1	2
7	`j = 1`	5	1	**1**
8	`s = s + i * j`	**6**	1	1
9	`i = 2`	6	**2**	1
10	`j = 3`	6	2	**3**
11	`s = s + i * j`	**12**	2	3
12	`j = 2`	12	2	**2**
13	`s = s + i * j`	**16**	2	2
14	`i = 3`	16	**3**	2
15	`j = 3`	16	3	**3**
16	`s = s + i * j`	**25**	3	3
17	`i = 4`	25	**4**	3
18	`print(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	`s = s + y`	**26**	25	1	?
5	`y -= 5`	26	**20**	1	?
6	`i = 2`	26	20	**2**	?
7	`s = s + y`	**46**	20	2	?
8	`y -= 5`	46	**15**	2	?
9	`i = 3`	46	15	**3**	?
10	`s = s + y`	**61**	15	3	?
11	`y -= 5`	61	**10**	3	?
12	`ans = input()`	61	10	3	**"NO"**
13	`if ans != "YES": break`	True			
14	`print(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	`s = s + y`	**26**	25	1	?
5	`y -= 5`	26	**20**	1	?
6	`i = 2`	26	20	**2**	?
7	`s = s + y`	**46**	20	2	?
8	`y -= 5`	46	**15**	2	?
9	`i = 3`	46	15	**3**	?
10	`s = s + y`	**61**	15	3	?
11	`y -= 5`	61	**10**	3	?
12	`ans = input()`	61	10	3	**“YES”**
13	`if ans != "YES": break`	False			
14	`i = 1`	61	10	**1**	“YES”
15	`s = s + y`	**71**	10	1	“YES”
16	`y -= 5`	71	**5**	1	“YES”
17	`i = 2`	71	5	**2**	“YES”
18	`s = s + y`	**76**	5	2	“YES”
19	`y -= 5`	76	**0**	2	“YES”
20	`i = 3`	76	0	**3**	“YES”
21	`s = s + y`	**76**	0	3	“YES”
22	`y -= 5`	76	**-5**	3	“YES”
23	`ans = input()`	76	-5	3	**“NO”**
24	`if ans != "YES": break`	True			
25	`print(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	`s = s + y`	**26**	25	1	?
5	`y -= 5`	26	**20**	1	?
6	`i = 2`	26	20	**2**	?
7	`s = s + y`	**46**	20	2	?

8	`y -= 5`	46	**15**	2	?
9	`i = 3`	46	15	**3**	?
10	`s = s + y`	**61**	15	3	?
11	`y -= 5`	61	**10**	3	?
12	`ans = input()`	61	10	3	**"YES"**
13	`if ans != "YES": break`	False			
14	`i = 1`	61	10	**1**	"YES"
15	`s = s + y`	**71**	10	1	"YES"
16	`y -= 5`	71	**5**	1	"YES"
17	`i = 2`	71	5	**2**	"YES"
18	`s = s + y`	**76**	5	2	"YES"
19	`y -= 5`	76	**0**	2	"YES"
20	`i = 3`	76	0	**3**	"YES"
21	`s = s + y`	**76**	0	3	"YES"
22	`y -= 5`	76	**-5**	3	"YES"
23	`ans = input()`	76	-5	3	**"YES"**
24	`if ans != "YES": break`	False			
25	`i = 1`	76	-5	**1**	"YES"
26	`s = s + y`	**71**	-5	1	"YES"
27	`y -= 5`	71	**-10**	1	"YES"
28	`i = 2`	71	-10	**2**	"YES"
29	`s = s + y`	**61**	-10	2	"YES"
30	`y -= 5`	61	**-15**	2	"YES"
31	`i = 3`	61	-15	**3**	"YES"
32	`s = s + y`	**46**	-15	3	"YES"
33	`y -= 5`	46	**-20**	3	"YES"
34	`ans = input()`	46	-20	3	**"NO"**
35	`if ans != "YES": break`	True			
36	`print(s)`	46 is displayed			

5. Solution

```
for hour in range(24):
    for minutes in range(60):
        print(hour, "\t", minutes)
```

6. Solution

```
for i in range(5, 0, -1):
```

```
    for j in range(i):
        print(i, end = "")
    print()
```

7. Solution

```
for i in range(6):
    for j in range(i + 1):
        print(j, end = "")
    print()
```

8. Solution

```
for i in range(4):
    for j in range(10):
        print("* ", end = "")
    print()
```

9. Solution

```
n = int(input("Enter an integer between 3 and 20: "))

for i in range(n):
    for j in range(n):
        print("* ", end = "")
    print()
```

10. Solution

```
n = int(input("Enter an integer between 3 and 20: "))

for j in range(n):
    print("* ", end = "")
print()

for i in range(n - 2):
    print("* ", end = "")
    for j in range(n - 2):
        print("  ", end = "")
    print("* ")

for j in range(n):
    print("* ", end = "")
```

11. Solution

```
for i in range(1, 5):
    for j in range(i):
        print("* ", end = "")
    print()

for i in range(5, 0, -1):
    for j in range(i):
```

```
        print("* ", end = "")
    print()
```

Chapter 27

27.10 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

27.11 Answers of Review Questions: Multiple Choice

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

27.12 Answers of Review Exercises

1. Solution

```
s = 0
for i in range(100):
    number = float(input())
    s = s + number
average = s / 100
print(average)
```

2. Solution

```
s = 0

denom = 1
for j in range(1, 101):
    denom *= j

for i in range(1, 101):
    s += i / denom
print(s)
```

3. Solution

```
import math
s = 10
i = 1
while i <= 10:
    s += math.sqrt(i)
```

```
    i += 1
print(s)
```

4. Solution

```
start = int(input())
finish = int(input())
i = start
while i <= finish:
    print(i)
    i += 1
```

5. Solution

```
start = int(input())
x = 1
i = start
while i <= start * 2:
    x = x ** 1.1 + i
    i += 1
print(x)
```

6. Solution

```
import math
x = 42
i = 0
while i <= 99:
    x = math.sqrt(x) + i
    print(x)
    i += 1
```

7. Solution

```
import math
s = 0
for i in range(100, 0, -5):
    s = s + math.sqrt(i)
print(s)
```

8. Solution

```
import math
s = 0
y = 0
for i in range(1, 21, 3):
    s = s + math.sqrt(y + i + 3)
    y = y + (i + 3) * 2
print(s)
```

9. Solution

```
y = 0
for i in range(1, 11, 2):
    a = float(input())
    a += i
    y = y + (a + i + 2) ** 3
print(y)
```

10. Solution

This conversion cannot be carried out in Python.

11. Solution

This conversion cannot be carried out in Python.

12. Solution

```
x = 0
for y in range(-10, 10):
    x = x + 2 ** y
print(x)
```

13. Solution

This conversion cannot be carried out in Python.

14. Solution

```
for i in range(1, 5):
    for j in range(1, 5):
        print(i, "x", j, "=", (i * j))
```

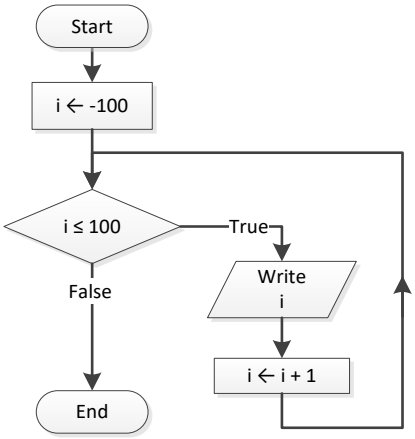
15. Solution

```
print("\t|\t", end = "")
for i in range(1, 13):
    print(i, "\t", end = "")
print()

for i in range(1, 13):
    print("---------", end = "")
print()

for i in range(1, 13):
    print(i, "\t|\t", end = "")
    for j in range(1, 13):
        print(i * j, end = "\t")
    print()
```

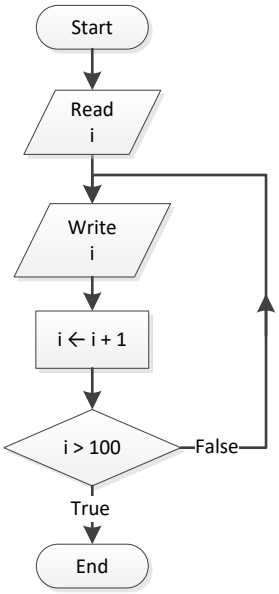
Chapter 28

28.4 Answers of Review Exercises

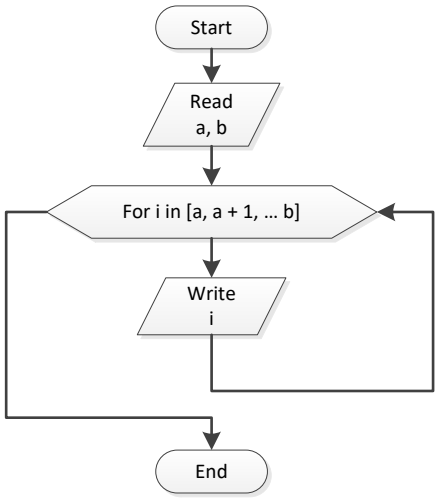
1. Solution



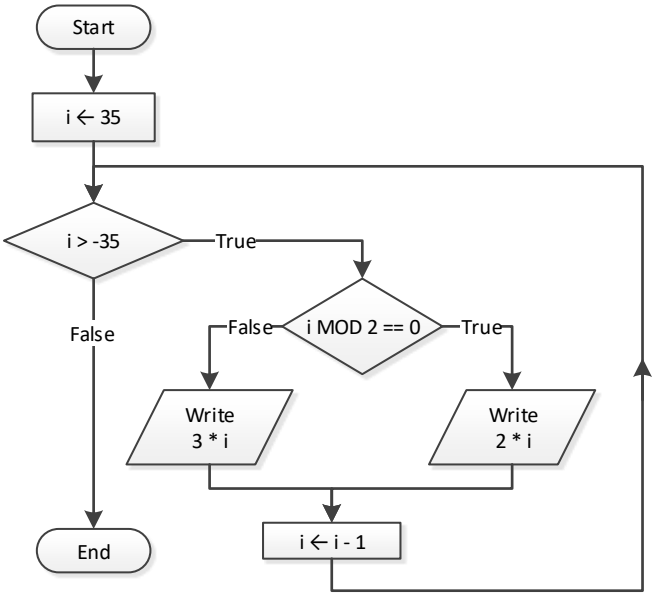
2. Solution



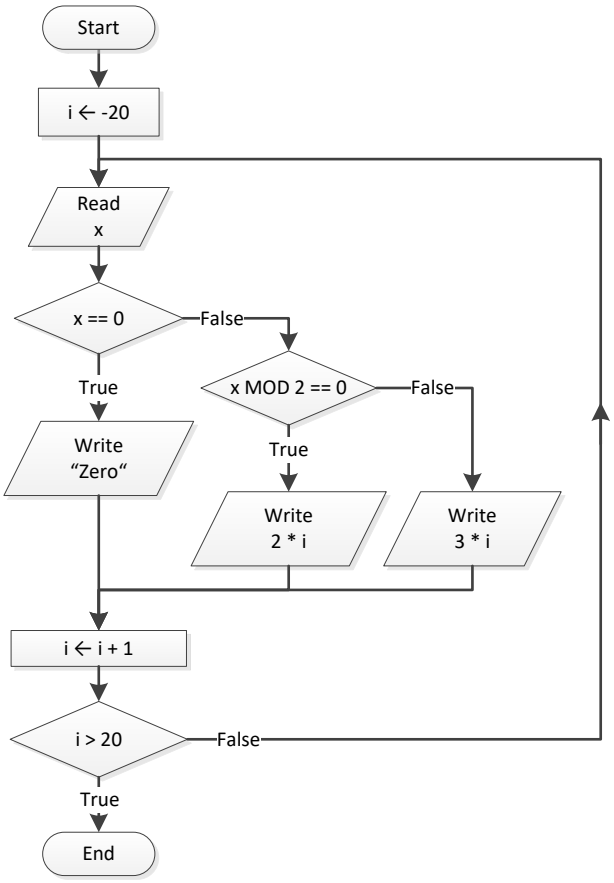
3. Solution



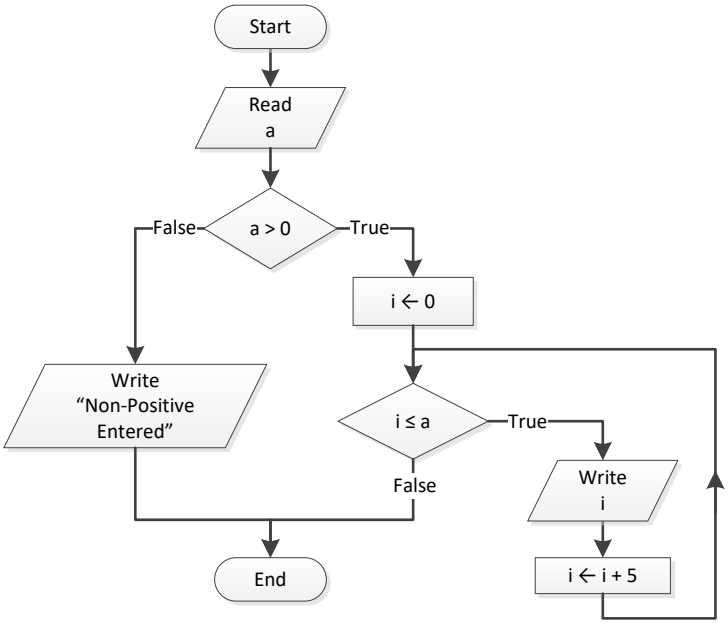
4. Solution



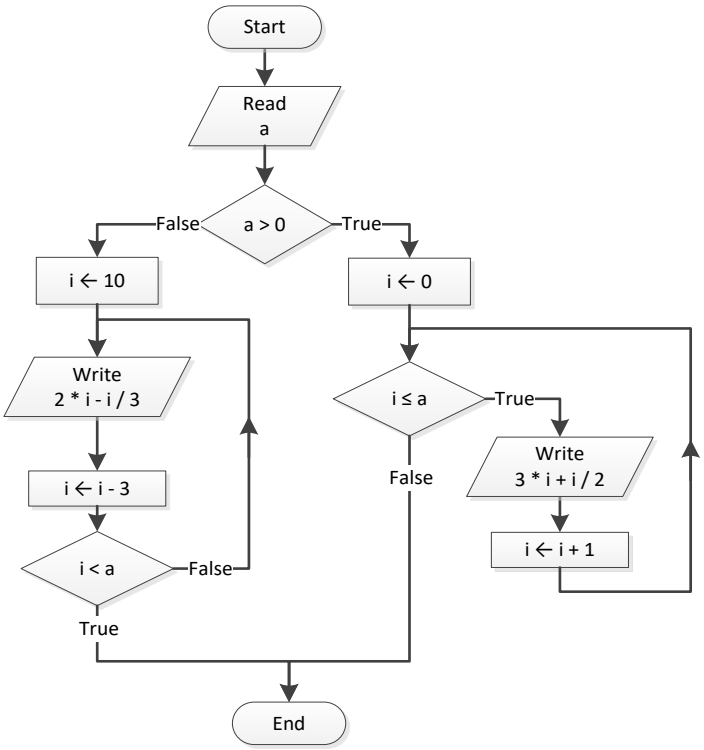
5. Solution



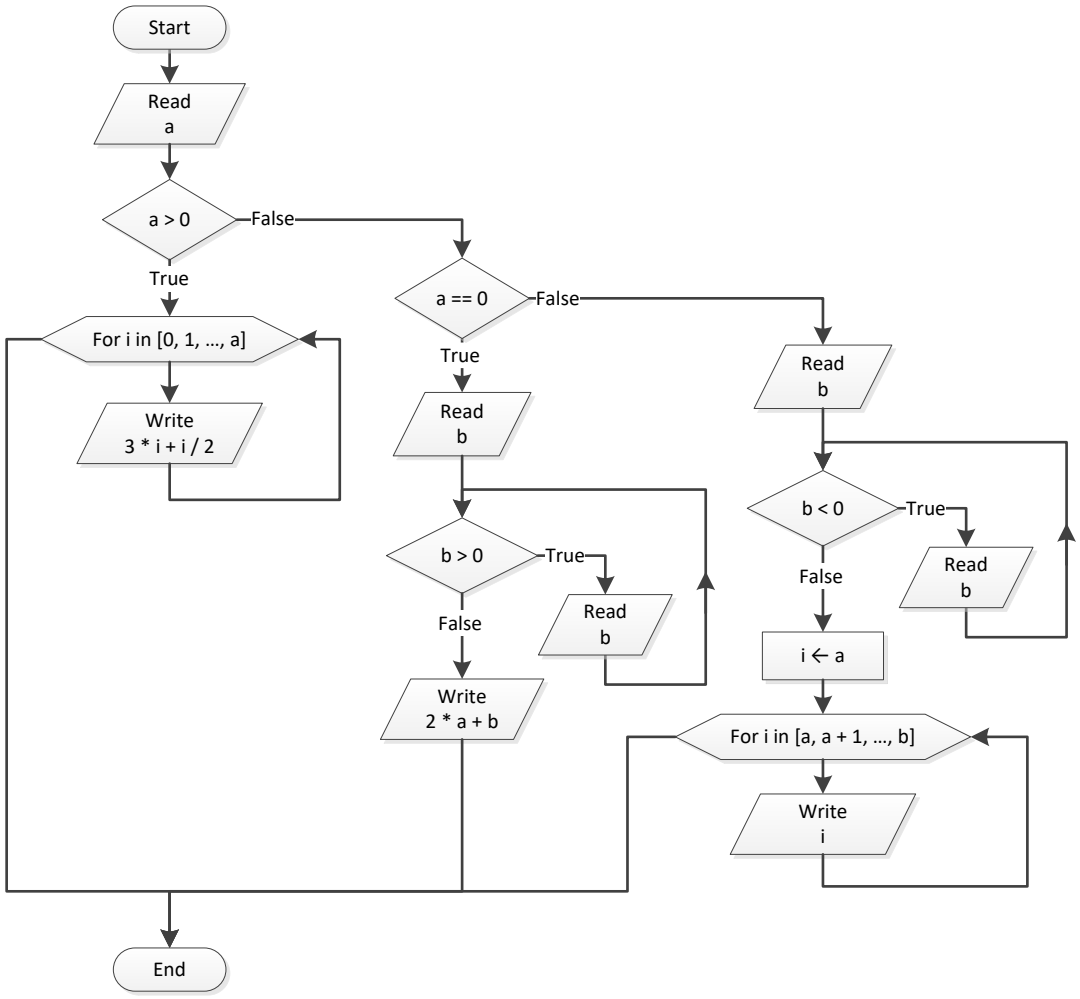
6. Solution



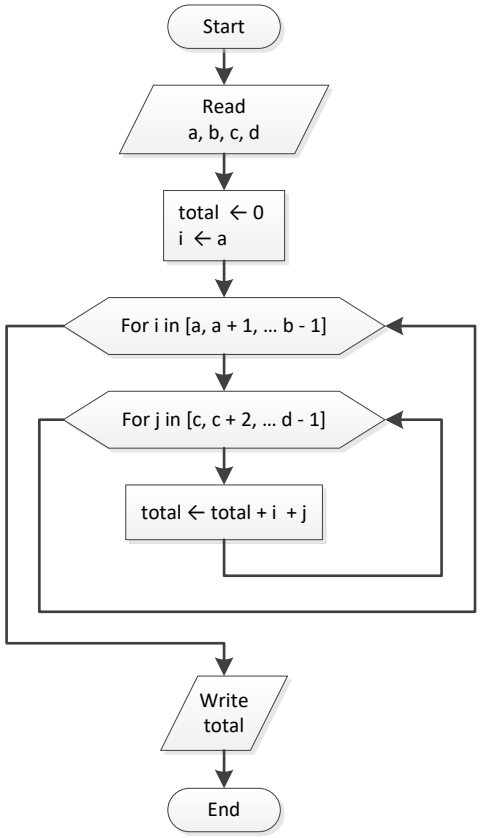
7. Solution


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

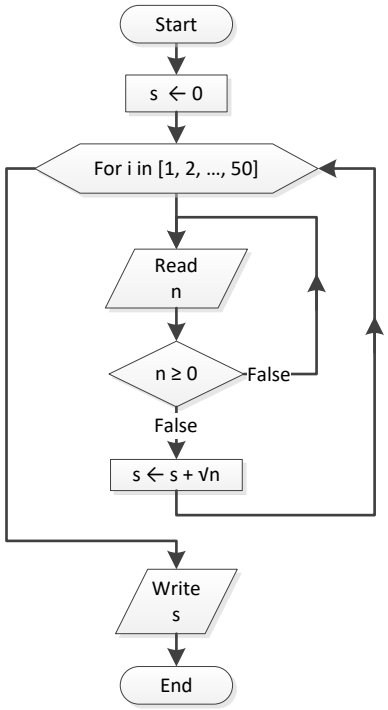
8. Solution


Start
Read a
a > 0
False
True
For i in [0, 1, ..., a]
Write 3 * i + i / 2
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
For i in [a, a + 1, ..., b]
Write i
End

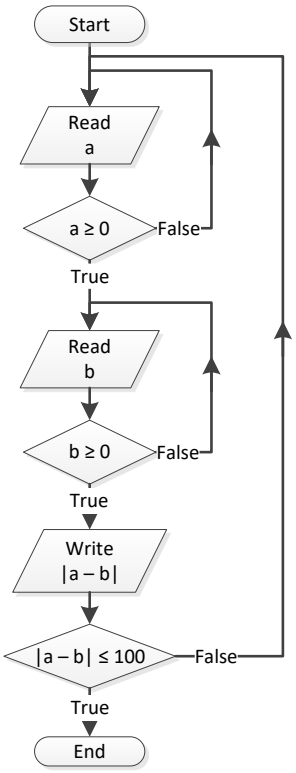
9. Solution



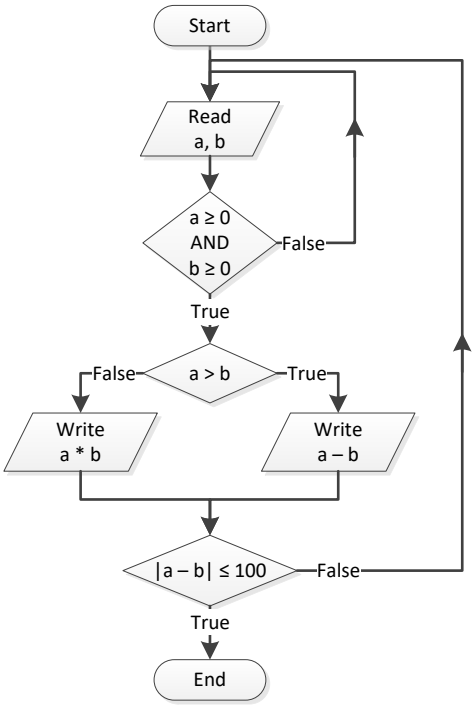
10. Solution



11. Solution



12. Solution



13. Solution

```
i = 1
while True:
    print(i)
    i += 5
    if i > 500: break
print("The End")
```

14. Solution

```
i = 0
a = int(input())
while True:
    if i % 2 != 0:
        print(i)
    i += 5
    if i >= a: break
```

15. Solution

```
a = int(input())
while a != -1:
    while True:
        b = int(input())
        if b > a: break
    i = a
    while i <= b:
        print(i)
        i += 1
    a = int(input())
```

16. Solution

```
i = 1
S = 0
P = 1
a = 0
while True:
    if i < 45:
        S += a
    else:
        P *= a
    i += 1
    if i >= 90: break
    a = float(input())

print(S, P)
```

Chapter 29

29.7 Answers of Review Questions: True/False

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

29.8 Answers of Review Exercises

1. Solution

```
s = 0
for i in range(1, 101, 2):
    s += i
print(s)
```

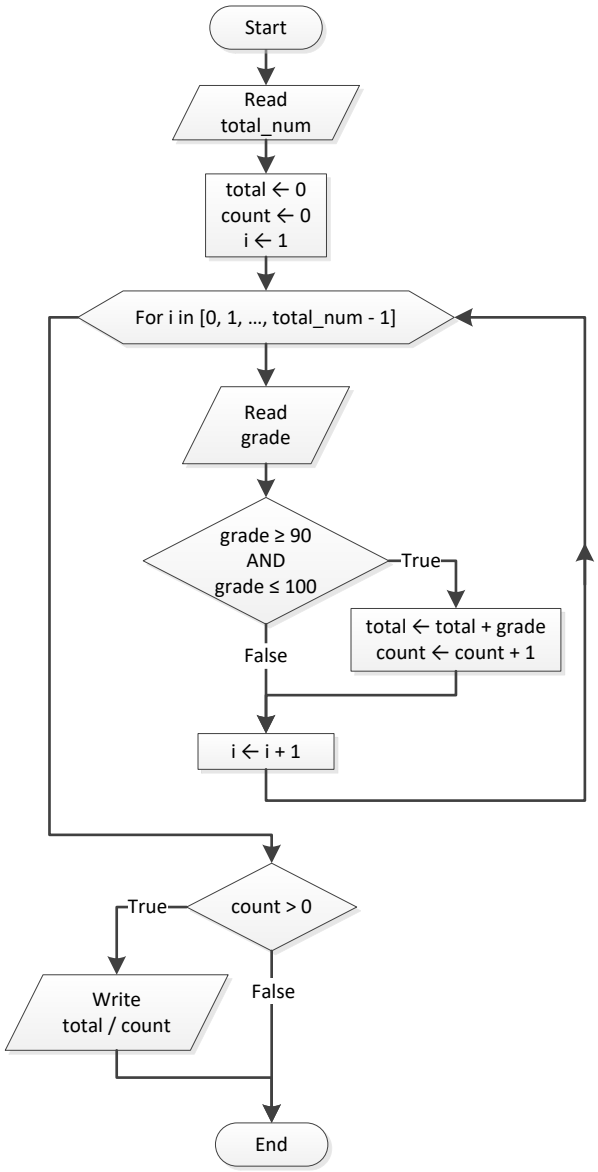
2. Solution

```
n = int(input())
p = 1
for i in range(2, 2 * n + 2, 2):
    p *= i ** (i - 1)
print(p)
```

3. Solution

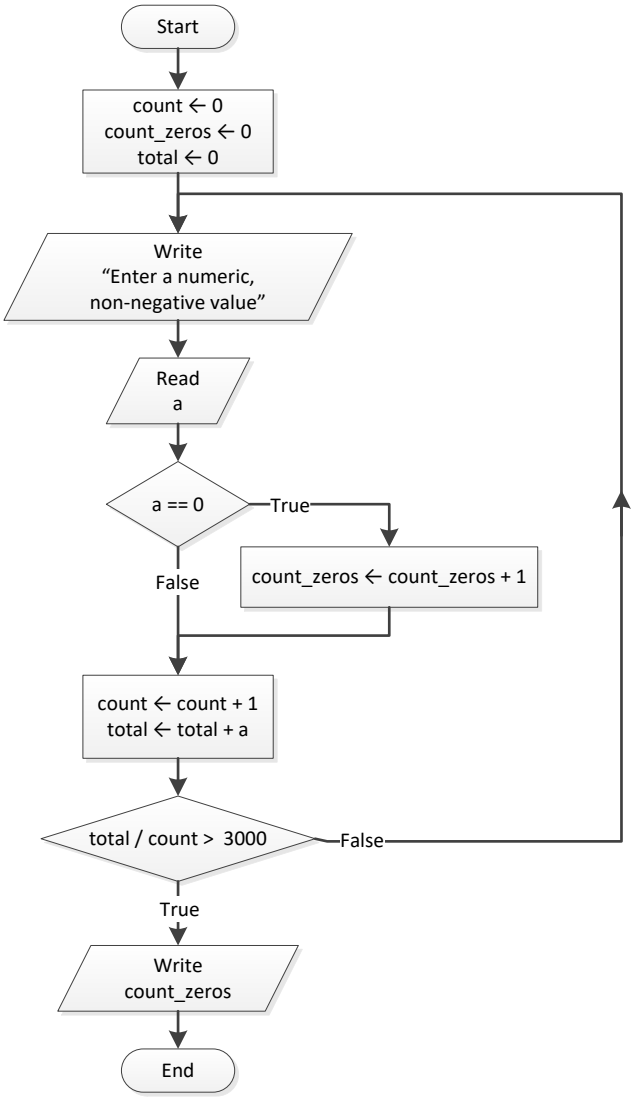
```
s = 0
i = 1
offset = 0
while i <= 191:
    s += i
    offset += 1
    i += offset
print(s)
```

4. Solution




```
total_num = int(input())
total = 0
count = 0
for i in range(total_num):
    grade = int(input())
    if 90 <= grade <= 100:
        total += grade
        count += 1
if count > 0:
    print(total / count)
```

5. Solution




```
count = 0
count_zeros = 0
total = 0

while True:
    a = float(input("Enter a numeric, non-negative value: "))
    if a == 0:
        count_zeros += 1
    count += 1
    total += a
    if total / count > 3000: break

print(count_zeros)
```

6. Solution

First Approach

```
a = int(input("Enter an integer between 1 and 20: "))
for i in range(1000, 10000):
    d4 = i % 10
    r = i // 10
    d3 = r % 10
    r = r // 10
    d2 = r % 10
    d1 = r // 10
    if d1 + d2 + d3 + d4 < a:
        print(i)
```

However, using the `divmod()` function, it can become:

```
a = int(input("Enter an integer between 1 and 20: "))
for i in range(1000, 10000):
    r, d4 = divmod(i, 10)
    r, d3 = divmod(r, 10)
    d1, d2 = divmod(r, 10)

    if d1 + d2 + d3 + d4 < a:
        print(i)
```

Second Approach

```
a = int(input("Enter an integer between 1 and 20: "))
for d1 in range(1, 10):
    for d2 in range(10):
        for d3 in range(10):
            for d4 in range(10):
                if d1 + d2 + d3 + d4 < a:
                    print(d1 * 1000 + d2 * 100 + d3 * 10 + d4)
```

Third Approach

```
a = int(input("Enter an integer between 1 and 20: "))
for i in range(1000, 10000):
    d1, d2, d3, d4 = str(i)                         #d1, d2, d3, d4 are strings
    if int(d1) + int(d2) + int(d3) + int(d4) < a:  #In order to sum the digits, convert to integer
        print(i)
```

7. Solution

First Approach

```
for i in range(1000, 10000):
    d1 = i // 1000
    r = i % 1000
    d2 = r // 100
    r = r % 100
    d3 = r // 10
    d4 = r % 10

    if d1 > d2 and d2 == d3 and d3 < d4:
        print(i)
```

However, using the `divmod()` function, it can become:

```
for i in range(1000, 10000):
    d1, r = divmod(i, 1000)
    d2, r = divmod(r, 100)
    d3, d4 = divmod(r, 10)
    if d1 > d2 and d2 == d3 and d3 < d4:
        print(i)
```

Second Approach

```
for d1 in range(1, 10):
    for d2 in range(10):
        for d3 in range(10):
            for d4 in range(10):
                if d1 > d2 and d2 == d3 and d3 < d4:
                    print(d1 * 1000 + d2 * 100 + d3 * 10 + d4)
```

Third Approach

```
for i in range(1000, 10000):
    d1, d2, d3, d4 = str(i)                    #d1, d2, d3, d4 are strings
    if d1 > d2 and d2 == d3 and d3 < d4:       #No need to convert to integer. Compare them as strings
        print(i)
```

8. Solution

```
inp = input()
while not re.match(IS_NUMERIC, inp) or int(inp) != 1 and int(inp) != 0:
    print("Error")
    inp = input()
x = int(inp)
```

9. Solution

```
while True:
    sex = input()
    if sex.upper() == "M" or sex.upper() == "F": break
```

10. Solution

```
import re
import math
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

inp = input("Enter a non-negative number: ")
count = 1
while not re.match(IS_NUMERIC, inp) or int(inp) < 0:
    print("Error: Invalid number!")
    inp = input("Enter a non-negative number: ")
    if not re.match(IS_NUMERIC, inp) or int(inp) < 0:
        count += 1
        if count == 3: break

if count < 3:
    x = int(inp)
```

```
    y = math.sqrt(x)
    print(y)
else:
    print("Dude, you are dumb!")
```

11. Solution

```
import math

while True:
    r = float(input("Enter the length of a radius: "))
    while r <= 0:
        r = float(input("Invalid radius. Enter the length of a radius: "))

    area = math.pi * r ** 2
    print("The area is:", area)

    answer = input("Would you like to repeat? ")
    if answer.upper() != "YES": break
```

12. Solution

```
for x in range(-100, 101):
    for y in range(-100, 101):
        if 5 * x + 3 * y ** 2 == 0:
            print(x, ",", y)
```

13. Solution

```
for x in range(-10, 11):
    for y in range(-10, 11):
        for z in range(-10, 11):
            if (x + y) / 2 + 3 * z ** 2 / (x + 3 * y + 45) == x / 3:
                print(x, ",", y, ",", z)
```

14. Solution

```
m1 = int(input())
m2 = int(input())
m3 = int(input())

s = 0
while m2 != 0:
    if m2 % 2 != 0:
        s += m1
    m1 *= 2
    m2 = m2 // 2

m1 = s
m2 = m3

s = 0
while m2 != 0:
```

```
    if m2 % 2 != 0:
        s += m1
    m1 *= 2
    m2 = m2 // 2

print(s)
```

15. Solution

```
a = int(input("Enter an integer greater than 1: "))
while a < 2:
    a = int(input("Wrong number. Please enter an integer greater than 1: "))

b = int(input("Enter a second integer greater than 1: "))
while b < 2:
    b = int(input("Wrong number. Please enter a second integer greater than 1: "))

if a > b:
    a, b = b, a

for x in range(a, b + 1):
    number_of_divisors = 2
    i = 2
    while i <= x // 2 and number_of_divisors == 2:
        if x % i == 0:
            number_of_divisors += 1
        i += 1
    if number_of_divisors == 2:
        print("Number", x, "is prime")
```

16. Solution

First Approach

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

inp = input("Enter a four-digit integer: ")
while not re.match(IS_NUMERIC, inp) or int(inp) < 1000 or int(inp) > 9999:
    inp = input("Wrong number. Please enter a four-digit integer: ")
a = int(inp)

inp = input("Enter a second four-digit integer: ")
while not re.match(IS_NUMERIC, inp) or int(inp) < 1000 or int(inp) > 9999:
    inp = input("Wrong number. Please enter a second four-digit integer: ")
b = int(inp)

if a > b:
    c = a           # Or you can do the following:
    a = b           # a, b = b, a
    b = c

for x in range(a, b + 1):
    r, d4 = divmod(x, 10)
```

```
    r, d3 = divmod(r, 10)
    d1, d2 = divmod(r, 10)

    if d1 == d4 and d2 == d3:
        print(x)
```

Second Approach

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

inp = input("Enter a four-digit integer: ")
while not re.match(IS_NUMERIC, inp) or int(inp) < 1000 or int(inp) > 9999:
    inp = input("Wrong number. Please enter a four-digit integer: ")
a = int(inp)

inp = input("Enter a second four-digit integer: ")
while not re.match(IS_NUMERIC, inp) or int(inp) < 1000 or int(inp) > 9999:
    inp = input("Wrong number. Please enter a second four-digit integer: ")
b = int(inp)

if a > b:
    a, b = b, a

for x in range(a, b + 1):
    if str(x) == str(x)[::-1]:
        print(x)
```

17. Solution

```
for i in range(31):
    print(2 ** i)
```

18. Solution

```
offset = 10
i = 1
while i <= 401:
    print(i)
    i += offset
    offset += 2
```

19. Solution

```
for i in range(1, 101):
    print(-i, "\n", i)
```

20. Solution

First Approach

```
value = 0
for i in range(8):
    offset = 10 ** i
    value += offset
```

```
    print(value)
```

Second Approach

```
value = "1"
for i in range(8):
    print(value)
    value += "1"
```

21. Solution

First Approach

```
a = int(input())

fib_prev_prev = 0
fib_prev = 1
fib = 1
for i in range(a):
    print(fib)
    fib = fib_prev + fib_prev_prev
    fib_prev_prev = fib_prev
    fib_prev = fib
```

Second Approach

```
a = int(input())

fib_prev_prev, fib_prev, fib = 0, 1, 1
for i in range(a):
    print(fib)
    fib = fib_prev + fib_prev_prev
    fib_prev_prev, fib_prev = fib_prev, fib
```

22. Solution

First Approach

```
a = int(input())

fib_prev_prev = 0
fib_prev = 1
fib = 1
while fib < a:
    print(fib)
    fib = fib_prev + fib_prev_prev
    fib_prev_prev = fib_prev
    fib_prev = fib
```

Second Approach

```
a = int(input())

fib_prev_prev, fib_prev, fib = 0, 1, 1
while fib < a:
    print(fib)
    fib = fib_prev + fib_prev_prev
    fib_prev_prev, fib_prev = fib_prev, fib
```

23. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

inp = input("Enter a positive integer: ")
while not re.match(IS_NUMERIC, inp) or int(inp) < 1:
    inp = input("Wrong number. Please enter a positive integer: ")
n = int(inp)

nominator = 0
for i in range(2, 2 * n + 2, 2):
    nominator += i

denominator = 1
for i in range(1, n + 1):
    denominator *= i

y = nominator / denominator
print(y)
```

24. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

inp = input("Enter a positive integer: ")
while not re.match(IS_NUMERIC, inp) or int(inp) < 1:
    inp = input("Wrong number. Please enter a positive integer: ")
n = int(inp)

nominator = 0
sign = 1
for i in range(1, 2 * n + 1 + 2, 2):
    nominator += sign * i
    sign = -sign

y = nominator / n
print(y)
```

25. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

inp = input("Enter a positive integer: ")
while not re.match(IS_NUMERIC, inp) or int(inp) < 1:
    inp = input("Wrong number. Please enter a positive integer: ")
n = int(inp)

y = 0.5  #This is equal to the first two terms: 1 - 1 / 2

sign = 1
```

```
for i in range(3, n + 2, 2):
    y += sign / i
    sign = -sign

print(y)
```

26. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

inp = input("Enter a positive integer: ")
while not re.match(IS_NUMERIC, inp) or int(inp) < 1:
    inp = input("Wrong number. Please enter a positive integer: ")
n = int(inp)

y = 0
for i in range(1, n + 1):
    y += 1 / i ** (n - i + 1)

print(y)
```

27. Solution

```
n = int(input("Enter a non-negative integer: "))

factorial = 1
for i in range(1, n + 1):
    factorial *= i

print(factorial)
```

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

28. Solution

First Approach

```
ACCURACY = 0.00001

x = float(input())

exponential = 0
i = 0
while True:
    exponential_previous = exponential

    factorial = 1
    for j in range(1, i + 1):
        factorial *= j

    exponential += x ** i / factorial

    i += 1
```

```
    if abs(exponential - exponential_previous) <= ACCURACY: break

print("e(", x, ") ~=", exponential)
```

Second Approach

```
ACCURACY = 0.00001

x = float(input())

exponential = 1
i = 1
factorial = 1
while True:
    exponential_previous = exponential

    factorial *= i

    exponential += x ** i / factorial

    i += 1
    if abs(exponential - exponential_previous) <= ACCURACY: break

print("e(", x, ") ~=", exponential)
```

29. Solution

First Approach

```
ACCURACY = 0.00001

x = float(input())

sign = 1
sinus = 0
i = 1
while True:
    sinus_previous = sinus

    factorial = 1
    for j in range(1, i + 1):
        factorial *= j

    sinus += sign * x ** i / factorial

    sign = -sign
    i += 2
    if abs(sinus - sinus_previous) <= ACCURACY: break

print("sin(", x, ") ~=", sinus)
```

Second Approach

```
ACCURACY = 0.00001

x = float(input())
```

```
sign = -1
sinus = x
i = 3
factorial = 1
while True:
    sinus_previous = sinus

    factorial *= i * (i - 1)

    sinus += sign * x ** i / factorial

    sign = -sign
    i += 2
    if abs(sinus - sinus_previous) <= ACCURACY: break

print("sin(", x, ") ~=", sinus)
```

30. Solution

First Approach

```
ACCURACY = 0.00001

x = float(input())

sign = 1
cosinus = 0
i = 0
while True:
    cosinus_previous = cosinus

    factorial = 1
    for j in range(1, i + 1):
        factorial *= j

    cosinus += sign * x ** i / factorial

    sign = -sign
    i += 2
    if abs(cosinus - cosinus_previous) <= ACCURACY: break

print("cos(", x, ") ~=", cosinus)
```

Second Approach

```
ACCURACY = 0.00001

x = float(input())

sign = -1
cosinus = 1
i = 2
factorial = 1
while True:
    cosinus_previous = cosinus
```

```
    factorial *= i * (i - 1)

    cosinus += sign * x ** i / factorial

    sign = -sign
    i += 2
    if abs(cosinus - cosinus_previous) <= ACCURACY: break

print("cos(", x, ") ~=", cosinus)
```

31. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

maximum = -460
total = 0
for i in range(31):
    while True:
        inp = input("Enter temperature for day" + str(i + 1) + ": ")

        failure = False
        if not re.match(IS_NUMERIC, inp):
            print("Please enter numeric values!")
            failure = True
        elif float(inp) < -459.67:
            print("Please enter a value greater than 459.67")
            failure = True
        if failure == False: break
    t = float(inp)

    total += t
    if t > maximum:
        maximum = t

print(total / 31, maximum)
```

32. Solution

```
level = float(input())
if level != 9999:
    hour = int(input())
    minutes = int(input())

    maximum = level
    max_hour = hour
    max_minutes = minutes

    minimum = level
    min_hour = hour
    min_minutes = minutes

    level = float(input())
```

```
    while level != 9999:
        hour = int(input())
        minutes = int(input())

        if level > maximum:
            maximum = level
            max_hour = hour
            max_minutes = minutes

        if level < minimum:
            minimum = level
            min_hour = hour
            min_minutes = minutes

        level = float(input())

    print(maximum, max_hour, max_minutes, minimum, min_hour, min_minutes)
```

33. Solution

```
import string
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

alphabet = string.ascii_lowercase

while True:
    inp = input("Enter an integer between 1 and 26: ")

    failure = False
    if not re.match(IS_NUMERIC, inp):
        print("Please enter numeric values!")
        failure = True
    elif int(inp) < 1:
        print("Please enter positive integers!")
        failure = True
    elif int(inp) > 26:
        print("Please enter a value less than or equal to 26!")
        failure = True
    if failure == False: break
a = int(inp)

while True:
    inp = input("Enter an integer between 1 and 26: ")

    failure = False
    if not re.match(IS_NUMERIC, inp):
        print("Please enter numeric values!")
        failure = True
    elif int(inp) < 1:
        print("Please enter positive integers!")
        failure = True
    elif int(inp) > 26:
        print("Please enter a value less than or equal to 26!")
```

```
        failure = True
    if failure == False: break
b = int(inp)

if a > b:
    a, b = b, a

for i in range(a, b + 1):
    print(alphabet[i - 1])
```

34. Solution

```
import random

secret_number = random.randrange(1, 101)

attempts = 1
guess = int(input("Enter a guess: "))
while guess != secret_number:
    if guess > secret_number:
        print("Your guess is bigger than my secret number. Try again.")
    else:
        print("Your guess is smaller than my secret number. Try again.")
    attempts += 1
    guess = int(input("Enter a guess: "))

print("You found it!")
print("Attempts:", attempts)
```

35. Solution

```
import random

for i in range(2):
    secret_number = random.randrange(1, 101)

    attempts = 1
    guess = int(input("Enter a guess: "))
    while guess != secret_number:
        if guess > secret_number:
            print("Your guess is bigger than my secret number. Try again.")
        else:
            print("Your guess is smaller than my secret number. Try again.")
        attempts += 1
        guess = int(input("Enter a guess: "))
    print("You found it!")
    print("Attempts:", attempts)

    if i == 0:
        first_player_attempts = attempts

if first_player_attempts < attempts:
    print("First player wins!")
```

```
elif first_player_attempts > attempts:
    print("Second player wins!")
else:
    print("It's a draw")
```

36. Solution

```
while True:
    print("1. 4/3 TV Screen")
    print("2. 16/9 TV Screen")
    print("3. Exit")
    choice = int(input("Enter a choice: "))

    if choice == 1:
        diagonal = int(input("Enter diagonal: "))
        print("Width:", diagonal * 0.8)
        print("Height:", diagonal * 0.6)
    elif choice == 2:
        diagonal = int(input("Enter diagonal: "))
        print("Width:", diagonal * 0.87)
        print("Height:", diagonal * 0.49)
    elif choice == 3: break
```

37. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

inp = input("Enter total number of students: ")
while not re.match(IS_NUMERIC, inp) or int(inp) < 1:
    inp = input("Wrong number. Please enter total number of students: ")
n = int(inp)

total = 0
total_a = 0
count_a = 0
total_b = 0
count_b = 0
total_a_boys = 0
count_a_boys = 0
count_cdef_girls = 0

maximum = -1
minimum = 101

for i in range(n):
    inp = input("Enter grade for student No" + str(i + 1) + ": ")
    while not re.match(IS_NUMERIC, inp) or int(inp) < 0 or int(inp) > 100:
        inp = input("Wrong grade. Please enter grade for student No" + str(i + 1) + ": ")
    grade = int(inp)

    sex = input("Enter sex for student No" + str(i + 1) + ": ").upper()
    while sex != "M" and sex != "F":
```

```
        sex = input("Wrong sex. Please enter sex for student No" + str(i + 1) + ": ").upper()

    if 90 <= grade <= 100:
        total_a += grade
        count_a += 1
        if sex == "M":
            total_a_boys += grade
            count_a_boys += 1
    elif 80 <= grade <= 89:
        total_b += grade
        count_b += 1
    else:
        if sex == "F":
            count_cdef_girls += 1

    if grade > maximum:
        maximum = grade

    if grade < minimum:
        minimum = grade

    total += grade

if count_a > 0:
    print("The average value of those who got an 'A' is: ")
    print(total_a / count_a)
if count_b > 0:
    print("The average value of those who got a 'B' is: ")
    print(total_b / count_b)
if count_a_boys > 0:
    print("The average value of boys who got an 'A' is: ")
    print(total_a_boys / count_a_boys)

print("The total number of girls that got less than 'B' is:", count_cdef_girls)
print("The highest grade is:", maximum)
print("The lowest grade is:", minimum)
print("The average grade of the whole class is:", total / n)
```

38. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

while True:
    inp = input("Enter amount: ")
    while not re.match(IS_NUMERIC, inp) or float(inp) < 0:
        inp = input("Wrong amount. Please enter amount: ")
    amount = float(inp)

    if amount < 20:
        discount = 0
    elif amount < 50:
        discount = 3
    elif amount < 100:
```

```
        discount = 5
    else:
        discount = 10

    print("Discount:", discount, "%", sep = "")
    print("Amount to pay (discount included):", (amount - amount * discount / 100))

    answer = input("Would you like to repeat? ")
    if answer.upper() != "YES": break
```

39. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"
TAX_RATE = 0.25

inp = input("Enter number of Kilowatt-hours consumed: ")
while not re.match(IS_NUMERIC, inp) or int(inp) < 0 and int(inp) != -1:
    inp = input("Wrong value. Please enter number of Kilowatt-hours consumed: ")
kwh = int(inp)

while kwh != -1:
    if kwh <= 400:
        t = kwh * 0.11
    elif kwh <= 1500:
        t = 400 * 0.11 + (kwh - 400) * 0.22
    elif 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
    print("Total amount to pay (taxes included):", t)

    inp = input("Enter number of Kilowatt-hours consumed: ")
    while not re.match(IS_NUMERIC, inp) or int(inp) < 0 and int(inp) != -1:
        inp = input("Wrong value. Please enter number of Kilowatt-hours consumed: ")
    kwh = int(inp)
```

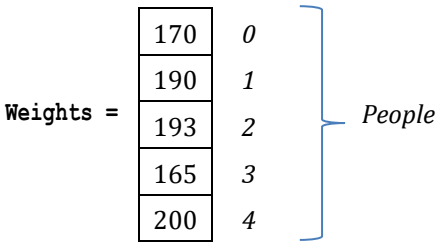
Chapter 30

30.3 Answers of Review Questions: True/False

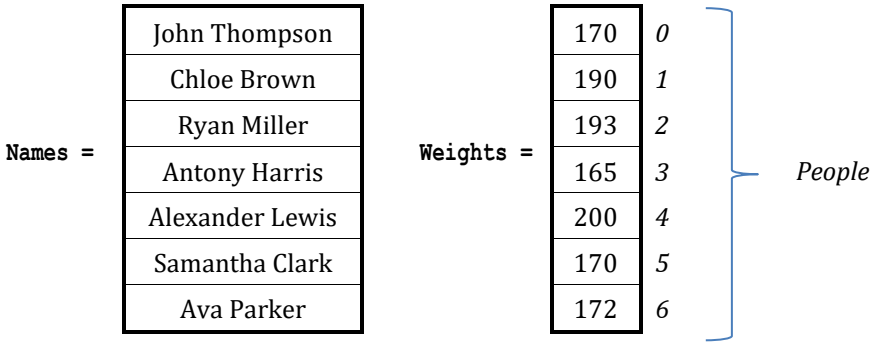
1. true
2. true
3. false
4. false
5. false
6. true
7. true
8. false
9. true

30.4 Answers of Review Exercises

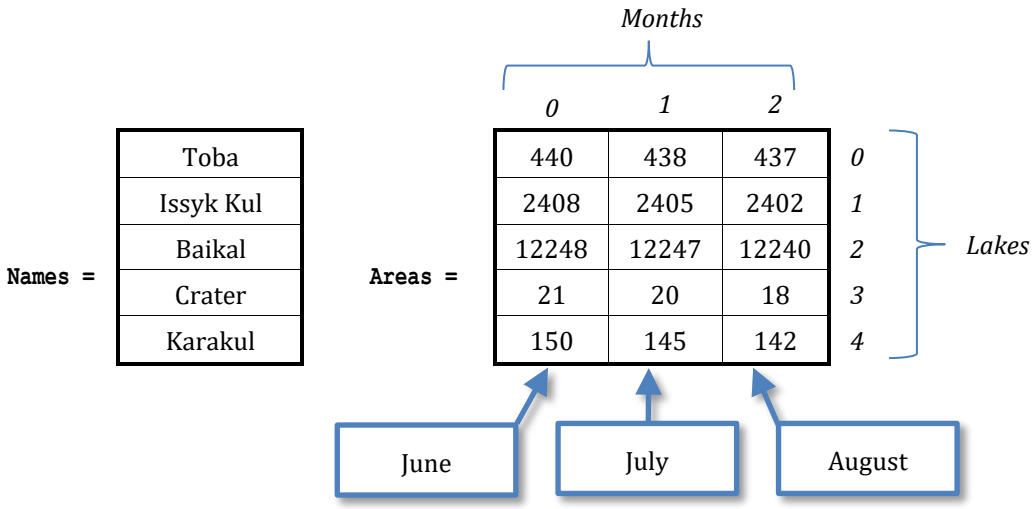
1. Solution



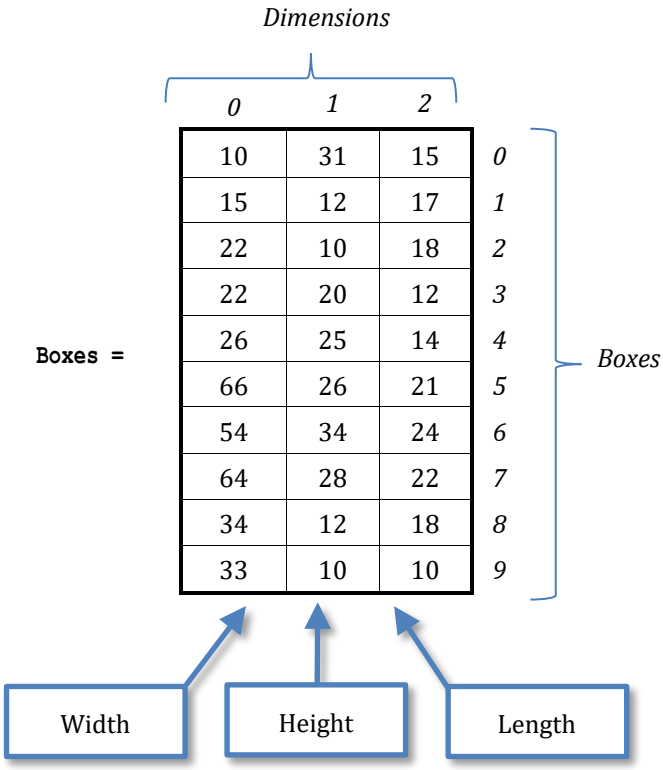
2. Solution



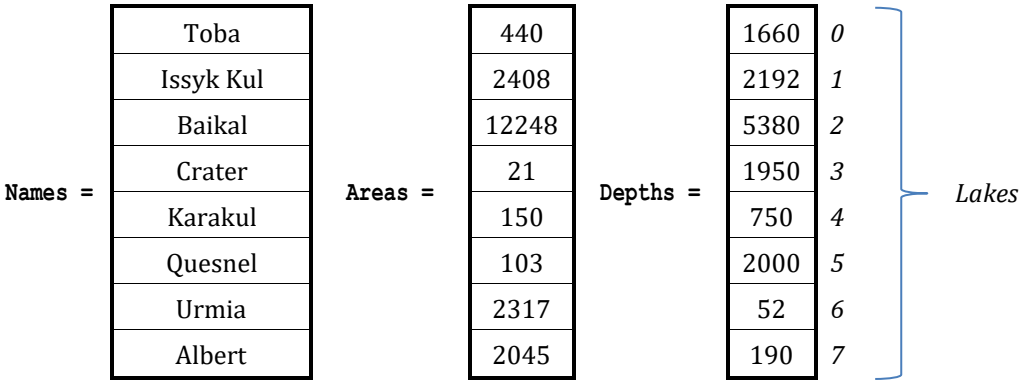
3. Solution



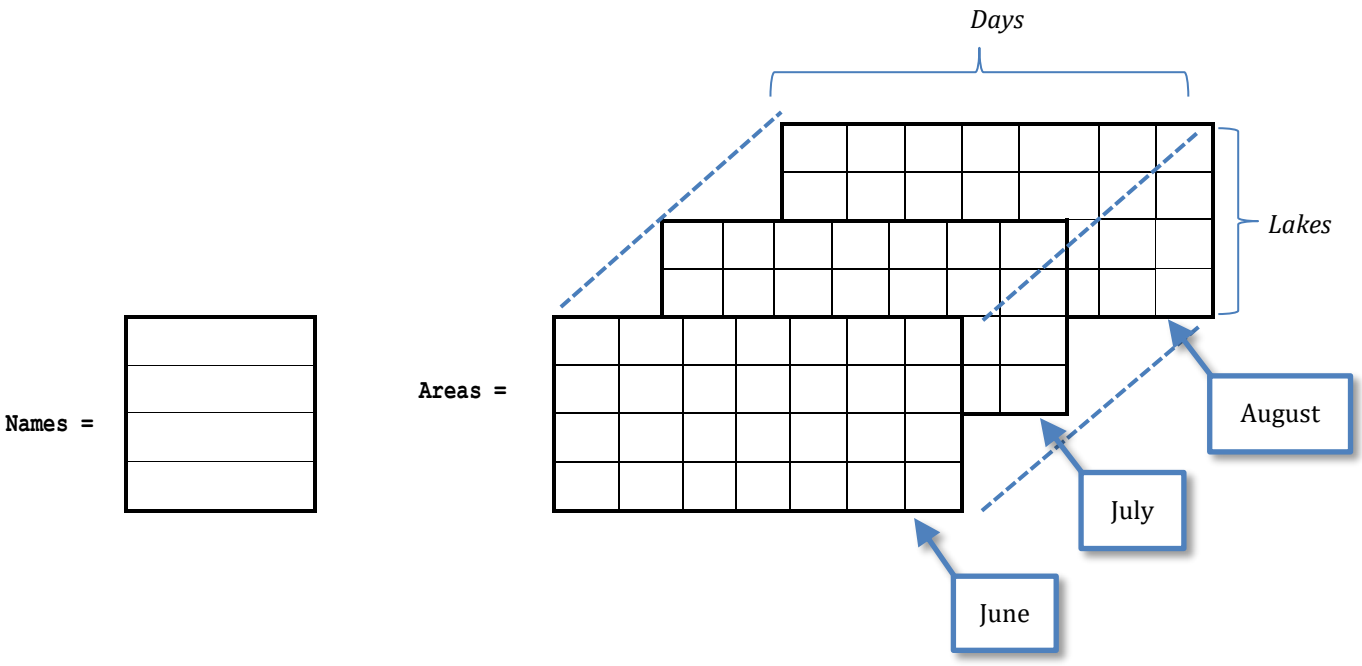
4. Solution



5. Solution



6. Solution



Chapter 31

31.5 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

31.6 Answers of Review Questions: Multiple Choice

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

31.7 Answers of Review Exercises

1. Solution

Step	Statement	x	a[0]	a[1]	a[2]
1	`a = [None] * 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	`a = [None] * 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	a = [None] * 4	?	?	?	?	?
2	a[1] = int(input())	?	?	**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	**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	a = [None] * 4	?	?	?	?	?
2	a[1] = int(input())	?	?	**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	**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	a = [None] * 4	?	?	?	?	?
2	a[1] = int(input())	?	?	**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	**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	`a = [None] * 4`	?	?	?	?	?
2	`a[1] = int(input())`	?	?	**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	`a = [None] * 4`	?	?	?	?	?
2	`a[1] = int(input())`	?	?	**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	`a = [None] * 4`	?	?	?	?	?
2	`a[1] = int(input())`	?	?	**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	`a = [None] * 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	`a = [17, 12, 45, 12, 12, 49]`	?	**17**	**12**	**45**	**12**	**12**	**49**
2	`i = 0`	**0**	17	12	45	12	12	49
3	`if a[i] == 12:`				False			
4	`a[i] += 1`	0	**18**	12	45	12	12	49
5	`i = 1`	**1**	18	12	45	12	12	49
6	`if a[i] == 12:`				True			
7	`a[i] -= 1`	1	18	**11**	45	12	12	49
8	`i = 2`	**2**	18	11	45	12	12	49
9	`if a[i] == 12:`				False			
10	`a[i] += 1`	2	18	11	**46**	12	12	49
11	`i = 3`	**3**	18	11	46	12	12	49
12	`if a[i] == 12:`				True			
13	`a[i] -= 1`	3	18	11	46	**11**	12	49
14	`i = 4`	**4**	18	11	46	11	12	49
15	`if a[i] == 12:`				True			
16	`a[i] -= 1`	4	18	11	46	11	**11**	49
17	`i = 5`	**5**	18	11	46	11	11	49
18	`if a[i] == 12:`				False			
19	`a[i] += 1`	5	18	11	46	11	11	**50**

7. Solution

Step	Statement	i	a[0]	a[1]	a[2]	a[3]	a[4]	a[5]
1	`a = [10, 15, 12, 23, 22, 19]`	?	**10**	**15**	**12**	**23**	**22**	**19**
2	`i = 1`	**1**	10	15	12	23	22	19
3	`a[i] = a[i + 1] + a[i - 1]`	1	10	**22**	12	23	22	19
4	`i = 2`	**2**	10	22	12	23	22	19
5	`a[i] = a[i + 1] + a[i - 1]`	2	10	22	**45**	23	22	19
6	`i = 3`	**3**	10	22	45	23	22	19
7	`a[i] = a[i + 1] + a[i - 1]`	3	10	22	45	**67**	22	19

8	i = 4	**4**	10	22	45	67	22	19
9	a[i] = a[i + 1] + a[i - 1]	4	10	22	45	67	**86**	19

8. Solution

```
ELEMENTS = 100

a = [None] * ELEMENTS
for i in range(ELEMENTS):
    a[i] = float(input())

for i in range(ELEMENTS):
    print(a[i] ** 3)
```

9. Solution

```
ELEMENTS = 80

a = [None] * ELEMENTS
for i in range(ELEMENTS):
    a[i] = float(input())

for i in range(ELEMENTS - 1, -1, -1):
    print(a[i] ** 2)
```

10. Solution

```
ELEMENTS = 90

a = [None] * ELEMENTS
for i in range(ELEMENTS):
    a[i] = int(input())

for i in range(ELEMENTS - 1, -1, -1):
    if a[i] % 5 == 0:
        print(a[i])
```

11. Solution

```
ELEMENTS = 50

a = [None] * ELEMENTS
for i in range(ELEMENTS):
    a[i] = int(input())

for i in range(ELEMENTS):
    if a[i] % 2 == 0 or a[i] > 10:
        print(a[i])
```

12. Solution

```
ELEMENTS = 30

a = [None] * ELEMENTS
```

```
for i in range(ELEMENTS):
    a[i] = float(input())

total = 0
for i in range(ELEMENTS):
    if a[i] > 0:
        total += a[i]

print(total)
```

13. Solution

```
ELEMENTS = 50

a = [None] * ELEMENTS
for i in range(ELEMENTS):
    a[i] = int(input())

total = 0
for i in range(ELEMENTS):
    if 10 <= a[i] <= 99:
        total += a[i]

print(total)
```

14. Solution

```
ELEMENTS = 40

a = [None] * ELEMENTS
for i in range(ELEMENTS):
    a[i] = float(input())

sum_pos = 0
sum_neg = 0
for i in range(ELEMENTS):
    if a[i] > 0 :
        sum_pos += a[i]
    elif a[i] < 0:
        sum_neg += a[i]

print(sum_pos, sum_neg)
```

15. Solution

```
ELEMENTS = 20

a = [None] * ELEMENTS
for i in range(ELEMENTS):
    a[i] = float(input())

total = 0
for i in range(ELEMENTS):
    total += a[i]

print(total / ELEMENTS)
```

16. Solution

```
WORDS = 50

a = [None] * WORDS
for i in range(WORDS):
    a[i] = input()

for i in range(WORDS):
    if len(a[i]) >= 10:
        print(a[i])
```

17. Solution

First Approach

```
WORDS = 40

a = [None] * WORDS
for i in range(WORDS):
    a[i] = input()

for i in range(WORDS):
    count = 0
    for j in range(len(a[i])):
        if a[i][j] == "w":
            count += 1
    if count >= 2:
        print(a[i])
```

Second Approach

```
WORDS = 40

a = [None] * WORDS
for i in range(WORDS):
    a[i] = input()

for i in range(WORDS):
    count = 0
    for letter in a[i]:
        if letter == "w":
            count += 1
    if count >= 2:
        print(a[i])
```

Chapter 32

32.7 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

32.8 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

32.9 Answers of Review Exercises

1. Solution

Step	Statement	x	a
1	`a = [ [None] * 3 for i in range(2) ]`	?	? ? ? ? ? ?
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	a = [[None] * 3 for i in range(2)]	?	?	? ? ? ? ? ?
2	i = 0	**0**	?	? ? ? ? ? ?
3	j = 0	0	**0**	? ? ? ? ? ?
4	a[i][j] = [i + 1] * 5 + j	0	0	**5** ? ? ? ? ?
5	j = 1	0	**1**	5 ? ? ? ? ?
6	a[i][j] = [i + 1] * 5 + j	0	1	5 **6** ? ? ? ?
7	j = 2	0	**2**	5 6 ? ? ? ?
8	a[i][j] = [i + 1] * 5 + j	0	2	5 6 **7** ? ? ?
9	i = 1	**1**	2	5 6 7 ? ? ?

10	`j = 0`	1	**0**	5 6 7 ? ? ?
11	`a[i][j] = [i + 1] * 5 + j`	1	0	5 6 7 **10** ? ?
12	`j = 1`	1	**1**	5 6 7 10 ? ?
13	`a[i][j] = [i + 1] * 5 + j`	1	1	5 6 7 10 **11** ?
14	`j = 2`	1	**2**	5 6 ? 10 11 ?
15	`a[i][j] = [i + 1] * 5 + j`	1	2	5 6 ? 10 11 **12**

3. Solution

Step	Statement	i	j	a
1	`a = [ [None] * 3 for i in range(3) ]`	?	?	? ? ? ? ? ? ? ? ?
2	`j = 0`	?	**0**	? ? ? ? ? ? ? ? ?
3	`i = 0`	0	**0**	? ? ? ? ? ? ? ? ?
4	`a[i][j] = [i + 1] * 2 + j * 4`	0	0	**2** ? ? ? ? ? ? ? ?
5	`i = 1`	**1**	0	2 ? ? ? ? ? ? ? ?

6	`a[i][j] = [i + 1] * 2 + j * 4`	1	0	2 ? ? **4** ? ? 6 ? ?
7	`i = 2`	**2**	0	2 ? ? 4 ? ? 6 ? ?
8	`a[i][j] = [i + 1] * 2 + j * 4`	2	0	2 ? ? 4 ? ? **6** ? ?
9	`j = 1`	2	**1**	2 ? ? 4 ? ? 6 ? ?
10	`i = 0`	**0**	1	2 ? ? 4 ? ? 6 ? ?
11	`a[i][j] = [i + 1] * 2 + j * 4`	0	1	2 **6** ? 4 ? ? 6 ? ?
12	`i = 1`	**1**	1	2 6 ? 4 ? ? 6 ? ?
13	`a[i][j] = [i + 1] * 2 + j * 4`	1	1	2 6 ? 4 **8** ? 6 ? ?
14	`i = 2`	**2**	1	2 6 ? 4 8 ? 6 ? ?
15	`a[i][j] = [i + 1] * 2 + j * 4`	2	1	2 6 ? 4 8 ? 6 **10** ?
16	`j = 2`	2	**2**	2 6 ? 4 8 ? 6 10 ?

17	`i = 0`	**0**	2	2 6 ? 4 8 ? 6 10 ?
18	`a[i][j] = [i + 1] * 2 + j * 4`	0	2	2 6 **10** 4 8 ? 6 10 ?
19	`i = 1`	**1**	2	2 6 10 4 8 ? 6 10 ?
20	`a[i][j] = [i + 1] * 2 + j * 4`	1	2	2 6 10 4 8 **12** 6 10 ?
21	`i = 2`	**2**	2	2 6 10 4 8 12 6 10 ?
22	`a[i][j] = [i + 1] * 2 + j * 4`	2	2	2 6 10 4 8 12 6 10 **14**

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

```
ROWS = 10
COLUMNS = 15

a = [ [None] * COLUMNS for i in range(ROWS) ]
for i in range(ROWS):
    for j in range(COLUMNS):
        a[i][j] = int(input())

for i in range(ROWS):
    for j in range(COLUMNS):
        if a[i][j] % 2 != 0:
            print(i, ",", j)
```

10. Solution

```
ROWS = 10
COLUMNS = 6

a = [ [None] * COLUMNS for i in range(ROWS) ]
for i in range(ROWS):
    for j in range(COLUMNS):
        a[i][j] = float(input())

for i in range(ROWS):
    for j in range(0, COLUMNS, 2):
        print(a[i][j])
```

11. Solution

```
ROWS = 12
COLUMNS = 8

a = [ [None] * COLUMNS for i in range(ROWS) ]
for i in range(ROWS):
    for j in range(COLUMNS):
        a[i][j] = float(input())

total = 0
for i in range(1, ROWS, 2):
    for j in range(0, COLUMNS, 2):
        total += a[i][j]
print(total)
```

12. Solution

```
N = 8

a = [ [None] * N for i in range(N) ]
for i in range(N):
    for j in range(N):
        a[i][j] = float(input())

sum_diagonal = 0
sum_antidiagonal = 0
for k in range(N):
    sum_diagonal += a[k][k]
    sum_antidiagonal += a[k][N - k - 1]

print(sum_diagonal / N, sum_antidiagonal / N)
```

13. Solution

```
N = 5

a = [ [None] * N for i in range(N) ]
for i in range(N):
```

```
    for j in range(N):
        if i == N - j - 1:
            a[i][j] = 5
        elif i > N - j - 1:
            a[i][j] = 88
        else:
            a[i][j] = 11

for i in range(N):
    for j in range(N):
        print(a[i][j], end = "\t")
    print()
```

14. Solution

```
N = 5

a = [ [None] * N for i in range(N) ]
for i in range(N):
    for j in range(N):
        if i == N - j - 1:
            a[i][j] = 5
        elif i > N - j - 1:
            a[i][j] = 88
        else:
            a[i][j] = 11
        if i == j:
            a[i][j] = 0

for i in range(N):
    for j in range(N):
        print(a[i][j], end = "\t")
    print()
```

15. Solution

```
ROWS = 5
COLUMNS = 4

a = [ [None] * COLUMNS for i in range(ROWS) ]
for i in range(ROWS):
    for j in range(COLUMNS):
        a[i][j] = float(input())

for i in range(ROWS):
    for j in range(COLUMNS):
        if a[i][j] == int(a[i][j]):
            print(i, ",", j)
```

16. Solution

```
ROWS = 10
COLUMNS = 4
```

```
a = [ [None] * COLUMNS for i in range(ROWS) ]
for i in range(ROWS):
    for j in range(COLUMNS):
        a[i][j] = float(input())

count = 0
for i in range(ROWS):
    for j in range(COLUMNS):
        if a[i][j] < 0:
            count += 1

print(count)
```

17. Solution

```
ROWS = 3
COLUMNS = 4

a = [ [None] * COLUMNS for i in range(ROWS) ]
for i in range(ROWS):
    for j in range(COLUMNS):
        a[i][j] = input()

for i in range(ROWS):
    for j in range(COLUMNS):
        print(a[i][j], " ", end = "")
```

18. Solution

```
ROWS = 20
COLUMNS = 14

a = [ [None] * COLUMNS for i in range(ROWS) ]
for i in range(ROWS):
    for j in range(COLUMNS):
        a[i][j] = input()

for i in range(ROWS):
    for j in range(COLUMNS):
        if len(a[i][j]) < 5:
            print(a[i][j])
```

19. Solution

First Approach

```
ROWS = 20
COLUMNS = 14

a = [ [None] * COLUMNS for i in range(ROWS) ]
for i in range(ROWS):
    for j in range(COLUMNS):
        a[i][j] = input()
```

```
length_limits = [5, 10, 20]

for k in range(3):
    for i in range(ROWS):
        for j in range(COLUMNS):
            if len(a[i][j]) < length_limits[k]:
                print(a[i][j])
```

Second Approach

```
ROWS = 20
COLUMNS = 14

a = [ [None] * COLUMNS for i in range(ROWS) ]
for i in range(ROWS):
    for j in range(COLUMNS):
        a[i][j] = input()

for k in range(3):
    for i in range(ROWS):
        for j in range(COLUMNS):
            if len(a[i][j]) < 5 * 2 ** k:
                print(a[i][j])
```

Chapter 33

33.8 Answers of Review Questions: True/False

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

33.9 Answers of Review Questions: Multiple Choice

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

33.10 Answers of Review Exercises

1. Solution

```
STUDENTS = 15
TESTS = 5

grades = [ [None] * TESTS for i in range(STUDENTS) ]
for i in range(STUDENTS):
    for j in range(TESTS):
        grades[i][j] = int(input())

average = [None] * STUDENTS
for i in range(STUDENTS):
    average[i] = 0
    for j in range(TESTS):
        average[i] += grades[i][j]
    average[i] /= TESTS

for i in range(STUDENTS):
    print("Student No", (i + 1), ": ")

    if average[i] < 60:
        print("E/F")
    elif average[i] < 70:
        print("D")
    elif average[i] < 80:
        print("C")
    elif average[i] < 90:
        print("B")
    else:
```

```
        print("A")
```

2. Solution

First Approach

```
OBJECTS = 5
FALLS = 10

g = [ [None] * FALLS for i in range(OBJECTS) ]
for i in range(OBJECTS):
    for j in range(FALLS):
        g[i][j] = int(input())

for i in range(OBJECTS):
    total = 0
    for j in range(FALLS):
        total += g[i][j]
    print("Average g for object No", (i + 1), ":", (total / FALLS))

for j in range(FALLS):
    total = 0
    for i in range(OBJECTS):
        total += g[i][j]
    print("Average g for fall No", (j + 1), ":", (total / OBJECTS))

total = 0
for i in range(OBJECTS):
    for j in range(FALLS):
        total += g[i][j]

print("Overall average g:", (total / (OBJECTS * FALLS)))
```

Second Approach

```
import math
OBJECTS = 5
FALLS = 10

g = [ [None] * FALLS for i in range(OBJECTS) ]
for i in range(OBJECTS):
    for j in range(FALLS):
        g[i][j] = int(input())

for row in g:
    total = math.fsum(row)
    print("Average g for object No", (i + 1), ":", (total / FALLS))

for j in range(FALLS):
    total = 0
    for i in range(OBJECTS):
        total += g[i][j]
    print("Average g for fall No", (j + 1), ":", (total / OBJECTS))

total = 0
for row in g:
```

```
    total += math.fsum(row)
print("Overall average g:", (total / (OBJECTS * FALLS)))
```

3. Solution

First Approach

```
PLAYERS = 15
MATCHES = 12

points = [ [None] * MATCHES for i in range(PLAYERS) ]
for i in range(PLAYERS):
    for j in range(MATCHES):
        points[i][j] = int(input())

for i in range(PLAYERS):
    total = 0
    for j in range(MATCHES):
        total += points[i][j]
    print("Total number of points for player No", (i + 1), ":", total)

for j in range(MATCHES):
    total = 0
    for i in range(PLAYERS):
        total += points[i][j]
    print("Total number of points for match No", (j + 1), ":", total)
```

Second Approach

```
import math
PLAYERS = 15
MATCHES = 12

points = [ [None] * MATCHES for i in range(PLAYERS) ]
for i in range(PLAYERS):
    for j in range(MATCHES):
        points[i][j] = int(input())

for row in points:
    total = math.fsum(row)
    print("Total number of points for player No", (i + 1), ":", total)

for j in range(MATCHES):
    total = 0
    for i in range(PLAYERS):
        total += points[i][j]
    print("Total number of points for match No", (j + 1), ":", total)
```

4. Solution

```
CITIES = 20
HOURS = 24

temperatures = [ [None] * HOURS for i in range(CITIES) ]
for i in range(CITIES):
```

```
    for j in range(HOURS):
        temperatures[i][j] = float(input())

for j in range(HOURS):
    total = 0
    for i in range(CITIES):
        total += temperatures[i][j]
    if total / CITIES < 10:
        print("Hour:", (j + 1))
```

5. Solution

```
PLAYERS = 24
MATCHES = 10

names = [None] * PLAYERS
goals = [ [None] * MATCHES for i in range(PLAYERS) ]
for i in range(PLAYERS):
    names[i] = input()
    for j in range(MATCHES):
        goals[i][j] = int(input())

                                        # Or you can do the following:
for i in range(PLAYERS):                # for row in goals:
    total = 0                           #     print(names[i], ":", math.fsum(row) / MATCHES)
    for j in range(MATCHES):            #
        total += goals[i][j]            #
    print(names[i], ":", (total / MATCHES)) #

for j in range(MATCHES):
    total = 0
    for i in range(PLAYERS):
        total += goals[i][j]
    print("Match No", (j + 1), ":", total)
```

6. Solution

```
STUDENTS = 24
LESSONS = 10

names = [None] * STUDENTS
grades = [ [None] * LESSONS for i in range(STUDENTS) ]
for i in range(STUDENTS):
    names[i] = input()
    for j in range(LESSONS):
        grades[i][j] = int(input())

average = [None] * STUDENTS             # Or you can do the following:
for i in range(STUDENTS):               # for row in grades:
    total = 0                           #     average[i] = math.fsum(row) / LESSONS
    for j in range(LESSONS):            #     print(names[i], ":", average[i])
        total += grades[i][j]           #
    average[i] = total / LESSONS        #
```

```
    print(names[i], ":", average[i])    #

for j in range(LESSONS):
    total = 0
    for i in range(STUDENTS):
        total += grades[i][j]
    print(total / STUDENTS)

for i in range(STUDENTS):
    if average[i] < 60:
        print(names[i])

for i in range(STUDENTS):
    if average[i] > 89:
        print(names[i], "Bravo!")
```

7. Solution

```
ARTISTS = 15
JUDGES = 5

judge_names = [None] * JUDGES
for j in range(JUDGES):
    judge_names[j] = input("Enter name for judge No" + str(j + 1) + ": ")

artist_names = [None] * ARTISTS
song_titles = [None] * ARTISTS
score = [ [None] * JUDGES for i in range(ARTISTS) ]
for i in range(ARTISTS):
    artist_names[i] = input("Enter name for artist No" + str(i + 1) + ": ")
    song_titles[i] = input("Enter song title for artist " + artist_names[i] + ": ")
    for j in range(JUDGES):
        score[i][j] = int(input("Enter score for artist " + artist_names[i] + " gotten from judge " + \
                                judge_names[j] + ": "))

for i in range(ARTISTS):
    total = 0
    for j in range(JUDGES):
        total += score[i][j]
    print(artist_names[i], ",", song_titles[i], ":", total)

for j in range(JUDGES):
    total = 0
    for i in range(ARTISTS):
        total += score[i][j]
    print(judge_names[j], ":", total / ARTISTS)
```

8. Solution

```
PEOPLE = 30
MONTHS = 12

weights = [ [None] * MONTHS for i in range(PEOPLE) ]
```

```
heights = [ [None] * MONTHS for i in range(PEOPLE) ]
for i in range(PEOPLE):
    for j in range(MONTHS):
        weights[i][j] = int(input())
        heights[i][j] = int(input())

for i in range(PEOPLE):
    sum_weights = 0
    sum_heights = 0
    for j in range(MONTHS):
        sum_weights += weights[i][j]
        sum_heights += heights[i][j]
    average_weight = sum_weights / MONTHS
    average_height = sum_heights / MONTHS
    print(average_weight, ",", average_height)
    print(average_weight * 702 / average_height ** 2)

for i in range(PEOPLE):
    print(weights[i][4] * 702 / heights[i][4] ** 2)
    print(weights[i][7] * 702 / heights[i][7] ** 2)
```

9. Solution

```
VAT = 0.19
CONSUMERS = 1000

meter_reading = [ [None] * 2 for i in range(CONSUMERS) ]
for i in range(CONSUMERS):
    meter_reading[i][0] = int(input())
    meter_reading[i][1] = int(input())

total = 0
for i in range(CONSUMERS):
    consumed = meter_reading[i][1] - meter_reading[i][0]
    print(consumed)
    payment = consumed * 0.07
    payment += VAT * payment
    print(payment)

    total += consumed

print(total, total * 0.07 + total * 0.07 * VAT)
```

10. Solution

First Approach

```
CURRENCIES = 4
DAYS = 5

usd = float(input("Enter an amount in US dollars: "))
currency = ["British Pound Sterling", "Euro", "Canadian Dollar", "Australian Dollar"]

rate = [[1.579, 1.577, 1.572, 1.58, 1.584],
```

```
        [1.269, 1.27, 1.265, 1.24, 1.255],
        [0.895, 0.899, 0.884, 0.888, 0.863],
        [0.811, 0.815, 0.822, 0.829, 0.819]
       ]

for i in range(CURRENCIES):
    total = 0
    for j in range(DAYS):
        total += rate[i][j]
    average = total / DAYS
    print(usd, " US dollars = ", (usd / average), " ", currency[i], "s", sep = "")
```

Second Approach

```
import math
DAYS = 5

usd = float(input("Enter an amount in US dollars: "))
currency = ["British Pound Sterling", "Euro", "Canadian Dollar", "Australian Dollar"]

rate = [[1.579, 1.577, 1.572, 1.58, 1.584],
        [1.269, 1.27, 1.265, 1.24, 1.255],
        [0.895, 0.899, 0.884, 0.888, 0.863],
        [0.811, 0.815, 0.822, 0.829, 0.819]
       ]

for i in range(len(rate)):
    average = math.fsum(rate[i]) / DAYS     #rate[i] returns the whole row
    print(usd, " US dollars = ", (usd / average), " ", currency[i], "s", sep = "")
```

11. Solution

```
EMPLOYEES = 10
DAYS = 5

weekdays = ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday"]

pay_rate = float(input())

names = [None] * EMPLOYEES
hours_worked_per_day = [ [None] * DAYS for i in range(EMPLOYEES) ]
for i in range(EMPLOYEES):
    names[i] = input()
    for j in range(DAYS):
        hours_worked_per_day[i][j] = int(input())

hours_worked_per_week = [None] * EMPLOYEES
for i in range(EMPLOYEES):
    hours_worked_per_week[i] = 0
    for j in range(DAYS):
        hours_worked_per_week[i] += hours_worked_per_day[i][j]
    if hours_worked_per_week[i] > 40:
        print(names[i])

for i in range(EMPLOYEES):
```

```
    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)
    print(names[i], gross_pay)

for i in range(EMPLOYEES):
    if hours_worked_per_week[i] > 40:
        for j in range(DAYS):
            if hours_worked_per_day[i][j] > 8:
                print(names[i], weekdays[j], "Overtime!")

for j in range(DAYS):
    total = 0
    for i in range(EMPLOYEES):
        if hours_worked_per_day[i][j] <= 8:
            gross_pay = pay_rate * hours_worked_per_day[i][j]
        else:
            gross_pay = pay_rate * 8 + 1.5 * pay_rate * (hours_worked_per_day[i][j] - 8)
        total += gross_pay
    print(weekdays[j], total)
```

12. Solution

First Approach

```
ROWS = 3
COLUMNS = 4
ELEMENTS = ROWS * COLUMNS

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

b = [None] * ELEMENTS
k = 0
for i in range(ROWS):
    for j in range(COLUMNS):
        b[k] = a[i][j]
        k += 1

for k in range(ELEMENTS):
    print(b[k], end = " ")
```

Second Approach

```
a = [[9, 9, 2, 6],
     [4, 1, 10, 11],
     [12, 15, 7, 3]
    ]

b = []
for row in a:
    b = b + row
```

```
for element in b:
    print(element, end = " ")
```

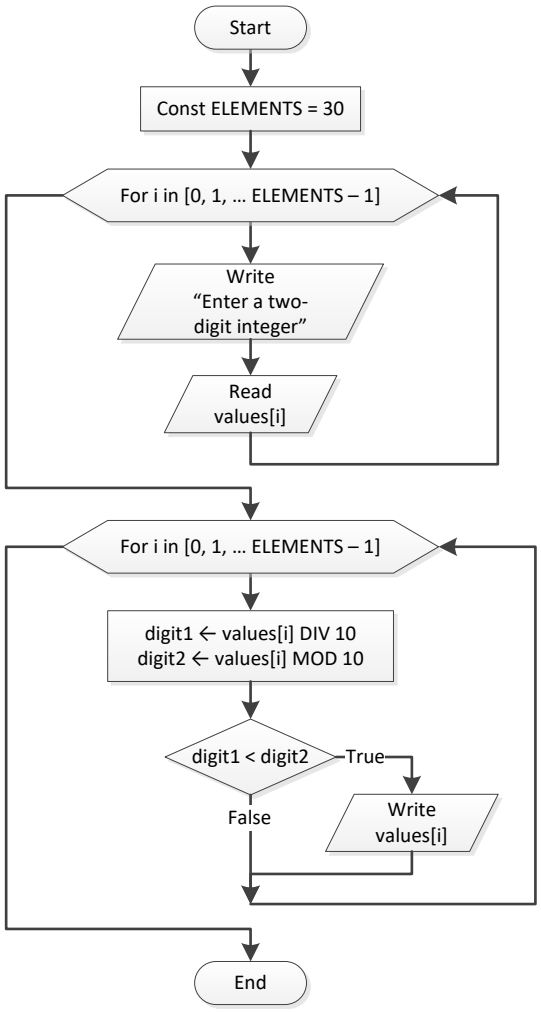
13. Solution

```
ROWS = 3
COLUMNS = 3

a = [16, 12, 3, 5, 6, 9, 18, 19, 20]

b = [ [None] * COLUMNS for i in range(ROWS) ]
k = 0
for i in range(ROWS - 1, -1, -1):
    for j in range(COLUMNS):
        b[i][j] = a[k]
        k += 1

for i in range(ROWS):
    for j in range(COLUMNS):
        print(b[i][j], end = "\t")
    print()
```

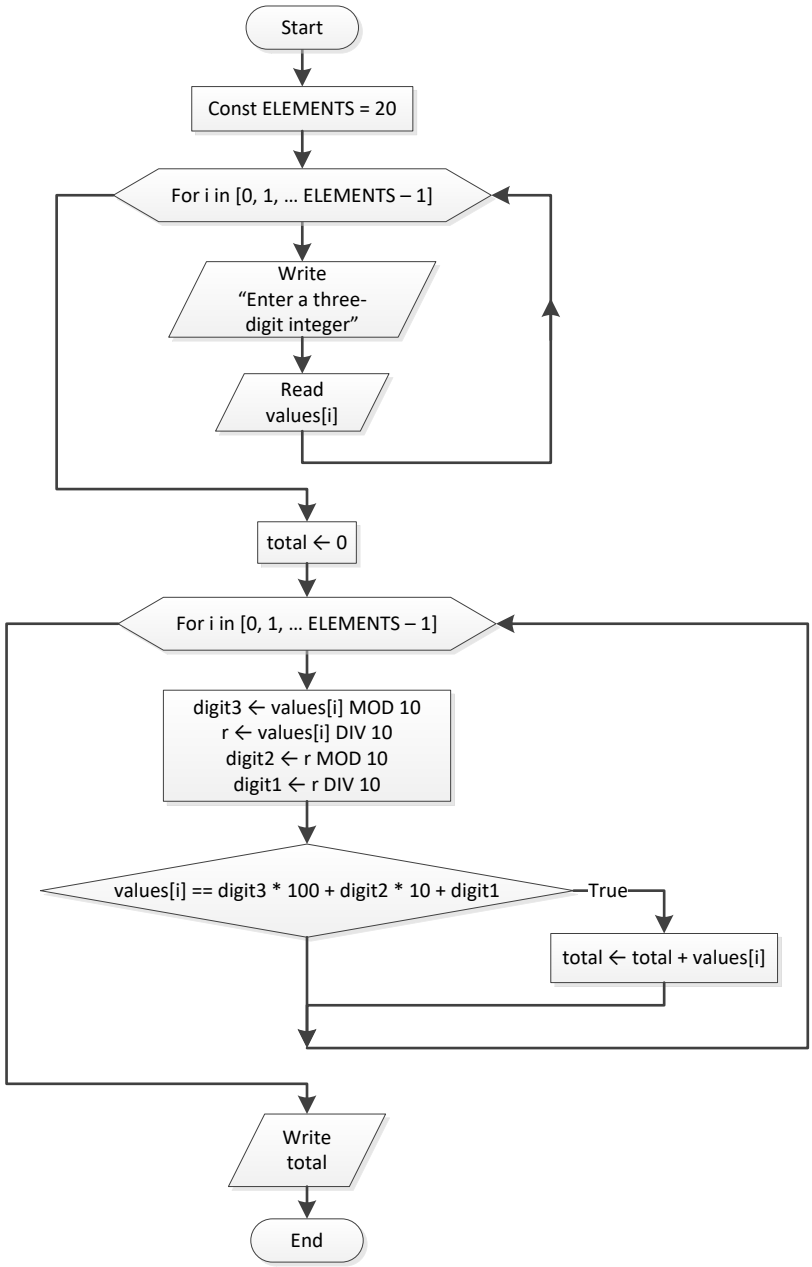
Chapter 34

34.4 Answers of Review Exercises

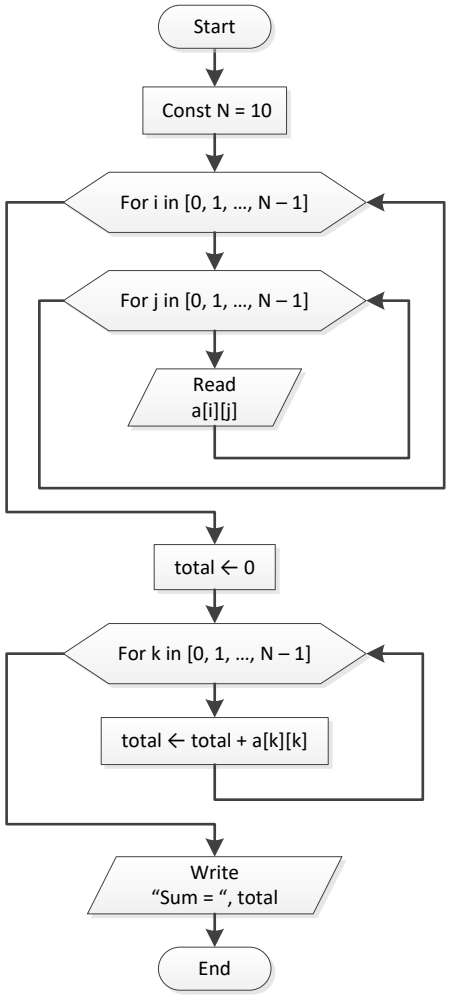
1. Solution



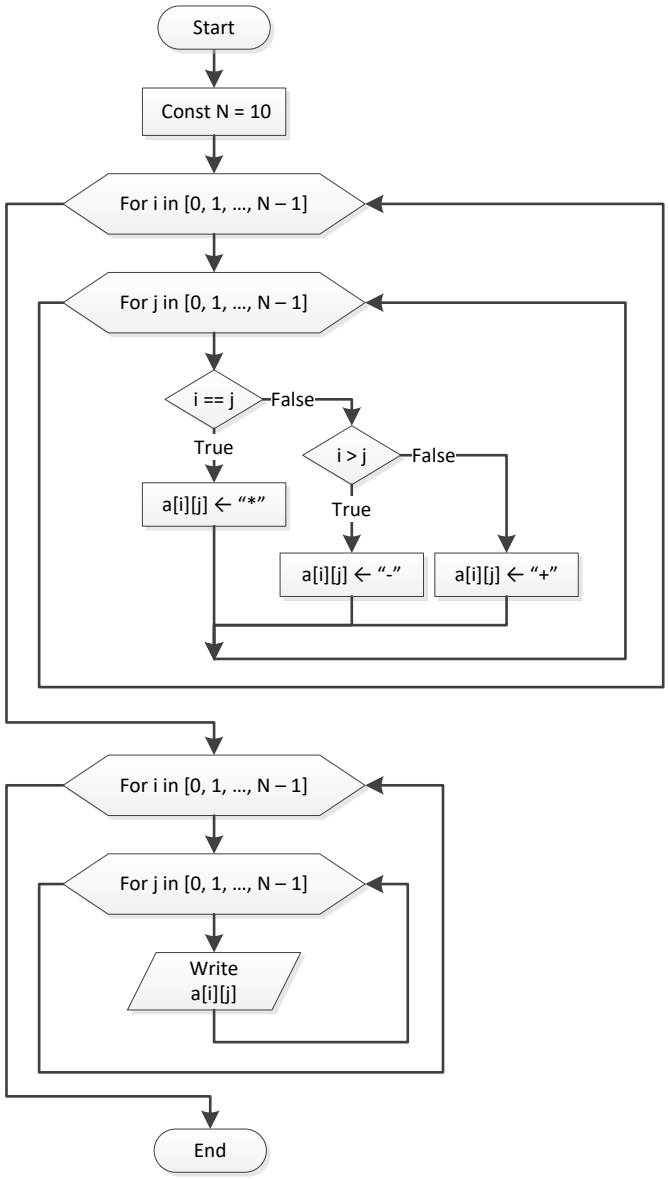
2. Solution


Start
Const ELEMENTS = 20
For i in [0, 1, … ELEMENTS – 1]
Write
“Enter a three-
digit integer”
Read
values[i]
total ← 0
For i in [0, 1, … ELEMENTS – 1]
digit3 ← values[i] MOD 10
r ← values[i] DIV 10
digit2 ← r MOD 10
digit1 ← r DIV 10
values[i] == digit3 * 100 + digit2 * 10 + digit1
True
total ← total + values[i]
Write
total
End

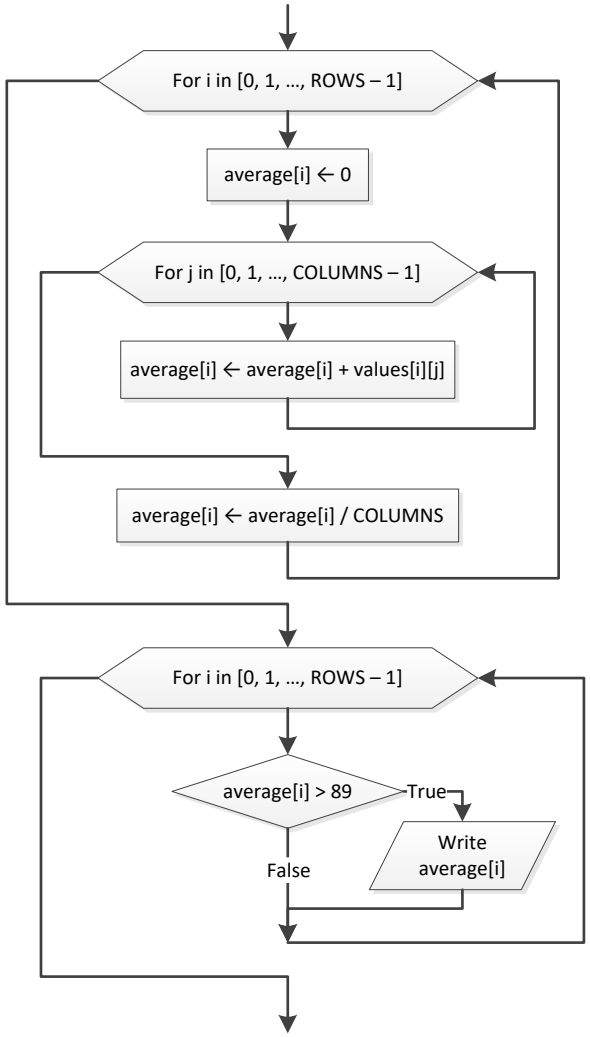
3. Solution


Start
Const N = 10
For i in [0, 1, …, N – 1]
For j in [0, 1, …, N – 1]
Read
a[i][j]
total ← 0
For k in [0, 1, …, N – 1]
total ← total + a[k][k]
Write
“Sum = “, total
End

4. Solution


Start
Const N = 10
For i in [0, 1, ..., N – 1]
For j in [0, 1, ..., N – 1]
i == j
False
True
i > j
False
True
a[i][j] ← "*"
a[i][j] ← "-"
a[i][j] ← "+"
For i in [0, 1, ..., N – 1]
For j in [0, 1, ..., N – 1]
Write
a[i][j]
End

5. Solution



6. Solution

```
i = 0
while i < PEOPLE:
    while True:
        a[i] = int(input())
        if a[i] % 2 == 2: break
    i += 1
```

7. Solution

```
for i in range(ELEMENTS):
    a[i] = float(input())
    while a[i] < 0:
        print("Error")
        a[i] = float(input())
```

8. Solution

```
i = 0
S = 0
a[i] = float(input())
i += 1
while i < 90:
    S += a[i - 1] * i
    a[i] = float(input())
    i += 1
print(S)
while i >= 0:
    print(a[i])
    i -= 5
```

9. Solution

```
for i in range(ROWS):
    maximum = a[i][0]
    for j in range(1, COLUMNS):
        if a[i][j] > maximum:
            maximum = a[i][j]
    print(maximum)
```

10. Solution

```
for i in range(ROWS):
    for j in range(COLUMNS):
        a[i][j] = float(input())
        while a[i][j] == 0:
            print("Error")
            a[i][j] = float(input())
```

Chapter 35

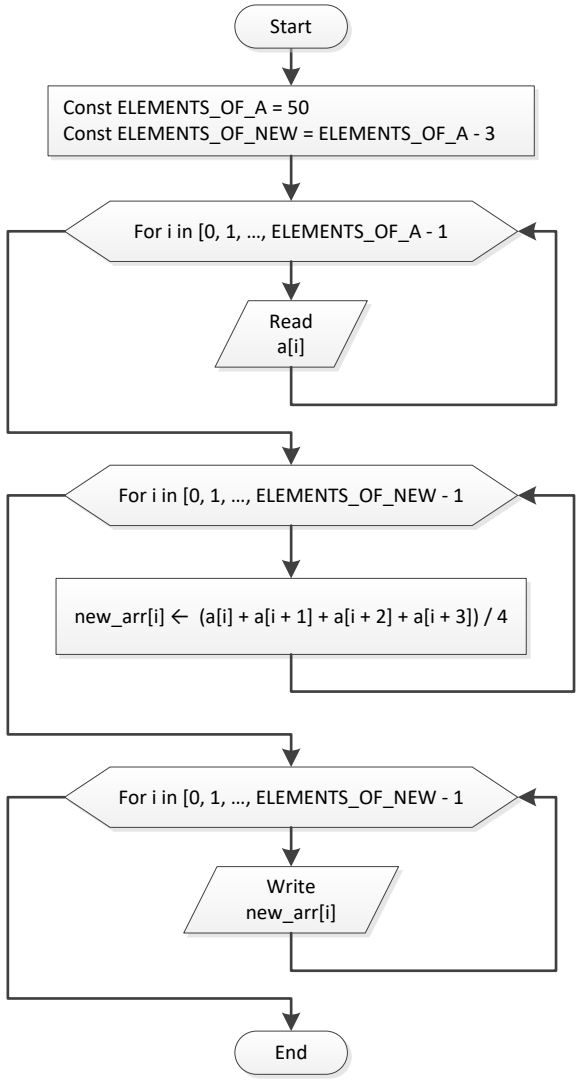
35.7 Answers of Review Questions: True/False

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

35.8 Answers of Review Exercises

1. Solution

First Approach



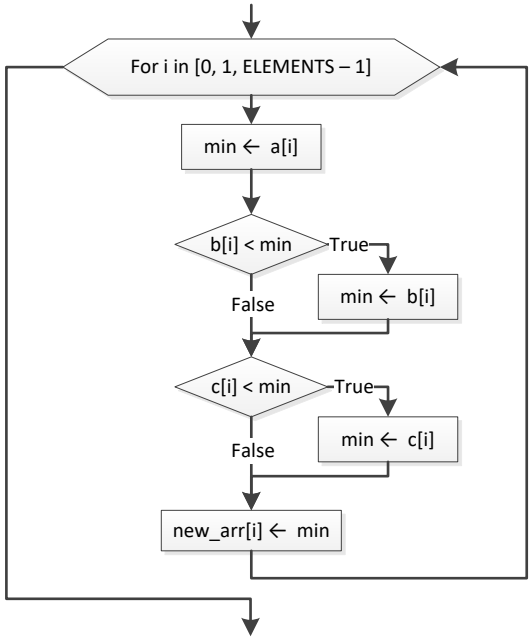

```
ELEMENTS_OF_A = 50
ELEMENTS_OF_NEW = ELEMENTS_OF_A - 3

a = [None] * ELEMENTS_OF_A
for i in range(ELEMENTS_OF_A):
    a[i] = float(input())

new_arr = [None] * ELEMENTS_OF_NEW
for i in range(ELEMENTS_OF_NEW):
    new_arr[i] = (a[i] + a[i + 1] + a[i + 2] + a[i + 3]) / 4

for i in range(ELEMENTS_OF_NEW):
    print(new_arr[i], end = "\t")
```

Second Approach

```
import math
ELEMENTS_OF_A = 50

a = [None] * ELEMENTS_OF_A

for i in range(ELEMENTS_OF_A):
    a[i] = float(input())

new_arr = []
for i in range(ELEMENTS_OF_A - 3):
    new_arr.append(math.fsum(a[i:i + 4]) / 4)

for element in new_arr:
    print(element, end = "\t")
```

2. Solution

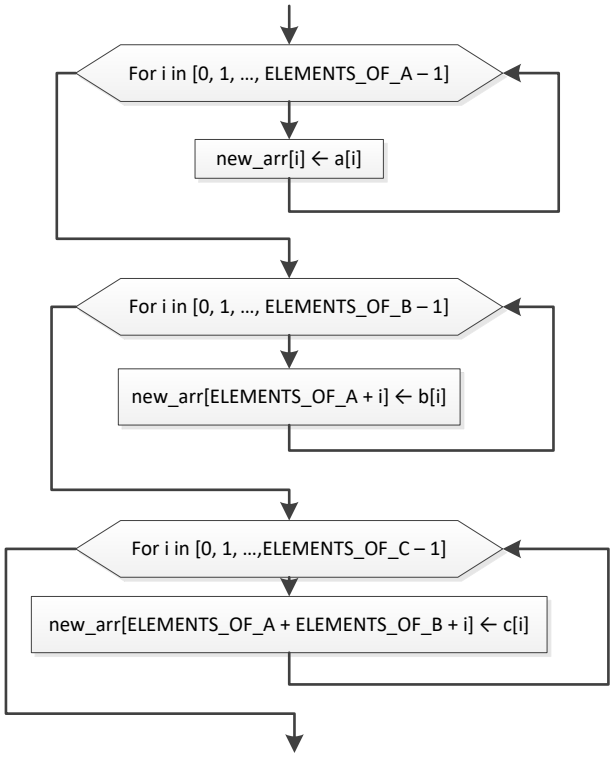
```
ELEMENTS = 15

a = [None] * ELEMENTS
for i in range(ELEMENTS):
    a[i] = float(input())

b = [None] * ELEMENTS
for i in range(ELEMENTS):
    b[i] = float(input())

c = [None] * ELEMENTS
for i in range(ELEMENTS):
    c[i] = float(input())

new_arr = [None] * ELEMENTS
for i in range(ELEMENTS):       # Or you can do the following:
    minimum = a[i]              # new_arr[i] = min(a[i], b[i], c[i])
    if b[i] < minimum:          #
        minimum = b[i]          #
    if c[i] < minimum:          #
        minimum = c[i]          #
    new_arr[i] = minimum        #

for i in range(ELEMENTS):
    print(new_arr[i])
```



3. Solution

First Approach

```
ELEMENTS_OF_A = 10
ELEMENTS_OF_B = 5
ELEMENTS_OF_C = 15
ELEMENTS_OF_NEW = ELEMENTS_OF_A + ELEMENTS_OF_B + ELEMENTS_OF_C

a = [None] * ELEMENTS_OF_A
for i in range(ELEMENTS_OF_A):
    a[i] = float(input())

b = [None] * ELEMENTS_OF_B
for i in range(ELEMENTS_OF_B):
    b[i] = float(input())

c = [None] * ELEMENTS_OF_C
for i in range(ELEMENTS_OF_C):
    c[i] = float(input())

new_arr = [None] * ELEMENTS_OF_NEW
for i in range(ELEMENTS_OF_C):
    new_arr[i] = c[i]
for i in range(ELEMENTS_OF_B):
    new_arr[ELEMENTS_OF_C + i] = b[i]
for i in range(ELEMENTS_OF_A):
    new_arr[ELEMENTS_OF_B + ELEMENTS_OF_C + i] = a[i]

for i in range(ELEMENTS_OF_NEW):
    print(new_arr[i], end = "\t")
```



Second Approach

```
ELEMENTS_OF_A = 10
ELEMENTS_OF_B = 5
ELEMENTS_OF_C = 15

a = [None] * ELEMENTS_OF_A
for i in range(ELEMENTS_OF_A):
    a[i] = float(input())

b = [None] * ELEMENTS_OF_B
for i in range(ELEMENTS_OF_B):
    b[i] = float(input())

c = [None] * ELEMENTS_OF_C
for i in range(ELEMENTS_OF_C):
    c[i] = float(input())

new_arr = []
for element in c:
    new_arr.append(element)
for element in b:
    new_arr.append(element)
for element in a:
    new_arr.append(element)

for element in new_arr:
    print(element, end = "\t")
```

Third Approach

```
ELEMENTS_OF_A = 10
ELEMENTS_OF_B = 5
ELEMENTS_OF_C = 15

a = [None] * ELEMENTS_OF_A
for i in range(ELEMENTS_OF_A):
    a[i] = float(input())

b = [None] * ELEMENTS_OF_B
for i in range(ELEMENTS_OF_B):
    b[i] = float(input())

c = [None] * ELEMENTS_OF_C
for i in range(ELEMENTS_OF_C):
    c[i] = float(input())

new_arr = c + b + a

for element in new_arr:
    print(element, end = "\t")
```

4. Solution

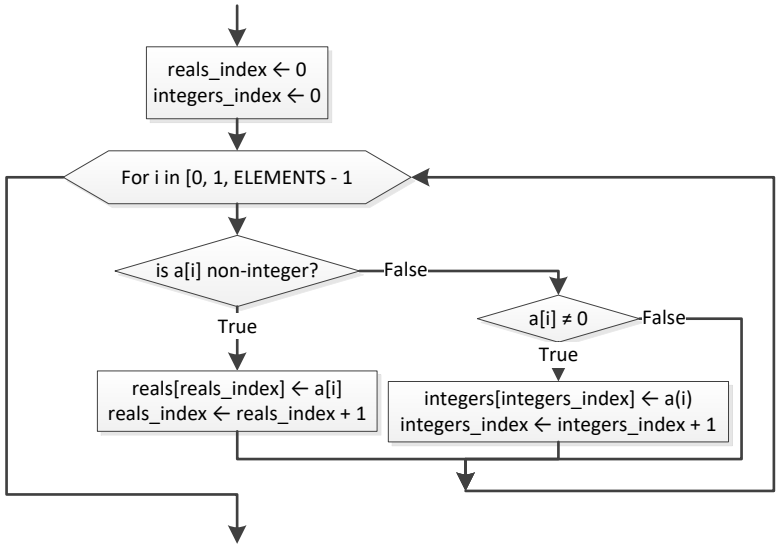
```
COLUMNS_OF_A = 10
COLUMNS_OF_B = 15
COLUMNS_OF_C = 20
ROWS = 5
COLUMNS = COLUMNS_OF_A + COLUMNS_OF_B + COLUMNS_OF_C

a = [ [None] * COLUMNS_OF_A for i in range(ROWS) ]
for i in range(ROWS):
    for j in range(COLUMNS_OF_A):
        a[i][j] = float(input())

b = [ [None] * COLUMNS_OF_B for i in range(ROWS) ]
for i in range(ROWS):
    for j in range(COLUMNS_OF_B):
        b[i][j] = float(input())

c = [ [None] * COLUMNS_OF_C for i in range(ROWS) ]
for i in range(ROWS):
    for j in range(COLUMNS_OF_C):
        c[i][j] = float(input())

new_arr = [ [None] * COLUMNS for i in range(ROWS) ]
for i in range(ROWS):
    for j in range(COLUMNS_OF_A):
        new_arr[i][j] = a[i][j]
for i in range(ROWS):
    for j in range(COLUMNS_OF_B):
        new_arr[i][COLUMNS_OF_A + j] = b[i][j]
for i in range(ROWS):
```

```
    for j in range(COLUMNS_OF_C):
        new_arr[i][COLUMNS_OF_A + COLUMNS_OF_B + j] = c[i][j]

for i in range(ROWS):
    for j in range(COLUMNS):
        print(new_arr[i][j], end = "\t")
    print()
```

5. Solution

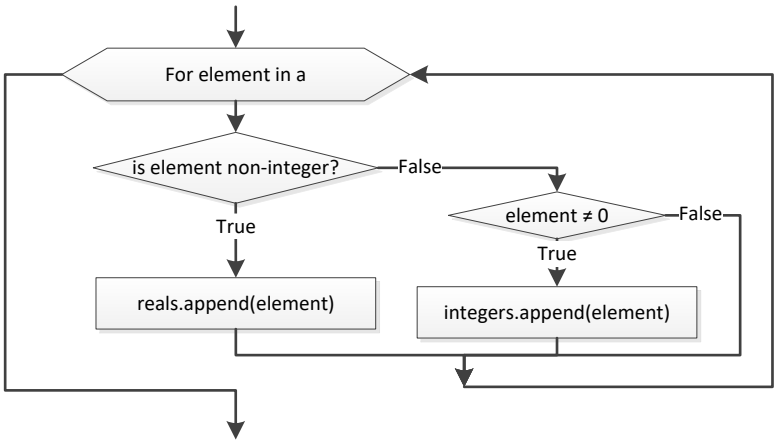
First Approach

```
ELEMENTS = 50

a = [None] * ELEMENTS
for i in range(ELEMENTS):
    a[i] = float(input())

reals = [None] * ELEMENTS
integers = [None] * ELEMENTS
reals_index = 0
integers_index = 0
for i in range(ELEMENTS):
    if a[i] != int(a[i]):
        reals[reals_index] = a[i]
        reals_index += 1
    elif a[i] != 0:
        integers[integers_index] = int(a[i])
        integers_index += 1

for i in range(reals_index):
    print(reals[i], end = "\t")

print()
for i in range(integers_index):
    print(integers[i], end = "\t")
```



Second Approach

```
ELEMENTS = 50

a = [None] * ELEMENTS
for i in range(ELEMENTS):
    a[i] = float(input())

reals = []
integers = []
for element in a:
    if element != int(element):
        reals.append(element)
    elif element != 0:
        integers.append(element)

for element in reals:
    print(element, end = "\t")

print()
for element in integers:
    print(element, end = "\t")
```



6. Solution

First Approach

```
ELEMENTS = 50

a = [None] * ELEMENTS
for i in range(ELEMENTS):
    a[i] = int(input())

b = []
for element in a:
    digit1, r = divmod(element, 100)
    digit2, digit3 = divmod(r, 10)

    if digit1 < digit2 < digit3:
        b.append(element)
```

```
for element in b:
    print(element, end = "\t")
```

Second Approach

```
ELEMENTS = 50

a = [None] * ELEMENTS
for i in range(ELEMENTS):
    a[i] = int(input())

b = []
for element in a:
    digit1, digit2, digit3 = str(element)     #digit1, digit2, digit3 are strings

    if digit1 < digit2 < digit3:              #Compare them as strings (no need to convert to integers)
        b.append(element)

for element in b:
    print(element, end = "\t")
```

7. Solution

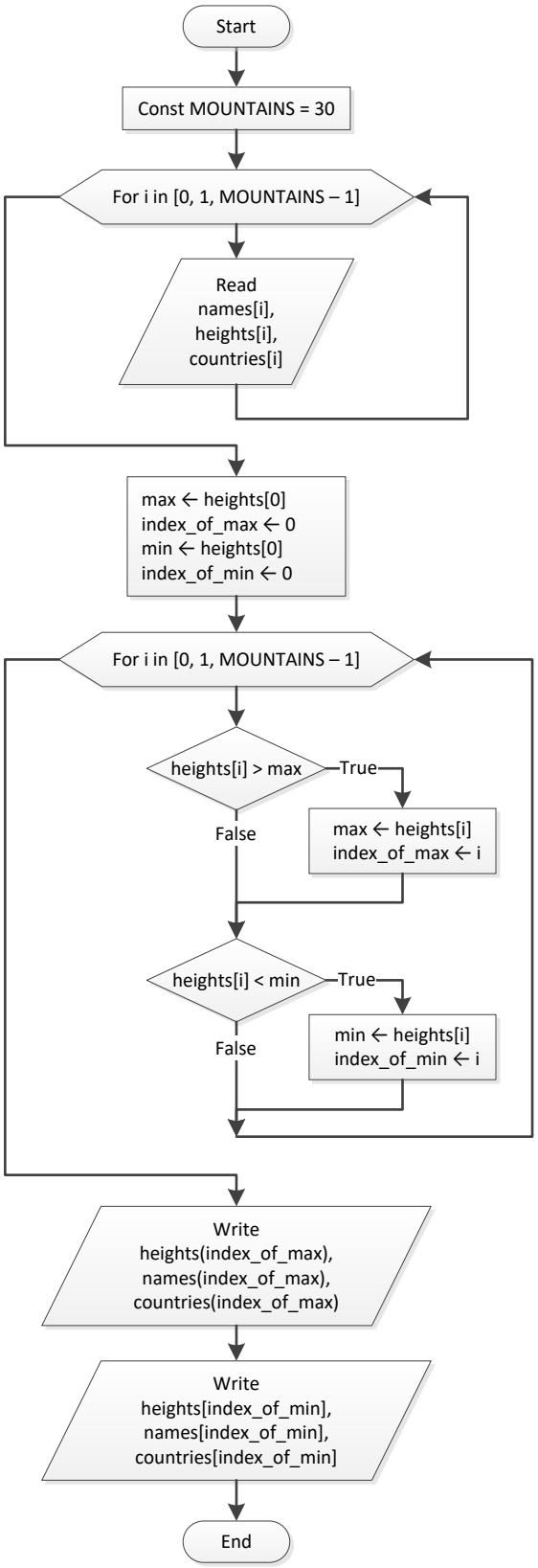
```
PRODUCTS = 10
CITIZENS = 1000

prod_names = [None] * PRODUCTS
answers = [ [None] * CITIZENS for i in range(PRODUCTS) ]
for i in range(PRODUCTS):
    prod_names[i] = input()
    for j in range(CITIZENS):
        answers[i][j] = input()
        while not("A" <= answers[i][j] <= "D"):
            print("Error! ")
            answers[i][j] = input()

count_A = [None] * PRODUCTS
for i in range(PRODUCTS):
    count_A[i] = 0
    for j in range(CITIZENS):
        if answers[i][j] == "A":
            count_A[i] += 1
    print(prod_names[i], count_A[i])

for j in range(CITIZENS):
    count_B = 0
    for i in range(PRODUCTS):
        if answers[i][j] == "B":
            count_B += 1
    print(count_B)

maximum = count_A[0]                  # This code fragment can be replaced by the statement
for i in range(1, PRODUCTS):          # maximum = max(count_A)
    if count_A[i] > maximum:          #
```

```
        maximum = count_A[i]          #

for i in range(PRODUCTS):
    if count_A[i] == maximum:
        print(prod_names[i])
```

8. Solution

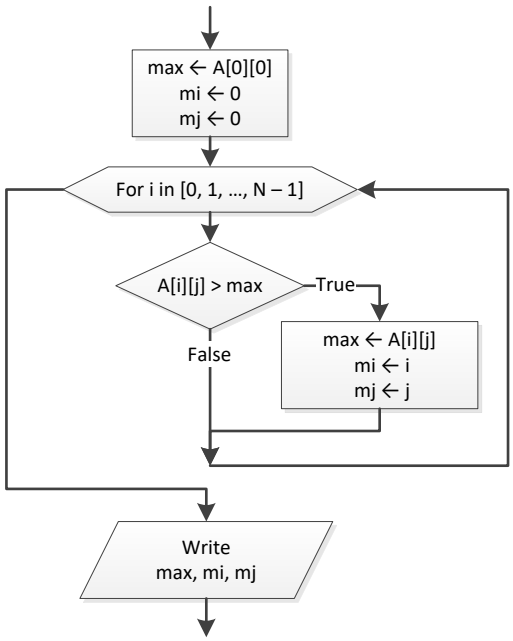
```
US_CITIES = 20
CANADIAN_CITIES = 20

us_names = [None] * US_CITIES
for i in range(US_CITIES):
    us_names[i] = input("Enter name for US city No" + str(i + 1) + ": ")

canadian_names = [None] * CANADIAN_CITIES
for j in range(CANADIAN_CITIES):
    canadian_names[j] = input("Enter name for Canadian city No" + str(j + 1) + ": ")

distances = [ [None] * CANADIAN_CITIES for i in range(US_CITIES) ]
for i in range(US_CITIES):
    for j in range(CANADIAN_CITIES):
        distances[i][j] = float(input("Enter distance between " + us_names[i] + " and " +        \
                                      canadian_names[j] + ": "))

for i in range(US_CITIES):
    minimum = distances[i][0]
    min_j = 0
    for j in range(1, CANADIAN_CITIES):
        if distances[i][j] < minimum:
            minimum = distances[i][j]
            min_j = j
    print("Closest Canadian city to", us_names[i], "is", canadian_names[min_j])
```

9. Solution


Start
Const MOUNTAINS = 30
For i in [0, 1, MOUNTAINS – 1]
Read
names[i],
heights[i],
countries[i]
max ← heights[0]
index_of_max ← 0
min ← heights[0]
index_of_min ← 0
For i in [0, 1, MOUNTAINS – 1]
heights[i] > max
True
False
max ← heights[i]
index_of_max ← i
heights[i] < min
True
False
min ← heights[i]
index_of_min ← i
Write
heights(index_of_max),
names(index_of_max),
countries(index_of_max)
Write
heights[index_of_min],
names[index_of_min],
countries[index_of_min]
End

```
MOUNTAINS = 30

names = [None] * MOUNTAINS
heights = [None] * MOUNTAINS
countries = [None] * MOUNTAINS
for i in range(MOUNTAINS):
    names[i] = input()
    heights[i] = float(input())
    countries[i] = input()

maximum = heights[0]
index_of_max = 0
minimum = heights[0]
index_of_min = 0
for i in range(1, MOUNTAINS):
    if heights[i] > maximum:
        maximum = heights[i]
        index_of_max = i
    if heights[i] < minimum:
        minimum = heights[i]
        index_of_min = i

print(heights[index_of_max], names[index_of_max], countries[index_of_max])
print()
print(heights[index_of_min], names[index_of_min], countries[index_of_min])
```

10. Solution



11. Solution

```
TEAMS = 26
```

```
GAMES = 15

names = [None] * TEAMS
results = [ [None] * GAMES for i in range(TEAMS) ]
for i in range(TEAMS):
    names[i] = input()
    for j in range(GAMES):
        results[i][j] = input()

points = [None] * TEAMS
for i in range(TEAMS):
    points[i] = 0
    for j in range(GAMES):
        if results[i][j] == "W":
            points[i] += 3
        elif results[i][j] == "T":
            points[i] += 1

maximum = points[0]
m_i = 0
for i in range(1, TEAMS):
    if points[i] > maximum:
        maximum = points[i]
        m_i = i

print(names[m_i])
```

12. Solution

First Approach

```
OBJECTS = 10
FALLS = 20

heights = [ [None] * FALLS for i in range(OBJECTS) ]
times = [ [None] * FALLS for i in range(OBJECTS) ]
for i in range(OBJECTS):
    for j in range(FALLS):
        heights[i][j] = float(input())
        times[i][j] = float(input())

g = [ [None] * FALLS for i in range(OBJECTS) ]
for i in range(OBJECTS):
    for j in range(FALLS):
        g[i][j] = 2 * heights[i][j] / times[i][j] ** 2

minimum = [None] * OBJECTS
maximum = [None] * OBJECTS
for i in range(OBJECTS):
    minimum[i] = g[i][0]
    maximum[i] = g[i][0]
    for j in range(1, FALLS):
        if g[i][j] < minimum[i]:
            minimum[i] = g[i][j]
```

```
        if g[i][j] > maximum[i]:
            maximum[i] = g[i][j]

for i in range(OBJECTS):
    print(minimum[i], maximum[i])

maxim = maximum[0]
minim = minimum[0]
for i in range(1, OBJECTS):
    if maximum[i] > maxim:
        maxim = maximum[i]
    if minimum[i] < minim:
        minim = minimum[i]

print(minim, maxim)
```

Second Approach

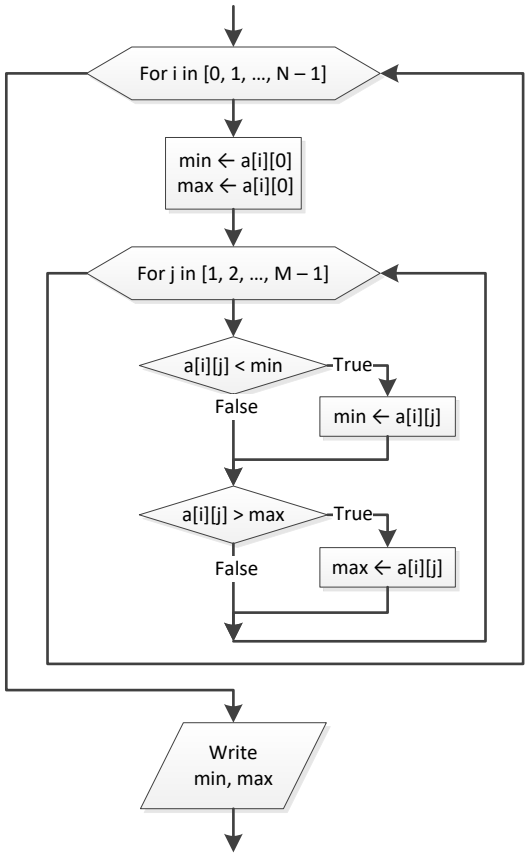
```
OBJECTS = 10
FALLS = 20

g = [ [None] * FALLS for i in range(OBJECTS) ]
for i in range(OBJECTS):
    for j in range(FALLS):
        height = float(input())
        time = float(input())
        g[i][j] = 2 * height / time ** 2

minimum = [None] * OBJECTS
maximum = [None] * OBJECTS
for i in range(OBJECTS):
    minimum[i] = min(g[i])      # g[i] returns the whole row
    maximum[i] = max(g[i])

for i in range(OBJECTS):
    print(minimum[i], maximum[i])

print(min(minimum), max(maximum))
```

13. Solution

```
STATIONS = 10
DAYS = 365

names = [None] * STATIONS
co2 = [ [None] * DAYS for i in range(STATIONS) ]
for i in range(STATIONS):
    names[i] = input()
    for j in range(DAYS):
        co2[i][j] = float(input())

                                    # Or you can do the following:
average = [None] * STATIONS         # average = []
for i in range(STATIONS):           # for row in co2:
    average[i] = 0                  #     average.append(math.fsum(row) / DAYS)
```

```
    for j in range(DAYS):           #
        average[i] += co2[i][j]     #
    average[i] /= DAYS              #

minimum = average[0]
m_i = 0
for i in range(1, STATIONS):
    if average[i] < minimum:
        minimum = average[i]
        m_i = i
print(names[m_i])
```

14. Solution



15. Solution

```
TEAMS = 20
GAMES = 10

names = [None] * TEAMS
results = [ [None] * GAMES for i in range(TEAMS) ]
for i in range(TEAMS):
    names[i] = input("Enter team name: ")
    for j in range(GAMES):
```

```
        results[i][j] = input("Enter result for team " + names[i] + " for game No" + str(j + 1) + ": ")
        while results[i][j] not in ["W", "L", "T"]:
            results[i][j] = input("Error! Enter only value W, L, or T: ")

points = [None] * TEAMS
for i in range(TEAMS):
    points[i] = 0
    for j in range(GAMES):
        if results[i][j] == "W":
            points[i] += 3
        elif results[i][j] == "T":
            points[i] += 1

for m in range(TEAMS - 1):
    swaps = False
    for n in range(TEAMS - 1, m, -1):
        if points[n] > points[n - 1]:
            points[n], points[n - 1] = points[n - 1], points[n]
            names[n], names[n - 1] = names[n - 1], names[n]
            swaps = True
    if swaps == False: break

print("Gold:", names[0])
print("Silver:", names[1])
print("Bronze:", names[2])
```

16. Solution

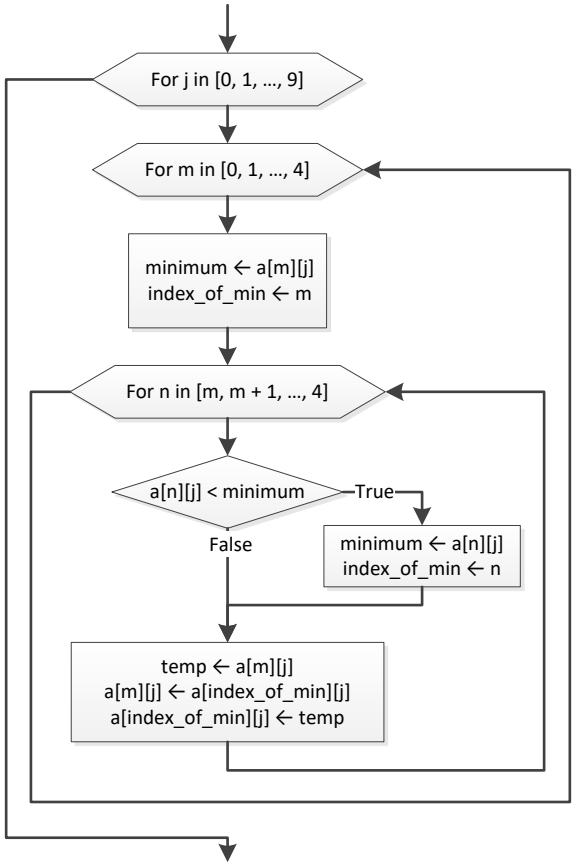
```
PEOPLE = 50

names = [None] * PEOPLE
heights = [None] * PEOPLE
for i in range(PEOPLE):
    names[i] = input("Enter name for person No." + str(i + 1) + ": ")
    heights[i] = float(input("Enter height for person No." + str(i + 1) + ": "))

for m in range(PEOPLE - 1):
    for n in range(PEOPLE - 1, m, -1):
        if heights[n] > heights[n - 1]:
            heights[n], heights[n - 1] = heights[n - 1], heights[n]
            names[n], names[n - 1] = names[n - 1], names[n]
        elif heights[n] == heights[n - 1]:
            if names[n] < names[n - 1]:
                names[n], names[n - 1] = names[n - 1], names[n]

for i in range(PEOPLE):
    print(heights[i], names[i])
```

17. Solution

```
ARTISTS = 12
JUDGES = 10
```

```
artist_names = [None] * ARTISTS
score = [ [None] * JUDGES for i in range(ARTISTS) ]
for i in range(ARTISTS):
    artist_names[i] = input("Enter name for artist No." + str(i + 1) + ": ")
    for j in range(JUDGES):
        print("Enter score for artist: ", artist_names[i])
        score[i][j] = int(input(" gotten from judge No." + str(j + 1) + ": "))

total = [None] * ARTISTS            # Or you can do the following:
for i in range(ARTISTS):            # for row in score:
    total[i] = 0                    #     total[i] = math.fsum(row)
    for j in range(1, JUDGES):      #
        total[i] += score[i][j]     #

                                            # Or you can do the following:
for i in range(ARTISTS):                    # for i in range(ARTISTS):
    minimum = score[i][0]                   #     minimum = min(score[i])
    maximum = score[i][0]                   #     maximum = max(score[i])
    for j in range(1, JUDGES):              #     total[i] -= minimum - maximum
        if score[i][j] < minimum:           #     print(total[i])
            minimum = score[i][j]           #
        if score[i][j] > maximum:           #
            maximum = score[i][j]           #
    total[i] = total[i] - minimum - maximum #
    print(total[i])                         #

for m in range(ARTISTS - 1):
    for n in range(ARTISTS - 1, m, -1):
        if total[n] > total[n - 1]:
            total[n], total[n - 1] = total[n - 1], total[n]
            artist_names[n], artist_names[n - 1] = artist_names[n - 1], artist_names[n]
        elif total[n] == total[n - 1]:
            if artist_names[n] < artist_names[n - 1]:
                artist_names[n], artist_names[n - 1] = artist_names[n - 1], artist_names[n]

for i in range(ARTISTS):
    print(artist_names[i], total[i])
```

18. Solution

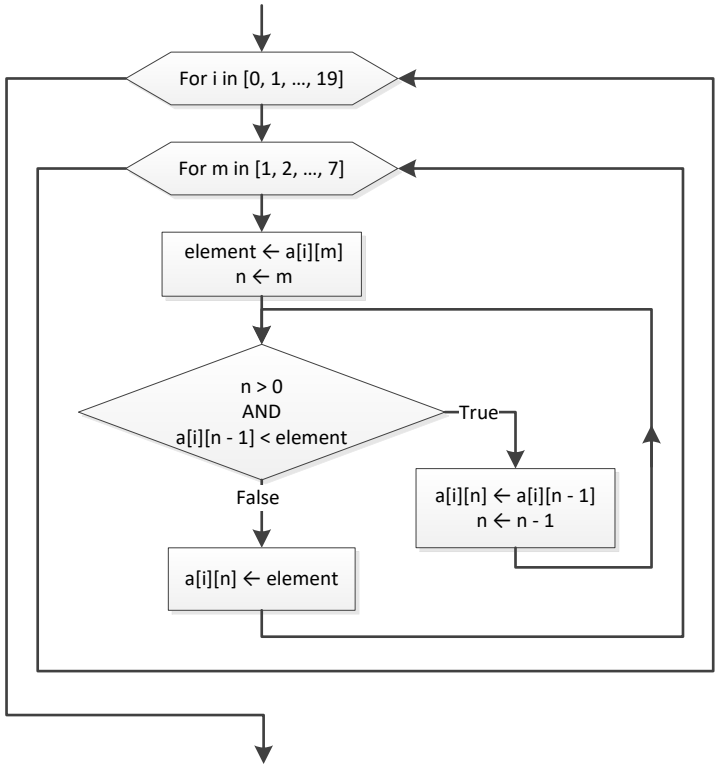


19. Solution

```
PEOPLE = 10
PUZZLES = 8

names = [None] * PEOPLE
times = [ [None] * PUZZLES for i in range(PEOPLE) ]
for i in range(PEOPLE):
    names[i] = input()
    for j in range(PUZZLES):
        times[i][j] = float(input())

for i in range(PEOPLE):
    for m in range(PUZZLES):
        minimum = times[i][m]
        index_of_min = m
        for n in range(m, PUZZLES):
            if times[i][n] < minimum:
                minimum = times[i][n]
                index_of_min = n
        times[i][m], times[i][index_of_min] = times[i][index_of_min], times[i][m]
```

```
for i in range(PEOPLE):
    print(names[i])
    for j in range(3):
        print(times[i][j])

                                    # Or you can do the following:
average = [None] * PEOPLE           # average = []
for i in range(PEOPLE):             # for row in times:
    average[i] = 0                  #     average.append(math.fsum(row) / PUZZLES)
    for j in range(PUZZLES):        #
        average[i] += times[i][j]   #
    average[i] /= PUZZLES           #

for m in range(PEOPLE):
    minimum = average[m]
    index_of_min = m
    for n in range(m, PEOPLE):
        if average[n] < minimum:
            minimum = average[n]
            index_of_min = n
    average[m], average[index_of_min] = average[index_of_min], average[m]
    names[m], names[index_of_min] = names[index_of_min], names[m]

print(names[0], names[1], names[2])
```

20. Solution

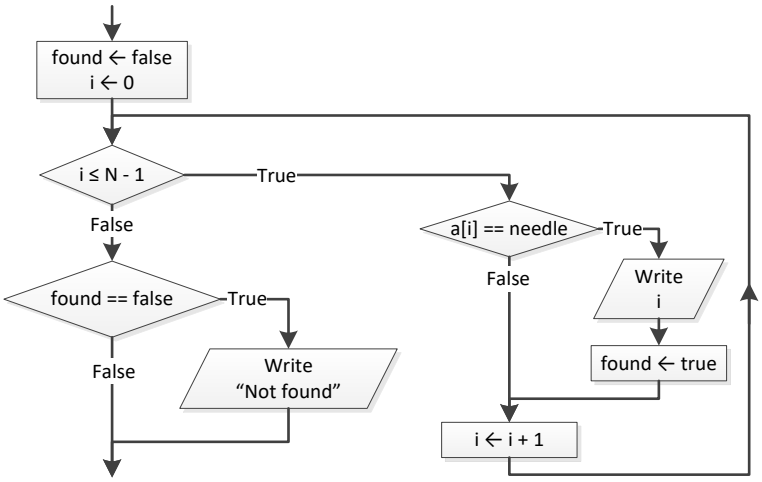


21. Solution

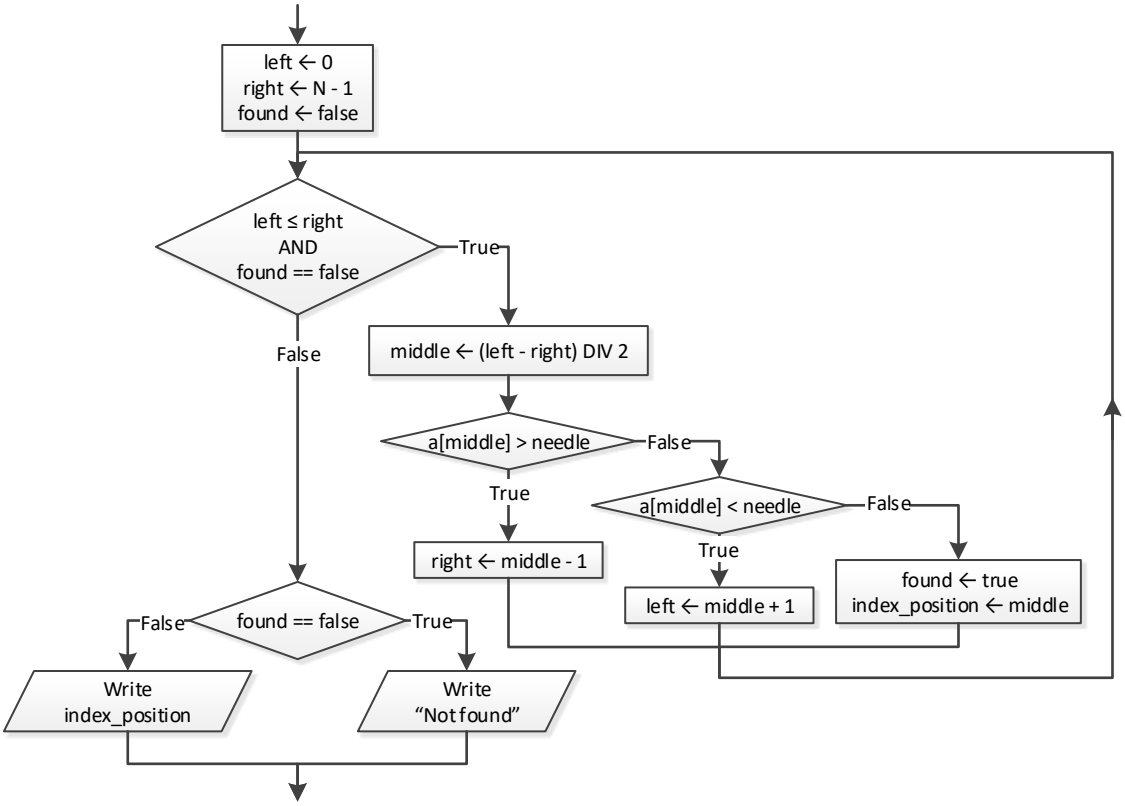
```
CITIES = 5
HOURS = 48

names = [None] * CITIES
CO2 = [ [None] * HOURS for i in range(CITIES) ]
for i in range(CITIES):
    names[i] = input()
    for j in range(HOURS):
        CO2[i][j] = float(input())

                                          # Or you can do the following:
average_per_hour = [None] * CITIES        # average_per_hour = []
for i in range(CITIES):                   # for row in CO2:
    average_per_hour[i] = 0               #     average_per_hour.append(math.fsum(row) / HOURS)
    for j in range(HOURS):                #
        average_per_hour[i] += CO2[i][j]  #
    average_per_hour[i] /= HOURS          #

for i in range(CITIES):
    print(names[i], average_per_hour[i])

average_per_city = [None] * HOURS
for j in range(HOURS):
    average_per_city[j] = 0
    for i in range(CITIES):
        average_per_city[j] += CO2[i][j]
    average_per_city[j] /= CITIES

for j in range(HOURS):
    print(average_per_city[j])

maximum = average_per_city[0]
m_j = 0
for j in range(1, HOURS):
    if average_per_city[j] > max:
        maximum = average_per_city[j]
        m_j = j
print(m_j)

maximum = CO2[0][0]
m_i = 0
m_j = 0
for i in range(CITIES):
    for j in range(HOURS):
        if CO2[i][j] > max:
            maximum = CO2[i][j]
            m_i = i
            m_j = j
print(m_j, names[m_i])

for m in range(1, CITIES):
```

```
    element_1 = average_per_hour[m]
    element_2 = names[m]

    n = m
    while n > 0 and average_per_hour[n - 1] < element_1:
        average_per_hour[n] = average_per_hour[n - 1]
        names[n] = names[n - 1]
        n -= 1

    average_per_hour[n] = element_1
    names[n] = element_2

print(names[0], names[1],  names[2])
```

22. Solution



23. Solution



24. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

TEAMS = 10
GAMES = 16

names = [None] * TEAMS
goals_scored = [ [None] * GAMES for i in range(TEAMS) ]
goals_let_in = [ [None] * GAMES for i in range(TEAMS) ]
for i in range(TEAMS):
    names[i] = input("Enter team name: ")
    for j in range(GAMES):
        inp = input("Enter goals scored: ")
        while not re.match(IS_NUMERIC, inp) or int(inp) < 0:
            inp = input("Error! Enter goals scored: ")
        goals_scored[i][j] = int(inp)

        inp = input("Enter goals let in: ")
        while not re.match(IS_NUMERIC, inp) or int(inp) < 0:
            inp = input("Error! Enter goals let in: ")
        goals_let_in[i][j] = int(inp)
```

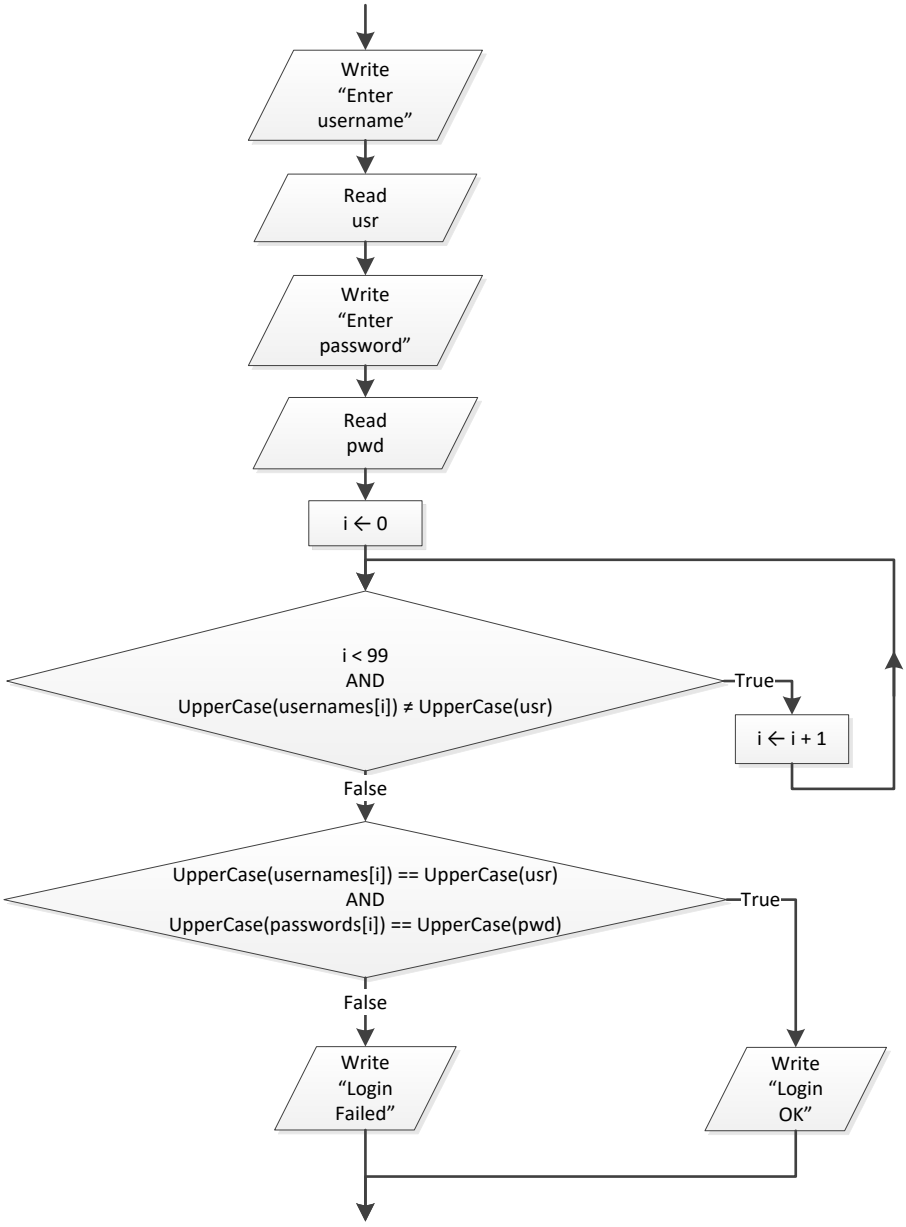
```
needle = input("Enter a team to search: ")

i = 0
while i < TEAMS - 1 and names[i] != needle:
    i += 1

if names[i] != needle:
    print("This team does not exist")
else:
    total = 0
    for j in range(GAMES):
        if goals_scored[i][j] > goals_let_in[i][j]:
            total += 3
        elif goals_scored[i][j] == goals_let_in[i][j]:
            total += 1
    print(total)
```

25. Solution

```
CLASS1 = 20
CLASS2 = 25

print("Class A")
names1 = [None] * CLASS1
for i in range(CLASS1):
    names1[i] = input("Enter name: ")

print("Class B")
names2 = [None] * CLASS2
for i in range(CLASS2):
    names2[i] = input("Enter name: ")

#Insertion sort algorithm
for m in range(1, CLASS1):
    element = names1[m]
    n = m
    while n > 0 and names1[n - 1] > element:
        names1[n] = names1[n - 1]
        n -= 1
    names1[n] = element
for m in range(1, CLASS2):
    element = names2[m]
    n = m
    while n > 0 and names2[n - 1] > element:
        names2[n] = names2[n - 1]
        n -= 1
    names2[n] = element

print("\nClass A")
for i in range(CLASS1):
    print(names1[i])
print("\nClass B")
for i in range(CLASS2):
    print(names2[i])
```

```
needle = input("Enter a name to search: ")

left = 0
right = CLASS1 - 1
found = False
while left <= right and found == False:
    middle = (left + right) // 2

    if names1[middle] > needle:
        right = middle - 1
    elif names1[middle] < needle:
        left = middle + 1
    else:
        found = True

if found == True:
    print("Student found in class No 1")
else:
    left = 0
    right = CLASS2 - 1
    while left <= right and found == False:
        middle = (left + right) // 2

        if names2[middle] > needle:
            right = middle - 1
        elif names2[middle] < needle:
            left = middle + 1
        else:
            found = True

    if found == True:
        print("Student found in class No 2")
    else:
        print("Student not found in either class")
```

26. Solution




```
usr = input("Enter username: ")
pwd = input("Enter password: ")

i = 0
while i < 99 and usernames[i].upper() != usr.upper():
    i += 1

if usernames[i].upper() == usr.upper() and passwords[i].upper() == pwd.upper():
    print("Login OK!")
else:
    print("Login Failed!")
```

27. Solution

```
value_str = input("Enter a value to search: ")
found = False

#Check if entered value is a valid nine-digit SSN
if re.match(IS_NUMERIC, value_str) and value_str >= 100000000 and value_str <= 999999999:
    value = int(value_str)
    i = 0
    while i < 999 and SSNs[i] != value:
        i += 1

    if SSNs[i] == value:
        found = True
        print(names[i])
else:
    for i in range(1000):
        if names[i] == value_str:
            print(names[i])
            found = True

if found == False:
    print("This value does not exist")
```

28. Solution

```
STUDENTS = 12
LESSONS = 6

grades = [ [None] * LESSONS for i in range(STUDENTS) ]
for i in range(STUDENTS):
    for j in range(LESSONS):
        grades[i][j] = int(input())

                                    # Or you can do the following:
average = [None] * STUDENTS         # average = []
for i in range(STUDENTS):           # for row in grades:
    average[i] = 0                  #     average.append(math.fsum(row) / LESSONS)
    for j in range(LESSONS):        #
        average[i] += grades[i][j]  #
    average[i] /= LESSONS           #

found = False
for i in range(STUDENTS):
    if average[i] < 70:
        found = True
        break

if found == True:
    print("There is at least one student that has an average value below 70")
```

Chapter 36

36.4 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 37

37.5 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

37.6 Review Exercises

1. Solution

```
def find_max(a, b):
    if a > b:
        maximum = a
    else:
        maximum = b
    return maximum
```

2. Solution

Step	Statement	Main Code		Function sum_digits()		
		s	i	a	d1	d2
1	s = 0	**0**	?			
2	i = 25	0	**25**			
3	s += sum_digits(i)			**25**	?	?
4	d1 = a % 10			25	**5**	?
5	d2 = a // 10			25	5	**2**
6	return d1 + d2	**7**	25			
7	i = 26	7	**26**			
8	s += sum_digits(i)			**26**	?	?
9	d1 = a % 10			26	**6**	?
10	d2 = a // 10			26	6	**2**
11	return d1 + d2	**15**	26			
12	i = 27	15	**27**			

13	`s += sum_digits(i)`			**27**	?	?
14	`d1 = a % 10`			27	**7**	?
15	`d2 = a // 10`			27	7	**2**
16	`return d1 + d2`	**24**	27			
17	`print(s)`	24 is displayed				

3. Solution

Step	Statement	Main Code		Function sss()		
		s	i	a	total	k
1	`i = 1`	?	**1**			
2	`s = 0`	**0**	1			
3	`while i < 6:`	True				
4	`if i % 2 == 1:`	True				
5	`s += 1`	**1**	1			
6	`i += 1`	1	**2**			
7	`while i < 6:`	True				
8	`if i % 2 == 1:`	False				
9	`s += sss(i)`			**2**	?	?
10	`total = 0`			2	**0**	?
11	`k = 1`			2	0	**1**
12	`total += k`			2	**1**	1
13	`k = 2`			2	1	**2**
14	`total += k`			2	**3**	2
15	`return total`	**4**	2			
16	`i += 1`	4	**3**			
17	`while i < 6:`	True				
18	`if i % 2 == 1:`	True				
19	`s += 1`	**5**	3			
20	`i += 1`	5	**4**			
21	`while i < 6:`	True				
22	`if i % 2 == 1:`	False				
23	`s += sss(i)`			**4**	?	?
24	`total = 0`			4	**0**	?
25	`k = 1`			4	0	**1**
26	`total += k`			4	**1**	1

27	`k = 2`			4	1	**2**
28	`total += k`			4	**3**	2
29	`k = 3`			4	3	**3**
30	`total += k`			4	**6**	4
31	`k = 4`			4	6	**4**
32	`total += k`			4	**10**	4
33	`return total`	**15**	4			
34	`i += 1`	15	**5**			
35	`while i < 6:`	True				
36	`if i % 2 == 1:`	True				
37	`s += 1`	**16**	5			
38	`i += 1`	16	**6**			
39	`while i < 6:`	False				
40	`print(s)`	16 is displayed				

4. Solution

Step	Statement	Main Code				Function `custom_div()`	
		k	**m**	**a**	**x**	**b**	**d**
1	`k = int(input())`	**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 (b + d) // 2`	12	2	1	**4**		
8	`print(m, a, x)`	2 1 4 is displayed					
9	`a += 2`	12	2	**3**	4		
10	`m += 1`	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 (b + d) // 2`	12	3	3	**9**		
15	`print(m, a, x)`	3 3 9 is displayed					
16	`a += 2`	12	3	**5**	9		

17	`m += 1`	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 (b + d) // 2`	12	4	5	**13**		
22	`print(m, a, x)`	4 5 13 is displayed					
23	`a += 2`	12	4	**7**	13		
24	`m += 1`	12	**5**	7	13		
25	`while a < 6:`	False					

5. Solution

```
def my_round(x):
    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

```
def find_min(a, b):
    minimum = a
    if b < minimum:
        minimum = b
    return minimum

#Main code starts here
print("Enter four numbers: ")
x1 = float(input())
x2 = float(input())
x3 = float(input())
x4 = float(input())

                                # Or you can do the following
temp1 = find_min(x1, x2)        #
temp2 = find_min(x3, x4)        #
print(find_min(temp1, temp2))   # print(find_min(find_min(x1, x2), find_min(x3, x4)))
```

7. Solution

```
def Kelvin_to_Fahrenheit(kelvin):
    return 1.8 * kelvin - 459.67

def Kelvin_to_Celsius(kelvin):
    return kelvin - 273.15
```

```
#Main code starts here
k = float(input("Enter a temperature in degrees Kelvin: "))
print("Fahrenheit:", Kelvin_to_Fahrenheit(k))
print("Celsius:", Kelvin_to_Celsius(k))
```

8. Solution

```
import re
IS_NUMERIC = "^[-+]?\\d+(\\.\\d+)?$"

def bmi(w, h):
    b = w * 703 / h ** 2
    if b < 16:
        return_value = "You must add weight."
    elif b < 18.5:
        return_value = "You should add some weight."
    elif b < 25:
        return_value = "Maintain your weight."
    elif b < 30:
        return_value = "You should lose some weight."
    else:
        return_value = "You must lose weight."

    return return_value

#Main code starts here
inp = input("Enter your weight (in pounds): ")
while not re.match(IS_NUMERIC, inp) or float(inp) < 0:
    inp = input("Error! Enter your weight (in pounds): ")
weight = float(inp)

inp = input("Enter your age: ")
while not re.match(IS_NUMERIC, inp) or int(inp) < 18:
    inp = input("Error! Enter your age: ")
age = int(inp)

inp = input("Enter your height (in inches): ")
while not re.match(IS_NUMERIC, inp) or float(inp) < 0:
    inp = input("Error! Enter your height (in inches): ")
height = float(inp)

print(bmi(weight, height))
```

Chapter 38

38.5 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

38. 6 Review Exercises

1. Solution

Step	Statement	Main Code		Function display()
		i	x	a
1	i = 0	**0**	?	
2	x = int(input())	0	**3**	
3	display(x)			**3**
4	if a % 2 == 0:			False
5	print(a, "is odd")	The message “3 is odd” is displayed		
6	i = 1	**1**	3	
7	x = int(input())	1	**7**	
8	display(x)			**7**
9	if a % 2 == 0:			False
10	print(a, "is odd")	The message “7 is odd” is displayed		
11	i = 2	**2**	7	
12	x = int(input())	2	**9**	
13	display(x)			**9**
14	if a % 2 == 0:			False
15	print(a, "is odd")	The message “9 is odd” is displayed		
16	i = 3	**3**	9	
17	x = int(input())	3	**2**	
18	display(x)			**2**
19	if a % 2 == 0:			True
20	print(a + " is even")	The message “2 is even” is displayed		
21	i = 4	**4**	2	

22	`x = int(input())`	4	**4**	
23	`display(x)`			**4**
24	`if a % 2 == 0:`			True
25	`print(a, "is even")`	The message “4 is even” is displayed		

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 = b // a`			30	**0**
6	`print(a * b)`	0 is displayed			
7	`x = 4 * y`	**120**	30		
8	`y += 1`	120	**31**		
9	`while x % y < 30:`	True			
10	`division(y, x)`			**31**	**120**
11	`b = b // a`			31	**3**
12	`print(a * b)`	93 is displayed			
13	`x = 4 * y`	**124**	31		
14	`y += 1`	124	**32**		
15	`while x % y < 30:`	True			
16	`division(y, x)`			**32**	**124**
17	`b = b // a`			32	**3**
18	`print(a * b)`	96 is displayed			
19	`x = 4 * y`	**128**	32		
20	`y += 1`	128	**33**		
21	`while x % y < 30:`	True			
22	`division(y, x)`			**33**	**128**
23	`b = b // a`			33	**3**
24	`print(a * b)`	99 is displayed			
25	`x = 4 * y`	**132**	33		
26	`y += 1`	132	**34**		

27	`while x % y < 30:`	False		

3. Solution

Step	Statement	Main Code		Function `calculate()`		
		i	m	n	s	j
1	`i = 0`	**0**	?			
2	`m = int(input())`	0	**2**			
3	`calculate(m)`			**2**	?	?
4	`s = 0`			2	**0**	?
5	`j = 2`			2	0	**2**
6	`s = s + j ** 2`			2	**4**	2
7	`j = 4`			2	4	**4**
8	`s = s + j ** 2`			2	**20**	4
9	`print(s)`	20 is displayed				
10	`i = 1`	**1**	2			
11	`m = int(input())`	1	**3**			
12	`calculate(m)`			**3**	?	?
13	`s = 0`			3	**0**	?
14	`j = 2`			3	0	**2**
15	`s = s + j ** 2`			3	**4**	2
16	`j = 4`			3	4	**4**
17	`s = s + j ** 2`			3	**20**	4
18	`j = 6`			3	20	**6**
19	`s = s + j ** 2`			3	**56**	6
20	`print(s)`	56 is displayed				
21	`i = 2`	**2**	3			
22	`m = int(input())`	2	**4**			
23	`calculate(m)`			**4**	?	?
24	`s = 0`			4	**0**	?
25	`j = 2`			4	0	**2**
26	`s = s + j ** 2`			4	**4**	2
27	`j = 4`			4	4	**4**
28	`s = s + j ** 2`			4	**20**	4
29	`j = 6`			4	20	**6**

30	`s = s + j ** 2`			4	**56**	6
31	`j = 8`			4	56	**8**
32	`s = s + j ** 2`			4	**120**	8
33	`print(s)`	120 is displayed				

4. Solution

```
def maximum(a, b, c, d, e):
    m = a
    if b > m:
        m = b
    if c > m:
        m = c
    if d > m:
        m = d
    if e > m:
        m = e
    print(m)
```

5. Solution

```
def num_of_days(year, month):
    if month in [4, 6, 9, 11]:
        days = 30
    elif month == 2:
        if year % 4 == 0 and year % 100 != 0 or year % 400 == 0:
            days = 29
        else:
            days = 28
    else:
        days = 31
    print(days)

#Main code starts here
y = int(input("Enter a year: "))
for m in range(1, 13):
    num_of_days(y, m)
```

6. Solution

```
def display_menu():
    print()
    print("1. Convert meters to miles")
    print("2. Convert miles to meters")
    print("3. Exit")
    print("Enter a choice: ")

def meters_to_miles(meters):
    print(meters, "meters equals", (meters / 1609.344), "miles")

def miles_to_meters(miles):
```

```
    print(miles, "miles equals", (miles * 1609.344), "meters")

#Main code starts here
while True:
    display_menu()

    choice = int(input())

    if choice == 3:
        print("Bye!")
    else:
        distance = float(input("Enter distance: "))
        if choice == 1:
            meters_to_miles(distance)
        else:
            miles_to_meters(distance)
    if choice == 3: break
```

7. Solution

```
def amount_to_pay(seconds):
    if seconds <= 600:
        extra = 0
    elif 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

    print("Total amount to pay:", total)

#Main code starts here
seconds = int(input("Enter number of seconds: "))
amount_to_pay(seconds)
```

Chapter 39

39.9 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

39.10 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
1	`k = int(input())`	?	?	**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	`print(m, a, x)`	1 1 2 is displayed					
8	`a += 2`	**3**	1	12	2		

9	`m += 1`	3	**2**	12	2		
10	`a, m = swap(a, m)`					**3**	**2**
11	`x, y = y, x`					**2**	**3**
12	`while a < 8:`	**2**	**3**	12	2		
		True					
13	`if k % m != 0:`	False					
14	`x = a + m + int(a - m)`	2	3	12	**4**		
15	`print(m, a, x)`	3 2 4 is displayed					
16	`a += 2`	**4**	3	12	4		
17	`m += 1`	4	**4**	12	4		
18	`a, m = swap(a, m)`					**4**	**4**
19	`x, y = y, x`					**4**	**4**
20	`while a < 8:`	**4**	**4**	12	4		
		True					
21	`if k % m != 0:`	False					
22	`x = a + m + int(a - m)`	4	4	12	**8**		
23	`print(m, a, x)`	4 4 8 is displayed					
24	`a += 2`	**6**	4	12	8		
25	`m += 1`	6	**5**	12	8		
26	`a, m = swap(a, m)`					**6**	**5**
27	`x, y = y, x`					**5**	**6**
28	`while a < 8:`	**5**	**6**	12	8		
		True					
29	`if k % m != 0:`	False					
30	`x = a + m + int(a - m)`	5	6	12	**10**		
31	`print(m, a, x)`	6 5 10 is displayed					
32	`a += 2`	**7**	6	12	10		
33	`m += 1`	7	**7**	12	10		
34	`a, m = swap(a, m)`					**7**	**7**
35	`x, y = y, x`					**7**	**7**
36	`while a < 8:`	**7**	**7**	12	10		
		True					
37	`if k % m != 0:`	True					
38	`x = a % m`	7	7	12	**0**		
39	`m, a = swap(m, a)`					**7**	**7**

40	`x, y = y, x`					**7**	**7**
41	`print(m, a, x)`	**7**	**7**	12	0		
		7 7 0 is displayed					
42	`a += 2`	**9**	7	12	0		
43	`m += 1`	9	**8**	12	0		
44	`a, m = swap(a, m)`					**9**	**8**
45	`x, y = y, x`					**8**	**9**
46	`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

```
STUDENTS = 10
LESSONS = 5

def part1(names, grades):
    for i in range(STUDENTS):
        names[i] = input("Enter name for student No." + str(i + 1) + ": ")
        for j in range(LESSONS):
            grades[i][j] = input("Enter grade for lesson No." + str(j + 1) + ": ")

def part2(grades):
    average = [None] * STUDENTS

    for i in range(STUDENTS):
        average[i] = 0
        for j in range(LESSONS):
            average[i] += grades[i][j]
        average[i] /= LESSONS
    return average

def part3(average, names):
    for m in range(1, STUDENTS):
        for n in range(STUDENTS - 1, m - 1, -1):
            if average[n] > average[n - 1]:
                average[n], average[n - 1] = average[n - 1], average[n]
                names[n], names[n - 1] = names[n - 1], names[n]
```

```
            elif average[n] == average[n - 1]:
                if names[n].CompareTo(names[n - 1]) < 0:
                    names[n], names[n - 1] = names[n - 1], names[n]

#Main code starts here
names = [None] * STUDENTS
grades = [ [None] * LESSONS for i in range(STUDENTS) ]

part1(names, grades)
average = part2(grades)
part3(average, names)

for i in range(STUDENTS):
    print(names[i], "\t", average[i])
```

9. Solution

```
def part1():
    message = input("Enter a message: ").lower()
    return message

def part2(message):
    message_clean = ""
    for i in range(len(message)):
        if message[i] not in " ,.?":
            message_clean += message[i]
    return message_clean

def part3(message_clean):
    middle_pos = (len(message_clean) - 1) // 2
    j = len(message_clean) - 1
    palindrome = True
    for i in range(middle_pos + 1):
        if message_clean[i] != message_clean[j]:
            palindrome = False
            break
        j -= 1
    return palindrome

def part4(message):
    message_clean = part2(message)
    palindrome = part3(message_clean)
    return palindrome

#Main code starts here
message = part1()
palindrome = part4(message)
if palindrome == True:
    print("The message is palindrome")
```

10. Solution

```
a = int(input())
```

```
b = int(input())
c = int(input())
d = input()

maximum = a
if b > maximum:
    maximum = b
if c > maximum:
    maximum = c
if d > maximum:
    maximum = d

print(maximum)
```

11. Solution

```
def f1(a, b, c):
    total = a + b + c
    average = total / 3
    return total, average
```

12. Solution

```
def my_round(x, decimal_places = 2):
    digit_to_check = x * 10 ** (decimal_places + 1) % 10
    if digit_to_check >= 5:
        return_value = int(x * 10 ** decimal_places + 1) / 10 ** decimal_places
    else:
        return_value = int(x * 10 ** decimal_places) / 10 ** decimal_places
    return return_value
```

13. Solution

```
def get_input():

    while True:
        answer = input("Enter Yes or No: ").upper()
        if answer == "YES" or answer == "NO": break
    return answer

def find_area(b, h):
    return b * h

#Main code starts here
while True:
    bas = float(input("Enter the base of the parallelogram: "))
    height = float(input("Enter the height of the parallelogram: "))

    print("Area =", find_area(bas, height))

    print("Would you like to repeat? ")
    if get_input() == "NO": break
```

14. Solution

```
STUDENTS = 100

def get_lists(names, grades):
    for i in range(STUDENTS):
        names[i] = input("Enter name: ")
        grades[i] = int(input("Enter grade: "))

def get_average(grades):
    total = 0                   # Or you can do the following:
    for i in range(STUDENTS):   #
        total += grades[i]      #
    return total / STUDENTS     # return math.fsum(grades) / STUDENTS

def sort_lists(grades, names):
    for m in range(1, STUDENTS):
        element_grds = grades[m]
        element_nms = names[m]

        n = m
        while n > 0 and grades[n - 1] > element_grds:
            grades[n] = grades[n - 1]
            names[n] = names[n - 1]
            n -= 1

        grades[n] = element_grds
        names[n] = element_nms

#Main code starts here
names = [None] * STUDENTS
grades = [None] * STUDENTS
get_lists(names, grades)
average = get_average(grades)
sort_lists(grades, names)
for i in range(STUDENTS):
    if grades[i] < average:
        print(names[i])
```

15. Solution

First Approach

```
JUDGES = 10

def get_list():
    score = [None] * JUDGES
    for i in range(JUDGES):
        score[i] = int(input("Judge No" + str(i + 1) + ". Enter score: "))
    return score

def find_min_max(score):
    minimum = score[0]
    maximum = score[0]
```

```
    for i in range(1, JUDGES):
        if score[i] > maximum:
            maximum = score[i]
        if score[i] < minimum:
            minimum = score[i]

    return minimum, maximum

#Main code starts here
name = input("Enter artist's name: ")
score = get_list()
minimum, maximum = find_min_max(score)

total = 0
for i in range(JUDGES):
    total += score[i]

points = total - minimum - maximum
print("Artist", name, "got", points, "points")
```

Second Approach

```
import math
JUDGES = 10

def get_list():
    score = [None] * JUDGES
    for i in range(JUDGES):
        score[i] = int(input("Judge No" + str(i + 1) + ". Enter score: "))
    return score

def find_min_max(score):
    return min(score), max(score)

#Main code starts here
name = input("Enter artist's name: ")
score = get_list()
minimum, maximum = find_min_max(score)

points = math.fsum(score) - minimum - maximum
print("Artist", name, "got", points, "points")
```

16. Solution

```
def woc(index):
    if index == 1:
        return_value = 1
    else:
        return_value = 2 * woc(index - 1)
    return return_value

#Main code starts here
total = 0
for i in range(1, 65):
    total += woc(i)
```

```
print(total)
```

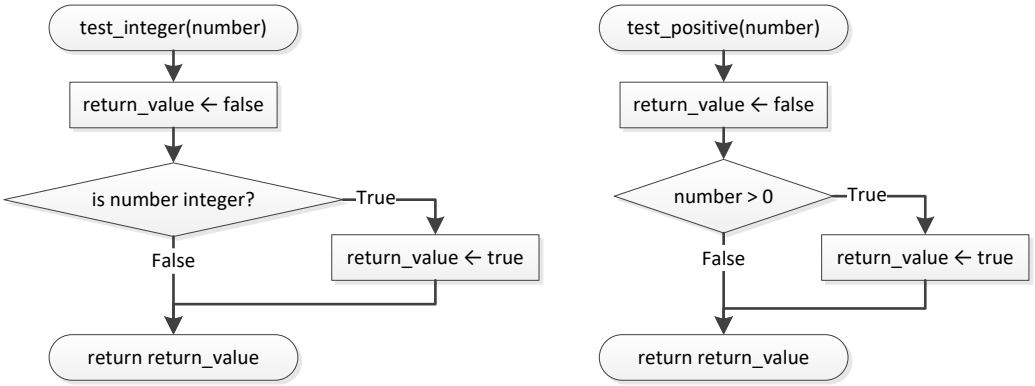
17. Solution

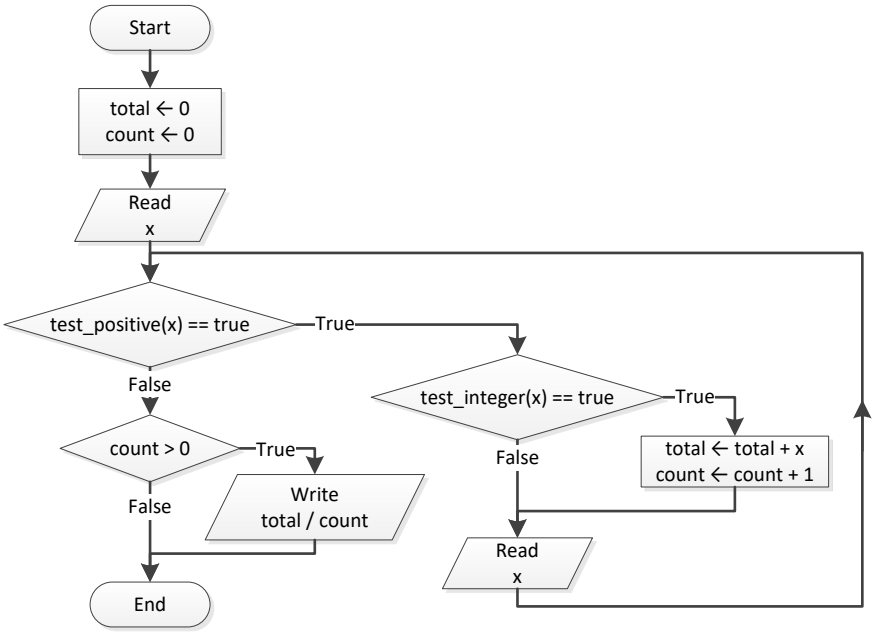
```
import math
def fact(value):
    if value == 1:
        return_value = 1
    else:
        return_value = value * fact(value - 1)

    return return_value

def my_cos(x, i = 40):
    if i == 0:
        return_value = 1
    else:
        return_value = my_cos(x, i - 4) + x ** i / fact(i) - x ** (i - 2) / fact(i - 2)

    return return_value

#Main code starts here
print(my_cos(math.pi / 4))
```

Chapter 40

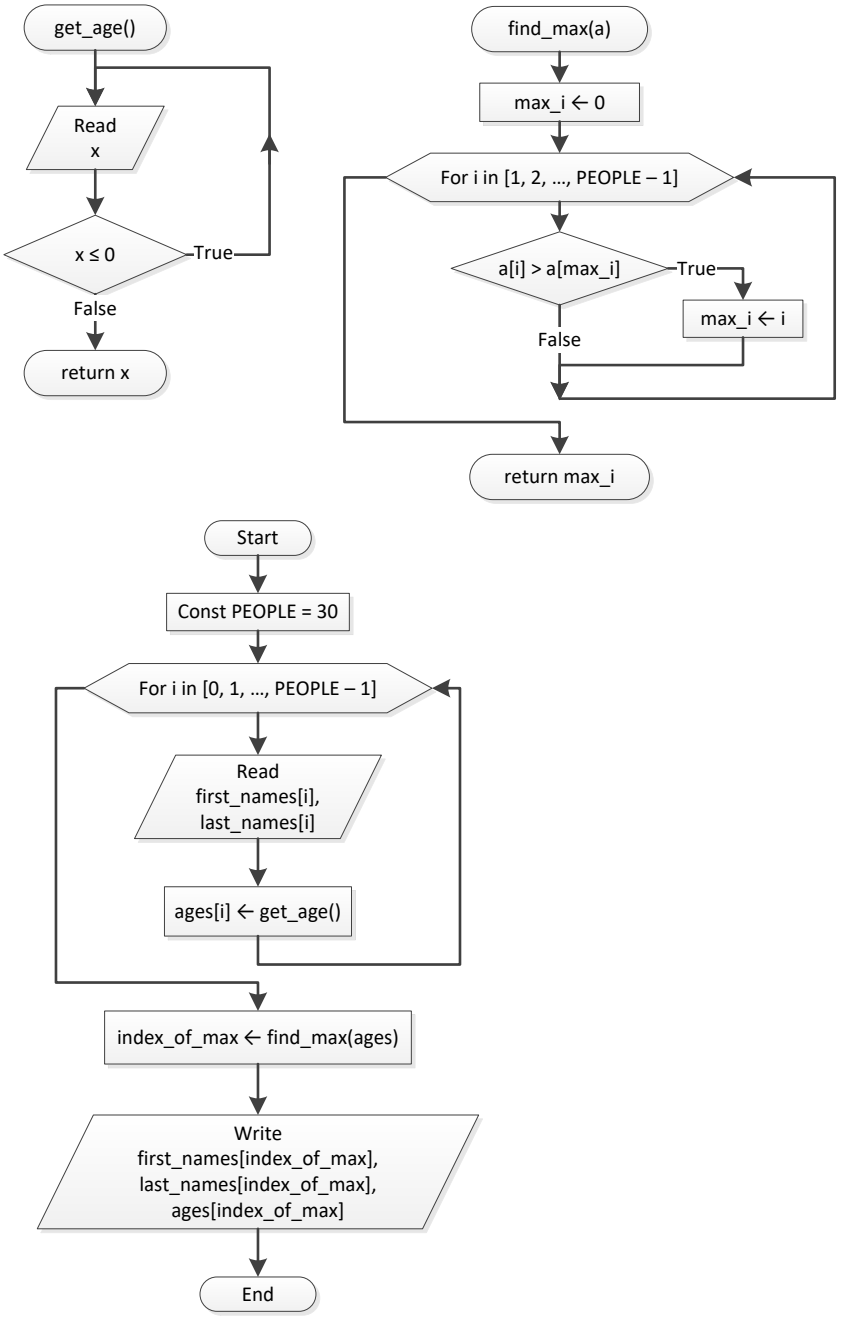
40.4 Review Exercises

1. Solution

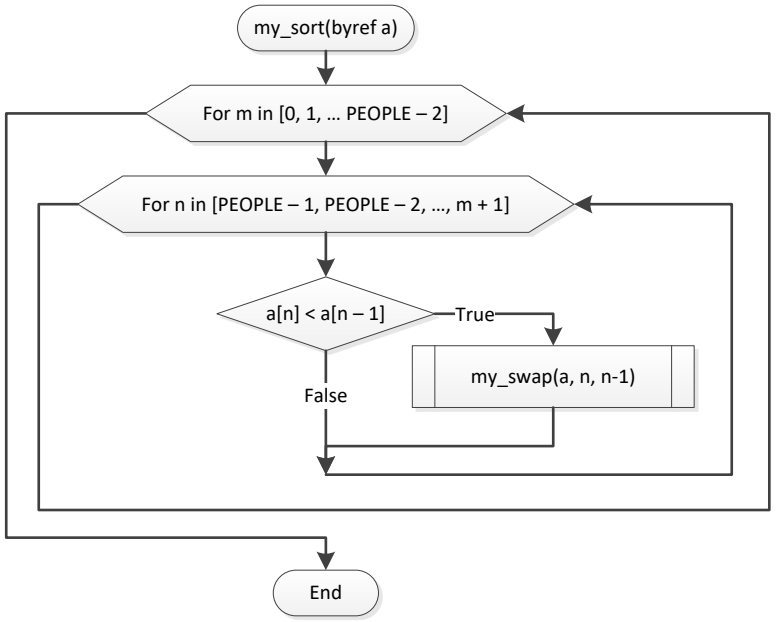


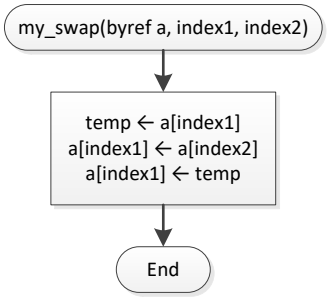


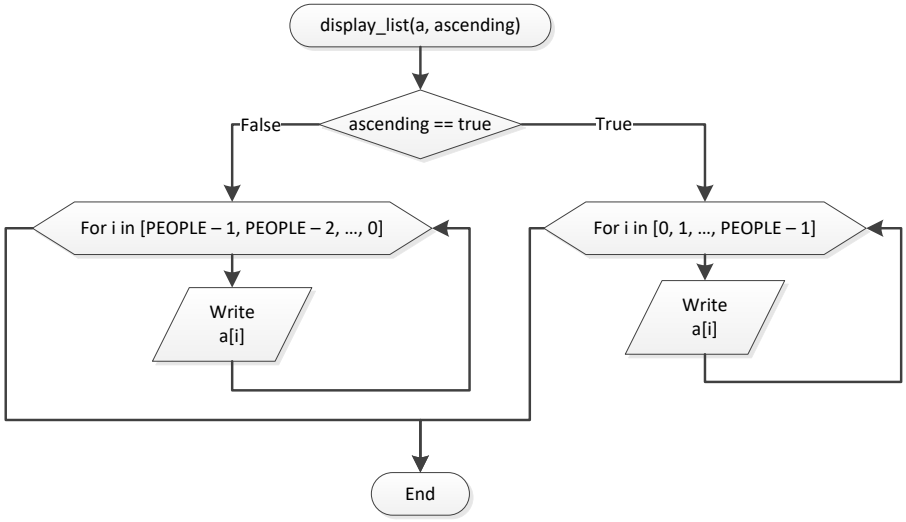
2. Solution

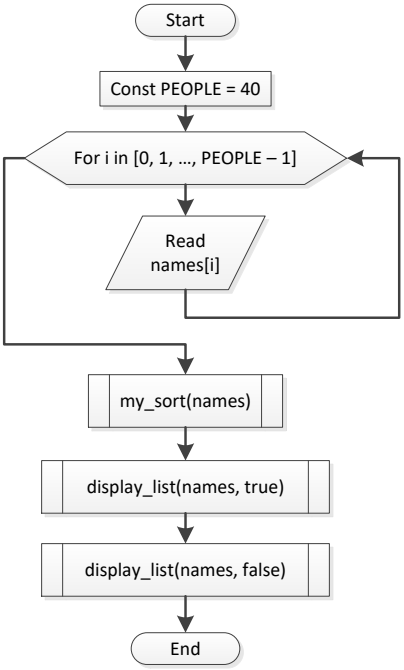

get_age()
Read
x
x ≤ 0
True
False
return x
find_max(a)
max_i ← 0
For i in [1, 2, …, PEOPLE – 1]
a[i] > a[max_i]
True
max_i ← i
False
return max_i
Start
Const PEOPLE = 30
For i in [0, 1, …, PEOPLE – 1]
Read
first_names[i],
last_names[i]
ages[i] ← get_age()
index_of_max ← find_max(ages)
Write
first_names[index_of_max],
last_names[index_of_max],
ages[index_of_max]
End

3. Solution

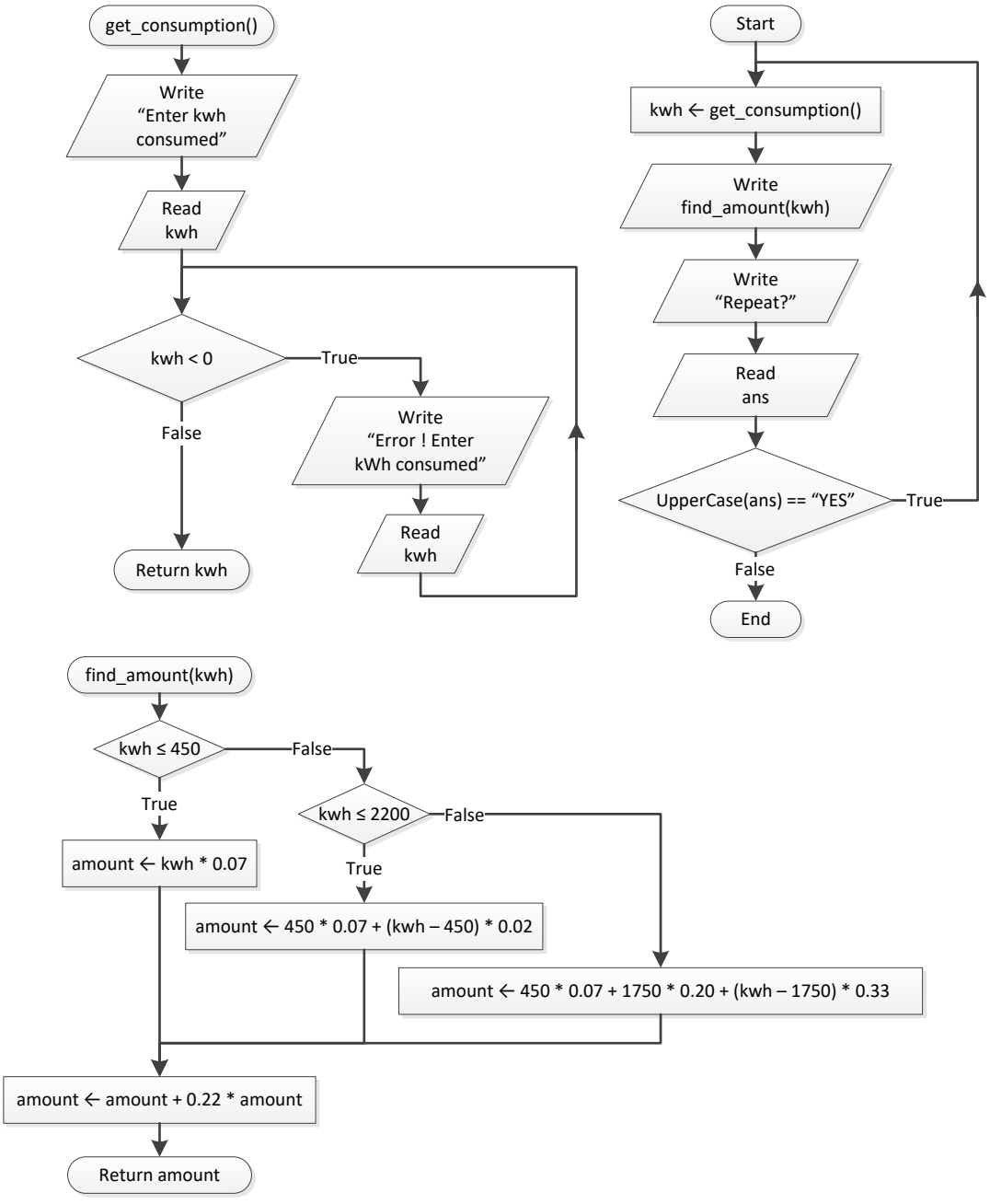

my_sort(byref a)
For m in [0, 1, … PEOPLE – 2]
For n in [PEOPLE – 1, PEOPLE – 2, …, m + 1]
a[n] < a[n – 1]
True
my_swap(a, n, n-1)
False
End


my_swap(byref a, index1, index2)
temp ← a[index1]
a[index1] ← a[index2]
a[index1] ← temp
End


display_list(a, ascending)
ascending == true
False
True
For i in [PEOPLE – 1, PEOPLE – 2, …, 0]
Write
a[i]
For i in [0, 1, …, PEOPLE – 1]
Write
a[i]
End


Start
Const PEOPLE = 40
For i in [0, 1, …, PEOPLE – 1]
Read
names[i]
my_sort(names)
display_list(names, true)
display_list(names, false)
End

4. Solution



5. Solution

```
STUDENTS = 20
LESSONS = 10

def get_lists(names, grades):
    for i in range(STUDENTS):
        names[i] = input()
        for j in range(LESSONS):
```

```
            grades[i][j] = int(input())

def find_average(grades):
    average = [None] * STUDENTS

    for i in range(STUDENTS):
        average[i] = 0
        for j in range(LESSONS):
            average[i] += grades[i][j]
        average[i] /= LESSONS
    return average

def display(names, average):
    for i in range(STUDENTS):
        if average[i] > 89:
            print(names[i], average[i])

#Main code starts here
names = [None] * STUDENTS
grades = [ [None] * LESSONS for i in range(STUDENTS) ]

get_lists(names, grades)
av = find_average(grades)
display(names, av)
```

6. Solution

```
def fib(n):
    if n == 0 or n == 1:
        return_val = n
    else:
        return_val = fib(n - 1) + fib(n - 2)

    return return_val

#Main code starts here
while True:
    n = int(input())
    while n < 0:
        n = int(input("Error"))

    print(fib(n))
    ans = input("Again? ")
    if ans != "Y": break
```

Chapter 41

41.3 Review Exercises

1. Solution

```
import math
ACCURACY = 0.000000001

def factorial(n):
    return_value = 1
    for i in range(1, n + 1):
        return_value *= i
    return return_value

def my_sin(x):
    sign = 1
    sinus = 0
    i = 1
    while True:
        sinus_previous = sinus
        sinus += sign * x ** i / factorial(i)

        sign = -sign
        i += 2
        if abs(sinus - sinus_previous) <= ACCURACY: break
    return sinus

def degrees_to_rad(degrees):
    return 2 * math.pi * degrees / 360

#Main code starts here
for i in range(361):
    print("sin(", i, ") ~= ", my_sin(degrees_to_rad(i)), sep = "")
```

2. Solution

```
def is_leap(year):
    return_value = False
    if year % 4 == 0 and year % 100 != 0 or year % 400 == 0:
        return_value = True
    return return_value

def num_of_days(year, month):
    if month in [4, 6, 9, 11]:
        days = 30
    elif month == 2:
        if is_leap(year) == True:
            days = 29
        else:
            days = 28
    else:
```

```
        days = 31

    return days

def check_date(day, month, year):
    return_value = True
    if month not in range(1, 13):
        return_value = False
    elif day < 1 or day > num_of_days(year, month):
        return_value = False
    return return_value

#Main code starts here
day = int(input("Enter day: "))
month = int(input("Enter month: "))
year = int(input("Enter year: "))
while check_date(day, month, year) == False:
    print("Error!")
    day = int(input("Enter day: "))
    month = int(input("Enter month: "))
    year = int(input("Enter year: "))

total = 0
for i in range(1, month):
    total += num_of_days(year, i)
total += day

print(total)
```

3. Solution

```
import random

def dice():
    return random.randrange(1, 7)

#Main code starts here
names = [None] * 2
names[0] = input("Player1 - Enter name: ")
names[1] = input("Player2 - Enter name: ")

for player in range(2):
    total = 0
    for i in range(10):
        print("Player " + names[player] + ", hit enter to roll the dice!")
        key = input()

        dice1 = dice()
        dice2 = dice()
        print(dice1, dice2)
        total += dice1 + dice2
    if player == 0:
        total_player1 = total
    else:
```

```
        total_player2 = total

if total_player1 == total_player2:
    print("Tie!")
elif total_player1 > total_player2:
    print(names[0], " wins")
else:
    print(names[1], " wins")
```

4. Solution

```
GAS = 1
DIESEL = 2
HYBRID = 3
TAX_RATE = 0.1
CARS = 40

def get_choice():
    print("1. Gas")
    print("2. Diesel")
    print("3. Hybrid")
    choice = int(input("Enter type of the car: "))
    return choice

def get_days():
    days = int(input("Enter total number of rental days: "))
    return days

def get_charge(car_type, rental_days):

    if car_type == GAS:
        if rental_days <= 5:
            charge = rental_days * 24
        elif rental_days <= 8:
            charge = 5 * 24 + (rental_days - 5) * 22
        else:
            charge = 5 * 24 + 3 * 22 + (rental_days - 8) * 18
    elif car_type == DIESEL:
        if rental_days <= 5:
            charge = rental_days * 28
        elif 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
        elif 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
```

```
#Main code starts here
rented_car_types = [None] * CARS
rented_days = [None] * CARS

for i in range(CARS):
    rented_car_types[i] = get_choice()
    rented_days[i] = get_days()

total = 0
for i in range(CARS):
    charge = get_charge(rented_car_types[i], rented_days[i])
    print("Car No", (i + 1), ":", charge)
    total += charge

count = 0
for i in range(CARS):
    if rented_car_types[i] == HYBRID:
        count += 1

print("Hybrids rented:", count)
print("Net profit:", total / (1 + TAX_RATE))
```

5. Solution

```
CHANNELS = 10
DAYS = 7

def get_data(names, viewers):
    day_names = ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"]

    for i in range(CHANNELS):
        names[i] = input("Enter name for channel No." + str(i + 1) + ": ")
        for j in range(DAYS):
            viewers[i][j] = int(input("Enter the number of viewers of the main news program on " +   \
                                      day_names[j] + " for channel " + names[i] + ": "))

def get_average(a):
    total = 0           # Or you can do the following
    for i in range(5):  #
        total += a[i]   #
    return total / 5    # return math.fsum(total) / 5

#Main code starts here
names = [None] * CHANNELS
viewers = [ [None] * DAYS for i in range(CHANNELS) ]
get_data(names, viewers)

for i in range(CHANNELS):
    weekend = (viewers[i][DAYS - 2] + viewers[i][DAYS - 1]) / 2
    if weekend >= 1.2 * get_average(viewers[i]):    # viewers[i] represents the whole row
        print(names[i])

for i in range(CHANNELS):
    increasing = True
```

```
    for j in range(1, DAYS):
        if viewers[i][j] <= viewers[i][j - 1]:
            increasing = False
    if increasing == True:
        print(names[i])
```

6. Solution

```
CITIZENS = 300

def input_data(SSNs, answers):
    for i in range(CITIZENS):
        SSNs[i] = int(input("Enter SSN: "))
        answers[i] = input("Enter answer: ")

def sort_lists(SSNs, answers):
    for m in range(CITIZENS):
        minimum = SSNs[m]
        index_of_min = m
        for n in range(m, CITIZENS):
            if SSNs[n] < minimum:
                minimum = SSNs[n]
                index_of_min = n
        SSNs[m], SSNs[index_of_min] = SSNs[index_of_min], SSNs[m]
        answers[m], answers[index_of_min] = answers[index_of_min], answers[m]

def search_list(SSNs, SSN):
    left = 0
    right = CITIZENS - 1
    found = False
    while left <= right and found == False:
        middle = (left + right) // 2

        if SSNs[middle] > SSN:
            right = middle - 1
        elif SSNs[middle] < SSN:
            left = middle + 1
        else:
            found = True
            index_position = middle

    if found == False:
        print("SSN not found!")
        return_value = -1
    else:
        return_value = index_position
    return return_value

def count_answers(answers, answer):

    count = 0
    for i in range(CITIZENS):
        if answers[i] == answer:
            count += 1
```

```
    return count

#Main code starts here
SSNs = [None] * CITIZENS
answers = [None] * CITIZENS

while True:
    input_data(SSNs, answers)
    sort_lists(SSNs, answers)

    SSN = int(input("Enter an SSN to search: "))

    index = search_list(SSNs, SSN)
    if index != -1:
        answer = answers[index]
        print(answer)

        count = count_answers(answers, answer)
        print(count * 100 / CITIZENS)
    answer = input("Repeat? ")
    if answer != "yes": break
```

7. Solution

```
TEAMS = 8
GAMES = 12

def input_data(names, results):
    for i in range(TEAMS):
        names[i] = input("Enter team name: ")
        for j in range(GAMES):
            results[i][j] = input("Enter result (W, L, T): ")

def display_result(names, results):
    result = input("Enter a result to search (W, L, T): ")
    for i in range(TEAMS):
        print("Team:", names[i])
        found = False
        for j in range(GAMES):
            if results[i][j] == result:
                print("Week:", (j + 1))
                found = True
        if found == False:
            print("nothing found")

def find_team(names):
    name = input("Enter a name to search: ")

    i = 0
    while i < TEAMS - 1 and names[i] != name:
        i += 1

    if names[i] != name:
        return_value = -1
```

```
    else:
        return_value = i
    return return_value

#Main code starts here
names = [None] * TEAMS
results = [ [None] * GAMES for i in range(TEAMS) ]

input_data(names, results)
display_result(names, results)

index = find_team(names)
while index != -1:
    total = 0
    for j in range(GAMES):
        if results[index][j] == "W":
            total += 3
        elif results[index][j] == "T":
            total += 1
    print("Points:", total)
    index = find_team(names)
```

8. Solution

```
import string
alphabet = " " + string.ascii_lowercase     #space is a valid character!

def my_encrypt(message, encryption_key):
    return_value = ""
    for i in range(len(message)):
        letter = message[i]
        index = alphabet.find(letter)
        new_index = (index + encryption_key) % 27 #26 letters + 1 space
        new_letter = alphabet[new_index]
        return_value += new_letter
    return return_value

def my_decrypt(message, decryption_key):
    return_value = ""
    for i in range(len(message)):
        letter = message[i]
        index = alphabet.find(letter)
        new_index = (index + 27 - decryption_key) % 27 #26 letters + 1 space
        new_letter = alphabet[new_index]
        return_value += new_letter
    return return_value

def display_menu():
    print()
    print("1. Encrypt a message")
    print("2. Decrypt a message")
    print("3. Exit")

#Main code starts here
```

```
while True:
    display_menu()
    choice = int(input("Enter a choice: "))

    if choice == 1:
        message = input("Enter a message to encrypt: ")
        encryption_key = int(input("Enter an encryption key: "))
        print("Your encrypted message is:", my_encrypt(message, encryption_key))
    elif choice == 2:
        message = input("Enter a message to decrypt: ")
        decryption_key = int(input("Enter an decryption key: "))
        print("Your decrypted message is:", my_decrypt(message, decryption_key))
    if choice == 3: break
```

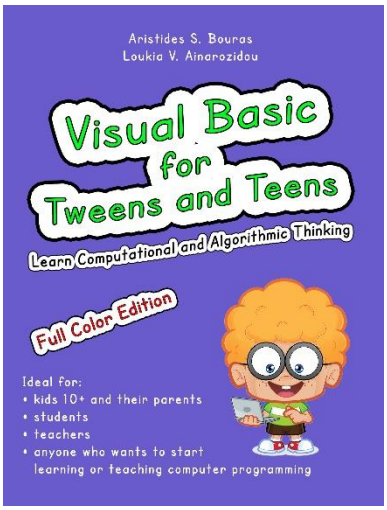
Some Final Words from the Authors

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.

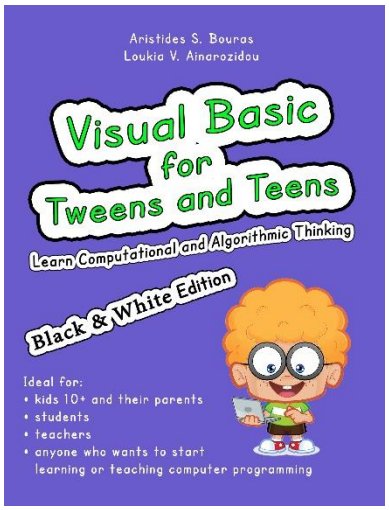
So if you liked this book, please visit the web store where you bought it and show us your gratitude by writing a good review and giving us as many stars as possible. By doing this, you will encourage us to continue writing and of course you'll help other readers to reach us.

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

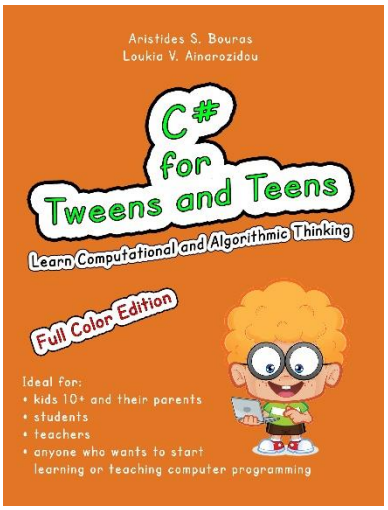
More of our Books



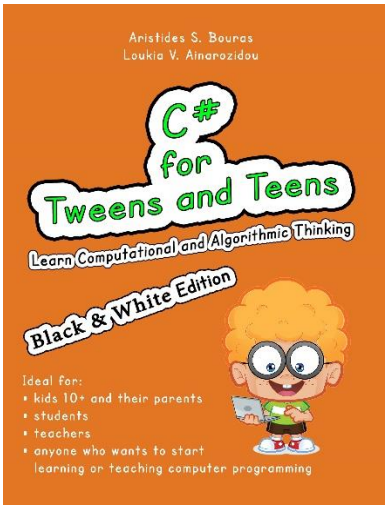
ISBN-10: 1982083670
ISBN-13: 978-1982083670



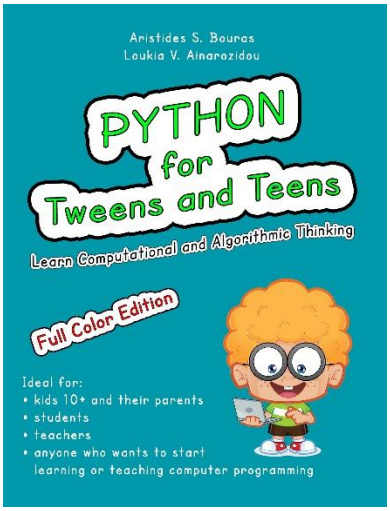
ISBN-10: 1982083697
ISBN-13: 978-1982083694



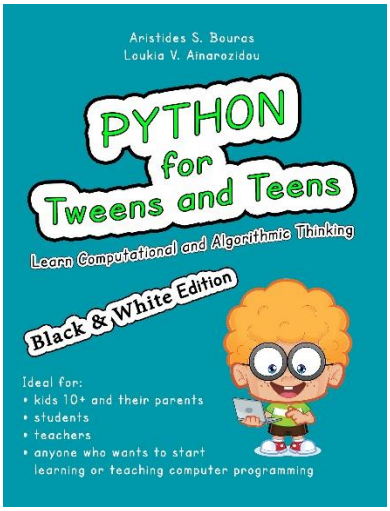
ISBN-10: 1973727684
ISBN-13: 978-1973727682



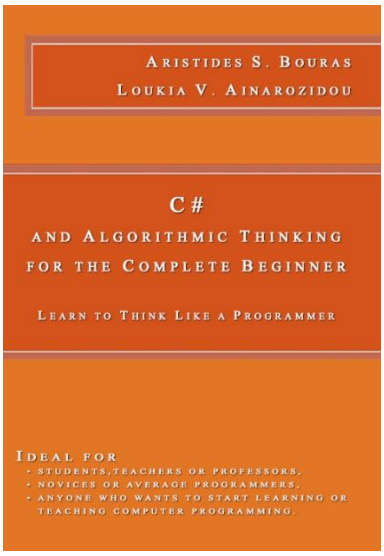
ISBN-10: 1973727765
ISBN-13: 978-1973727767



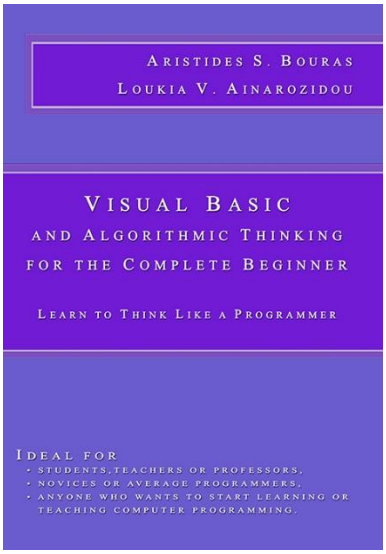
ISBN-10: 1543127940
ISBN-13: 978-1543127942



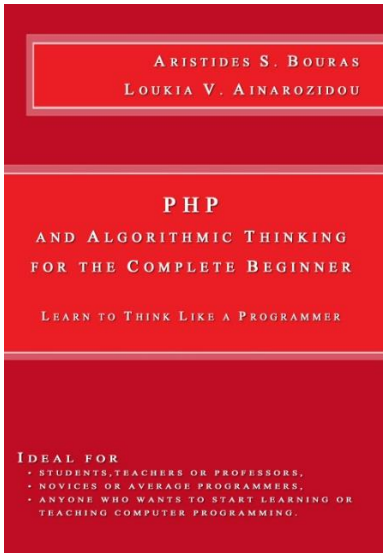
ISBN-10: 1546611215
ISBN-13: 978-1546611219



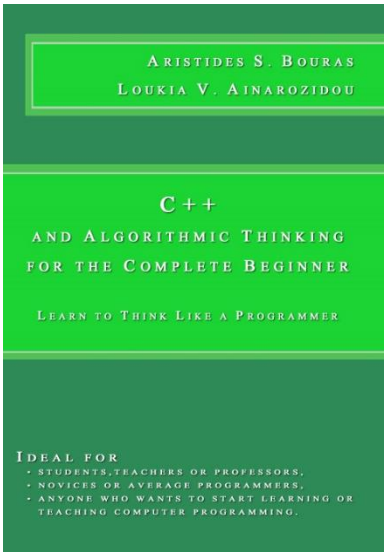
ISBN-10: 1508952485
ISBN-13: 978-1508952480



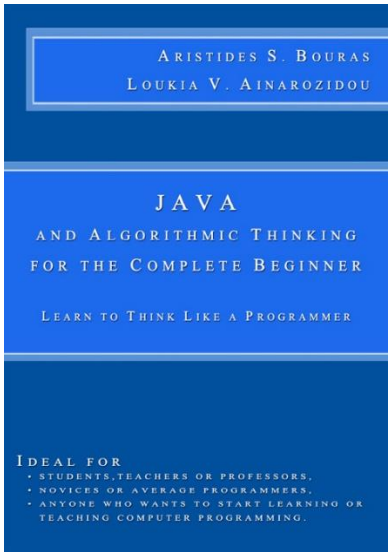
ISBN-10: 1511798963
ISBN-13: 978-1511798969



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



ISBN-10: 1508577552
ISBN-13: 978-1508577553



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