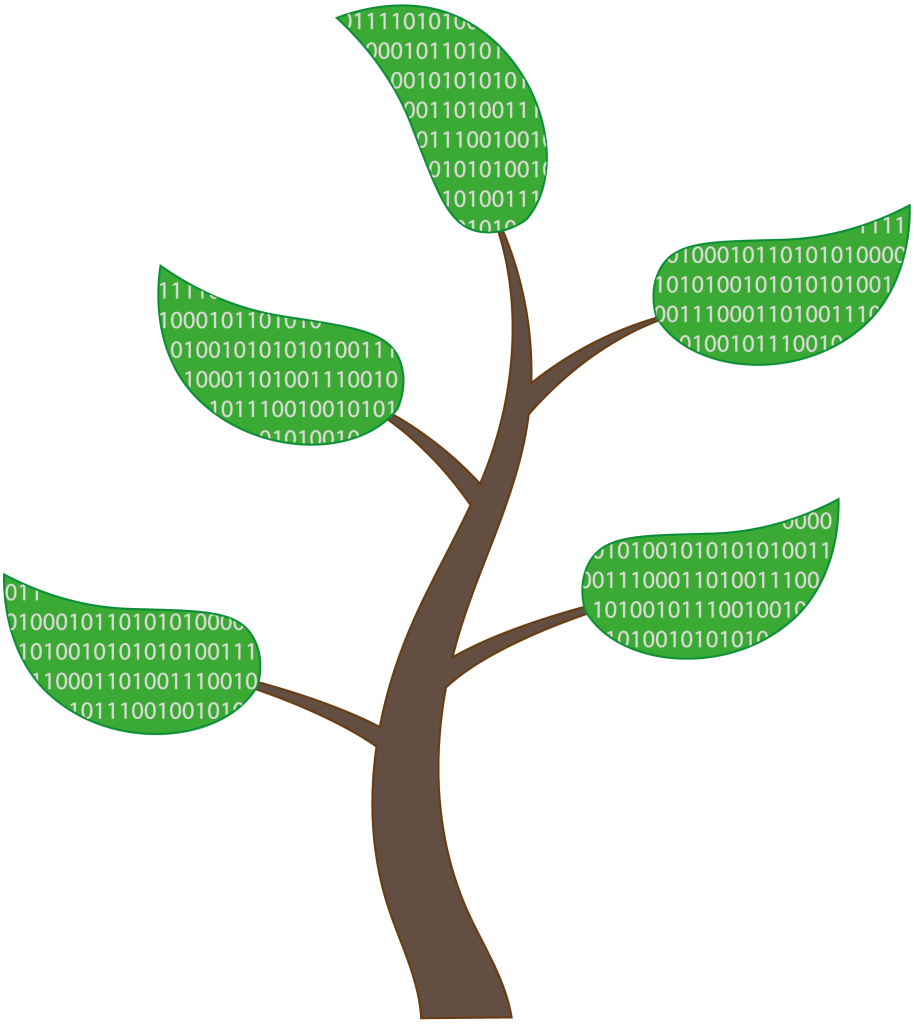

Programming Exercises for Beginners with examples

PART **0**

By Alexander Tyshchenko
http://tyshchenko.pro

November 16, 2022

Contents

About author

Alexander Tyshchenko is a consulting engineer with over 10 years of experience in software development. He also **Oracle Java Certified Programmer** and **Individual Member** at jcp.org (Java Community Process).

Introduction

Who is this book for

This book is not for reading but for acting

What is covered in this book

What you need to use this book

Conventions

1

Basics

1. Write a program that calculate:

Arithmetic expression 1

$$0.1 + 0.1$$

Arithmetic expression 2

$$0.1 + 0.2$$

Arithmetic expression 3

$$0.3 - 0.1$$

Arithmetic expression 4

$$0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1$$

Arithmetic expression 5

$$0.1 * 6$$

Arithmetic expression 6

$$0.1 * 3$$

Arithmetic expression 7

$$0.1 * 2$$

2. Write a program that calculate hals the sum of two variables (in two formulas).

Formula 1:

$$\frac{a+b}{2}$$

Formula 2:

$$\frac{a}{2}+\frac{b}{2}$$

Input data:

```
case 1:  a = 4, b = 8
case 2:  a = 1073741824, b = 1073741833
```

3. Write a program that display on a console your **name** and **age**

4. Write a program that create an integer variable with value **0** and increase that value to **1** and display on console

5. Write a program that create an integer variable with value **3** and subtract **1** from that value and display on console

6. Write a program that create two variables (type float point) **a = 1.1** and **b= 2.3**:

- display on a console multiplication of the variables
- display on a console a sum of the variables
- display on a console a difference of the variables

7. Write a program that do the following code instructions:

- create an integer variable
- assign value **3** to the created integer variable
- add **10** to the existing variable
- print variable value on console
- subtract **2** from the variable
- print variable value on console

8. Write a program that do the following code instructions:

- create an integer variable with assigning value **21**

- create another integer variable with assigning value **11**
- add to the first variable **1**
- subtract **3** from the second variable
- print on console first and second variable values

9. Write a program that do the following code instructions:

- create three **integer** type variables
- assign all variables by value **11**
- create fourth variable
- add all three variable values and assign result to the fourth variable
- display result on console

10. Write a program that do the following code instructions:

- create three **integer** type variables with names - **a, b, c**
- assign variable **a** with value **10**, variable **b** with value **100**, variable **c** with value **201**
- create fourth variable with name **d**
- assign the fourth variable with result of arithmetic equation $(c-b)+a$
- display on console value of variable **d**

11. Write a program that exchange values between two variables:

- create two **integer** variables
- assign with random numbers (random means by you)
- create third variable and assign with values that hold first variable
- assign second variable with values that hold first variable
- assign first variable with values that hold third variable
- print on console values of first ans second variables

12. Write a program that add two variables

- define a function that take two **integer** variables and return they sum
- call previously defined function with parameters values **345** and **6**
- print result on console

13. Write a program that subtract one variable from other:

- define a function that take two **integer** variables and return they subtraction result
- call previously defined function with parameters values **123** and **111**
- print result on console

14. Write a program that multiplying two variables:

- define a function that take two **integer** variables and return they multiplication result
- call previously defined function with parameters values **10** and **11**
- print result on console

15. Write a program that divide two variables:

- define a function that take two **integer** variables and return they division result
- call previously defined function with parameters values **12** and **4**
- print result on console

16. Write a program that do the following code instructions:

- define a function that take three **integer** type parameters
- function must return arithmetic equation $(parameter1 * parameter2)/parameter3$
- call function with values **123**, **324** and **30**
- print result on console

17. Write a program that find **max** values of three variables

18. Write a program that generate random number in range from **1** to **10** and check if generated value more than **5** print **"yes"** otherwise print **"no"**

19. Write a program that check if **10** does not belong to the interval $[49, 101]$

20. Write a program that check if **2** belong to the interval $[1, 6]$

21. Write a program that convert **1000000** kilobytes into megabytes (1 megabyte = 1024 kilobytes)

22. Write a program that convert **15** miles into kilometers (1 mile = 0,6214 kilometer)

23. Write a program that convert **78** Fahrenheit temperature into celsius $C = \dfrac{F - 32}{1.8}$

24. User enter from keyboard value of variable **a** and **b**, the program must decide and show on console result (if **a** more than **b**, or if **a** less than **b**, or **a** equal to **b**)

Example 1

```
> Enter a:
> 3
> Enter b:
> 1
> Result:
> a is more than b
```

Example 2

```
> Enter a:
> 2
> Enter b:
> 5
> Result:
> a is less than b
```

Example 3

```
> Enter a:
> 7
> Enter b:
> 7
> Result:
> a is equal to b
```

25. Write a program that print **day of week name** using **switch case**

26. Write a program that check if a number even or odd by using **switch case**

27. User is enter on a console a number from **21** to **65** than a program must print magical square.

n - 20	1	12	7
11	8	n-21	2
5	10	3	n - 18
4	n - 19	6	9

28. Write a program that calculate perimeter of rectangle with sides A (40cm) and B (100cm)

$$P = 2 \cdot (A + B)$$

29. Write a program that calculate area of rectangle with sides A (5cm) and B (500cm)

$$S = A \cdot B$$

30. Write a program that display on console result of

$$3^{10}$$

31. Write a program that display on console result of

$$\sqrt{100}$$

32. Write a program that display on console result of

$$\sqrt{10}$$

(show only integer part)

33. Write a program that calculate

$$e^3$$

where e - is Euler's number

34. Write a program that display on console result of

$$\ln 6$$

35. Write a program that convert **3** radians into degrees

$$radian = \frac{180}{\pi}$$

36. Write a program that calculate 5! (do not use loop or recursion)

$$n! = 1 \cdot 2 \cdot 3 \cdot ...n$$

37. Write a program that display on console result of

$$\log_{10} 123$$

38. Write a program that display on console result of

$$\sin(90^\circ)$$

39. Write a program that display on console result of

$$\cos(30°)$$

40. Write a program that display on console result of

$$\tan(30°)$$

41. Write a program that display on console result of

$$ctg(25°)$$

42. Write a program that multiply number **9898989898989** and number **9898989898989** (in different programming languages will be different results)

43. Write a program that find **the geometric mean**

$$10, 51.2, 8, 6$$

For example

`the geometric mean 2 and 18 ->` $\sqrt{2 * 18}$
`the geometric mean 2 and 18 and 3 ->` $\sqrt[3]{2 * 18 * 3}$

44. Write a program that calculate area of triangle with sides **A = 4, B = 5 , C = 6** (to calculate area of triangle use Heron formula)

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

where

$$p = \frac{a+b+c}{2}$$

45. Write a program that calculate area of triangle in cartesian coordinate system **A(0,0) B (5, 6) C(0, 10)**

$$S = \frac{1}{2}|(x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1)|$$

46. Write a program that calculate area and perimeter of a circle with radius

$$r = 10$$

Area

$$S = \pi \cdot r^2$$

Perimeter

$$P = 2\pi \cdot r$$

47. Write a program that calculate

$$\frac{\sqrt{6,3\cdot 1,7}\left(\frac{6,3}{1,7}-\frac{1,7}{6,3}\right)}{\sqrt{(6,3+1,7)^2-4\cdot 6,3\cdot 1,7}}$$

48. Write a program that find **x** from the quadratic equation

$$x^2-3x+3=0$$

Discriminant of the quadratic equation

$$D=b^2-4ac \quad (ax^2+bx+c=0)$$

$$x_{1,2}=\frac{-b\pm\sqrt{b^2-4ac}}{2a}$$

49. Write a program that generate a random number from **10** to **25** including.

50. Write a program that find a vector length

$$\vec{a}=(10,15)$$

for example

$$\vec{b}=(x,y)$$

length $|\vec{b}|=\sqrt{x^2+y^2}$

51. Write a program that find a vector length

$$\vec{c}=(54,100,1000)$$

for example

$$\vec{d}=(x,y,z)$$

length $|\vec{d}|=\sqrt{x^2+y^2+z^2}$

52. Write a program that find a distance between two points on a two-dimensional plane

$$A(1.2,4.5)\ and\ B(6.5,3.8)$$

for examples $A(x_a,y_a)$ $B(x_b,y_b)$

$$|AB|=\sqrt{(x_b-x_a)^2+(y_b-y_a)^2}$$

53. Write a program that find a distance between two points on a three-dimensional plane

$$A(5.7, 0, -45) \ and \ B(100, -1, 38)$$

for examples $A(x_a, y_a, z_a)$ $B(x_b, y_b, z_b)$

$$|AB| = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2 + (z_b - z_a)^2}$$

54. Write a program that evaluate your year of birth from your age

55. Write a program that add two complex numbers $z_1 = 1 - 33i$ and $z_2 = 5 + 6i$. A sum of two complex numbers

$$z = z_1 + z_2$$

where

$$z_1 = a_1 + b_1 i$$

$$z_2 = a_2 + b_2 i$$

$$z = (a_1 + a_2) + (b_1 + b_2)i$$

56. Write a program that subtract one complex number from other $z_1 = 2 + 4i$ and $z_2 = 2 + 9i$. A subtraction of two complex numbers

$$z = z_1 - z_2$$

where

$$z_1 = a_1 + b_1 i$$

$$z_2 = a_2 + b_2 i$$

$$z = (a_1 - a_2) + (b_1 - b_2)i$$

57. Write a program that calculate

$$2^{2^2}$$

58. Write a program that calculate

$$3^{3^{3^3}}$$

2

Loops

1. Write a program that create an integer variable and assigns value **10**, use loop to increase value of variable by **2** four times

2. Write a program that print on console all number from **20** to **40** with step **1** (40 must be not included)

3. Write a program that print on console all numbers from **-10** to **10** with step **1**

4. Write a program that create an integer variable with value **100** and decreasing by **1** in every loop iteration until the value of the variable won't reach **50**

5. Write a program that create an integer variable **b** and assigns value **200**, after that decrease the variable value by variable **a** (begin value **1**) in every loop iteration value if the variable a must be increased by **1**, iteration must be stopped when the value of the variable **b** will be less then 0

6. Write a program that multiply an integer variable **c** (initial value **1**) **5** times by **2**

7. Write a program that create an integer variable **d** with value **1** and multiply **10** times by **3**

8. Write a program that print all natural numbers from **1** to **100**

9. Write a program that find all divisors of a number **18**

Example

```
6
> 1 2 3 6 (divisors)
```

10. Write a program that display on console stars in several ways

Output 1

```
> * * * * *
```

Output 2

```
> * * * * *
> * * * * *
```

Output 3

```
> *
> *
> *
> *
> *
```

Output 4

```
> * *
> * *
> * *
> * *
> * *
```

Output 5

```
> *
>
> *
>
>
> *
>
>
>
> *
>
>
>
>
> *
```

Output 6

```
> * * * * *
```

Output 7

```
> * *  *   *    *     *      *
```

Output 8

```
> * * * * *
>  * * * * *
> * * * * *
>  * * * * *
> * * * * *
```

11. Write a program that display a figure on console (use loop)

Output 1

```
> *
> * *
```

Output 2

```
> *
> * *
> * * *
```

Output 3

```
> *
> * *
> * * *
> * * * *
> * * * * *
```

12. Write a program that print on console values of the function $y = \sin 2x$ on interval $[0.2; 0.4]$ with step 0.01

13. Write a program that print on console values of the function $y = \dfrac{2-x}{x^2+x-6}$ on interval $[10; 100]$ with step 1.5

14. Write a program that find first and last digits of the number **229345**

For Example

```
123
> first digit is 1
> last digit is 3
```

15. Write a program that find sum of first and last digits of the number **45678**

For Example

```
987
> sum = 9 + 7 = 16
```

16. Write a program that find multiplication of all digits of the number **876432**

For Example

```
234
> 2 * 3 * 4 = 24
```

17. Write a program that find swap first and last digits of the number **2345**

For Example

```
6524
> 4526
```

18. Write a program that counts the numbers of digits in a number **145797353**

For Example

```
456
> 3
```

19. Write a program that glue number **58** and number **123**

For Example

```
34 and 56
> 3456
```

20. Write a program that find **min** and **max** digits in the number **23456431**

For Example

```
423
> min number 2
> max number 4
```

21. Write a program that sum every even digit in a number **567898**

For Example

```
3467
> 4 + 7 = 11
```

22. Write a program that revers digits in the number **668892645**

For Example

```
3234
> 4323
```

23. Write a program that check if the numbers are palindromes

number 668892645

number 3456543

For Example

```
121
> palindrome

123
> not palindrome
```

24. Write a program that delete all digits from a number

23442554

For Example

```
13245631 (delete digit 3)
> 124561
```

25. Write a program that display on console

Output 1

```
> 1 1 1 1 1 1
> 1 1 1 1 1 1
> 1 1 1 1 1 1
> 1 1 1 1 1 1
> 1 1 1 1 1 1
> 1 1 1 1 1 1
```

Output 2

```
> 1 1 1 1 1 1
> 0 0 0 0 0 0
> 1 1 1 1 1 1
> 0 0 0 0 0 0
> 1 1 1 1 1 1
> 0 0 0 0 0 0
```

Output 3

```
> 0 1 0 1 0 1
> 0 1 0 1 0 1
> 0 1 0 1 0 1
> 0 1 0 1 0 1
> 0 1 0 1 0 1
> 0 1 0 1 0 1
```

Output 4

```
> 1 1 1 1 1 1
> 1 0 0 0 0 1
> 1 0 0 0 0 1
> 1 0 0 0 0 1
> 1 0 0 0 0 1
> 1 1 1 1 1 1
```

Output 5

```
> 1 0 1 1 0 1
> 1 0 1 1 0 1
> 1 0 1 1 0 1
> 1 0 1 1 0 1
> 1 0 1 1 0 1
> 1 0 1 1 0 1
```

Output 6

```
> 1 1 0 0 1 1
> 1 1 0 0 1 1
> 0 0 0 0 0 0
> 0 0 0 0 0 0
> 1 1 0 0 1 1
> 1 1 0 0 1 1
```

Output 7

```
> 1 1 1 1 1 1
> 1 1 1 1 1 1
> 1 1 0 0 1 1
> 1 1 0 0 1 1
> 1 1 1 1 1 1
> 1 1 1 1 1 1
```

Output 8

```
> 1 2 3 4 5 6
> 2 3 4 5 6 7
> 3 4 5 6 7 8
> 4 5 6 7 8 9
> 5 6 7 8 9 10
> 6 7 8 9 10 11
```

Output 9

```
> 1 2 3 4 5 6
> 2 3 4 5 6 6
> 3 4 5 6 6 6
> 4 5 6 6 6 6
> 5 6 6 6 6 6
> 6 6 6 6 6 6
```

26. Write a program that display on console a figure

Output

```
> --|--|--|--|--|--|--|--|--|--|--
```

27. Write a program that display on console a figure

Output

```
> /\/\/\/\/\/\/\/\/\/\
```

28. Write programs that display on console all numbers from **1** to **9** in several different ways

Output program 1

```
> 1 2 3 4 5 6 7 8 9
```

Output program 2

```
> 9 8 7 6 5 4 3 2 1
```

Output program 3

```
> 1
>  2
>   3
>    4
>     5
>      6
>       7
>        8
>         9
```

Hint for program 3

1								
	2							
		3						
			4					
				5				
					6			
						7		
							8	
								9

Output program 4

```
>         1
>        2
>       3
>      4
>     5
>    6
>   7
>  8
> 9
```

Hint for program 4

								1
							2	
						3		
					4			
				5				
			6					
		7						
	8							
9								

Output program 5

```
> 9
>  8
>   7
>    6
>     5
>      4
>       3
>        2
>         1
```

Hint for program 5

9								
	8							
		7						
			6					
				5				
					4			
						3		
							2	
								1

Output program 6

```
>         9
>        8
>       7
>      6
>     5
>    4
>   3
>  2
> 1
```

Hint for program 6

								9
							8	
						7		
					6			
				5				
			4					
		3						
	2							
1								

Output program 7

```
> 1       1
>  2     2
>   3   3
>    4 4
>     5
>    6 6
>   7   7
>  8     8
> 9       9
```

Hint for program 7

1								1
	2						2	
		3				3		
			4		4			
				5				
			6		6			
		7				7		
	8						8	
9								9

Output program 8

```
> 9       9
>  8     8
>   7   7
>    6 6
>     5
>    4 4
>   3   3
>  2     2
> 1       1
```

Hint for program 8

9								9
	8						8	
		7				7		
			6		6			
				5				
			4		4			
		3				3		
	2						2	
1								1

29. Write a program that display a figure

Output

```
>           *
>  *        *        *
>  * *      *      * *
>  *  *     *     *  *
>  *   *    *    *   *
>  *  *   * * *   *  *
>  * *   *     *   * *
>  *    *       *    *
>  *   *    *    *   *
>  * * * * * * * * * *
>       *       *
>           *
```

30. Write a program that display a figure

Output

```
> ***********
> **       **
> * *     * *
> *  *   *  *
> *   * *   *
> *    *    *
> *   * *   *
> *  *   *  *
> * *     * *
> **       **
> ***********
```

31. Write a program that display a figure on console

Output 1

```
> **
>  **
>   **
>    **
>     **
>      **
>       **
```

Output 2

```
>       **
>      **
>     **
>    **
>   **
>  **
> **
```

32. Write programs that display on console all numbers from **2** to **20** with spaces between numbers and step **2**

Output program 1

```
> 2  4  6  8  10  12  14  16  18  20
```

Output program 2

```
> 20  18  16  14  12  10  8  6  4  2
```

33. Write programs that display on console all squared numbers from **1** to **10** space separated

Output program 1

```
> 1  4  9  16  25  36  48  64  82  100
```

Output program 2

```
> 100  82  64  48  36  25  16  9  4  1
```

34. Write programs that display on console all cubes of numbers from **1** to **10** space separated

Output program 1

```
> 1  8  27  64  125  216  343  512  729  1000
```

Output program 2

```
> 1000  729  512  343  215  125  64  27  8  1
```

35. Write a program that display on console a figure

Output

```
>    *****
> ***     ***
> ***     ***
> ***     ***
> ***     ***
>    *****
```

36. Write a program that print numbers in different ways

Output 1

```
> 1
> 2 2
> 3 3 3
> 4 4 4 4
> 5 5 5 5 5
> 6 6 6 6 6 6
> 7 7 7 7 7 7 7
```

Output 2

```
> 7 7 7 7 7 7 7
> 6 6 6 6 6 6
> 5 5 5 5 5
> 4 4 4 4
> 3 3 3
> 2 2
> 1
```

Output 3

```
> 1
> 1 2
> 1 2 3
> 1 2 3 4
> 1 2 3 4 5
> 1 2 3 4 5 6
> 1 2 3 4 5 6 7
```

Output 4

```
> 1 2 3 4 5 6 7
> 1 2 3 4 5 6
> 1 2 3 4 5
> 1 2 3 4
> 1 2 3
> 1 2
> 1
```

Output 5

```
> 1
> 3 3
> 6 6 6
> 10 10 10 10
> 15 15 15 15 15
> 21 21 21 21 21 21
> 28 28 28 28 28 28 28
> 36 36 36 36 36 36 36 36
```

37. Write a program that display all numbers from **1** to **28** in triangle way (every line contains one more numbers than previous, first line is starting from 1)

Output 1

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

Output 1

```
>                   11
>                12 13
>             14 15 16
>          17 18 19 20
>       21 22 23 24 25
>    26 27 28 29 30 31
> 32 33 34 35 36 37 38
```

38. Write a program that display all numbers from **1** to **21** in triangle way

Output 1

```
>         11
>       12   13
>     14   15   16
>   17   18   19   20
> 21   22   23   24   25
```

Output 2

```
>         11
>       13   12
>     16   15   14
>   20   19   18   27
> 25   24   23   22   21
```

39. Write a program that display on console english alphabet as a triangle

Output

```
> a
> b b
> c c c
> d d d d
> e e e e e
> f f f f f f
> g g g g g g g
> h h h h h h h h
> i i i i i i i i i
> j j j j j j j j j j
> k k k k k k k k k k k
> l l l l l l l l l l l l
> m m m m m m m m m m m m m
> n n n n n n n n n n n n n n
> o o o o o o o o o o o o o o o
> p p p p p p p p p p p p p p p p
> q q q q q q q q q q q q q q q q q

                > etc.
```

40. Write a program that display on console symbols as a triangle

Output 1

```
>       A
>      ABA
>     ABCBA
>    ABCDCBA
>   ABCDEDCBA
>  ABCDEFEDCBA
> ABCDEFGFEDCBA
```

Output 2

```
> A
> ABA
> ABCBA
> ABCDCBA
> ABCDEDCBA
> ABCDEFEDCBA
> ABCDEFGFEDCBA
```

41. Write a program that print **pascal** triangle with rows **5**

Output

```
>     1
>    1 1
>   1 2 1
>  1 3 3 1
> 1 4 6 4 1
```

42. Write a program that display on console all ASCII symbols

43. Write a program that display multiplication table for **5**

Output

```
> 5 * 0 = 0;
> 5 * 1 = 5;
> 5 * 2 = 10;
> 5 * 3 = 15;
> 5 * 4 = 20;
> 5 * 5 = 25;
> 5 * 6 = 30;
> 5 * 7 = 35;
> 5 * 8 = 40;
> 5 * 9 = 45;
> 5 * 10 = 50;
```

44. Write a program that display a figure on console (use loop)

Output

```
> * * * * *
> * *     *
> *   *   *
> *     * *
> * * * * *
```

45. Write a program that display a figure on console (use loop)

Output

```
>     * * * * *
>    * * * * *
>   * * * * *
>  * * * * *
> * * * * *
```

Output

```
>     * * * * *
>    *       *
>   *       *
>  *       *
> * * * * *
```

46. Write a program that display rectangle figure on console

Output 1

```
> * * * * * *
> *         *
> *         *
> *         *
> * * * * * *
```

Hint for program 1

*		*		*		*		*		*
*										*
*										*
*										*
*		*		*		*		*		*

47. Write a program that display a figure on console

Output

```
>   *
>  ***
>   *
>  ***
>   *
>  ***
>   *
>  ***
>   *
>  ***
>   *
>  ***
>   *
```

48. Write a program that display a figure on console

Output

```
>     *
>     *
>     *
>     *
>     *
> *********
>     *
>     *
>     *
>     *
>     *
```

49. Write a program that display a figure on console

Output

```
> * * * * * * *
> *
> * * * * * *
> *
> *
> * * * * *
> *
> *
> *
> * * * *
> *
> *
> *
> *
> * * *
> *
> *
> *
> *
> *
> * *
```

50. Write a program that display a figure on console

Output

```
>     *  *  *  *  *  *  *  *  *
>   *****************************
>     *  *  *  *  *  *  *  *  *
>   *****************************
>     *  *  *  *  *  *  *  *  *
>   *****************************
>     *  *  *  *  *  *  *  *  *
```

51. Write a program that display a figure on console

Output

```
> * * * * * * * * * * * * *
> *                       *
> *     Hello, world!     *
> *                       *
> * * * * * * * * * * * * *
```

52. Write a program that display figures on console

Output 1

```
>
>             *
>               *
>                 *
> * * * * * * * * * *
>                 *
>               *
>             *
>
```

Output 2

```
>
>       *
>     *
>   *
> * * * * * * * * * *
>   *
>     *
>       *
>
```

Output 3

```
>       *
>     * * *
>   *   *   *
> *     *     *
>       *
>       *
>       *
>       *
>       *
```

Output 4

```
>       *
>       *
>       *
>       *
>       *
> *     *     *
>   *   *   *
>     * * *
>       *
```

53. Write a program that display a triangle figure in several different ways

Output 1

```
> *
> **
> ***
> ****
> *****
```

Hint for program 1

*				
*	*			
*	*	*		
*	*	*	*	
*	*	*	*	*

Output 2

```
>     *
>    **
>   ***
>  ****
> *****
```

Hint for program 2

				*
			*	*
		*	*	*
	*	*	*	*
*	*	*	*	*

Output 3

```
> *****
>  ****
>   ***
>    **
>     *
```

Hint for program 3

*	*	*	*	*
	*	*	*	*
		*	*	*
			*	*
				*

Output 4

```
> *****
> ****
> ***
> **
> *
```

Hint for program 4

*	*	*	*	*
*	*	*	*	
*	*	*		
*	*			
*				

Output 5

```
> *
> **
> * *
> *  *
> *   *
> *    *
> *     *
> ********
```

Hint for program 5

*							
*	*						
*		*					
*			*				
*				*			
*					*		
*						*	
*	*	*	*	*	*	*	*

Output 6

```
> ********
> *     *
> *    *
> *   *
> *  *
> * *
> **
> *
```

Hint for program 6

*	*	*	*	*	*	*	*
*						*	
*					*		
*				*			
*			*				
*		*					
*	*						
*							

Output 7

```
> ********
>  *     *
>   *    *
>    *   *
>     *  *
>      * *
>       **
>        *
```

Hint for program 7

*	*	*	*	*	*	*	*
	*						*
		*					*
			*				*
				*			*
					*		*
						*	*
							*

Output 8

```
>        *
>       **
>      * *
>     *  *
>    *   *
>   *    *
>  *     *
> ********
```

Hint for program 8

							*
						*	*
					*		*
				*			*
			*				*
		*					*
	*						*
*	*	*	*	*	*	*	*

54. Write a program that display on console a pyramid in several ways

Output 1

```
>    *
>   ***
>  *****
> *******
```

Hint for program 1

			*			
		*	*	*		
	*	*	*	*	*	
*	*	*	*	*	*	*

Output 2

```
> *******
>  *****
>   ***
>    *
```

Hint for program 2

*	*	*	*	*	*	*
	*	*	*	*	*	
		*	*	*		
			*			

Output 3

```
>     *
>    * *
>   *   *
>  *     *
> *********
```

Hint for program 3

				*				
			*		*			
		*				*		
	*						*	
*	*	*	*	*	*	*	*	*

Output 4

```
> *********
>  *     *
>   *   *
>    * *
>     *
```

Hint for program 4

*	*	*	*	*	*	*	*	*
	*						*	
		*				*		
			*		*			
				*				

55. Write a program that display on console a figure

Output 1

```
>    *
>
>   ***
>
>  *****
>
> *******
```

Output 2

```
> *******
>
>  *****
>
>   ***
>
>    *
```

56. Write a program that display on console a figure in several ways

Output 1

```
> *
> **
> ***
> ****
> ***
> **
> *
```

Hint for program 1

*			
*	*		
*	*	*	
*	*	*	*
*	*	*	
*	*		
*			

Output 2

```
>    *
>   **
>  ***
> ****
>  ***
>   **
>    *
```

Hint for program 2

			*
		*	*
	*	*	*
*	*	*	*
	*	*	*
		*	*
			*

Output 3

```
> *
> **
> * *
> *  *
> *   *
> *    *
> *   *
> *  *
> * *
> **
> *
```

Hint for program 4

*					
*	*				
*		*			
*			*		
*				*	
*					*
*				*	
*			*		
*		*			
*	*				
*					

Output 4

```
>      *
>     **
>    * *
>   *  *
>  *   *
> *    *
>  *   *
>   *  *
>    * *
>     **
>      *
```

Hint for program 3

					*
				*	*
			*		*
		*			*
	*				*
*					*
	*				*
		*			*
			*		*
				*	*
					*

57. Write a program that display on console a figure in several ways

Output 1

```
> *********
> *******
> *****
> ***
> *
> ***
> *****
> *******
> *********
```

Output 2

```
>   *********
>     *******
>       *****
>         ***
>           *
>         ***
>       *****
>     *******
>   *********
```

58. Write a program that display on console figure

Output

```
>      *
>     ***
>    *****
>   *******
>  *********
> ***********
>      *
>      *
```

Hint for program

					*					
				*	*	*				
			*	*	*	*	*			
		*	*	*	*	*	*	*		
	*	*	*	*	*	*	*	*	*	
*	*	*	*	*	*	*	*	*	*	*
					*					
					*					

59. Write a program that display a rhombus on console

Output 1

```
>     *
>    ***
>   *****
>  *******
> *********
>  *******
>   *****
>    ***
>     *
```

Hint for program 1

				*				
			*	*	*			
		*	*	*	*	*		
	*	*	*	*	*	*	*	
*	*	*	*	*	*	*	*	*
	*	*	*	*	*	*	*	
		*	*	*	*	*		
			*	*	*			
				*				

Output 2

```
>     *
>    * *
>   *   *
>  *     *
> *       *
>  *     *
>   *   *
>    * *
>     *
```

Hint for program 2

				*				
			*		*			
		*				*		
	*						*	
*								*
	*						*	
		*				*		
			*		*			
				*				

60. Write a program that display a figure on console

Output 1

```
>   *****
>  *******
> *********
>  *******
>   *****
```

Output 2

```
>   *****
>  *     *
> *       *
>  *     *
>   *****
```

61. Write a program that display a figure on console

Output

```
> ************
>            *
> ************
> *
> ************
>            *
> ************
> *
> ************
>            *
> ************
> *
> ************
```

62. Write a program that display a figure on console

Output

```
> *************           *
> *           *           *
> *           *           *
> *           *           *
> *           *           *
> *           *           *
> *           *           *
> *           *           *
> *           *************
```

63. Write a program that display a figure on console

Output

```
> *        *
> *        *
> *        *
> **********
> *        *
> *        *
> *        *
```

64. Write a program that display a figure on console

Output

```
> *     *
> **    *
> * *   *
> *  *  *
> *   * *
> *    **
> *     *
```

65. Write a program that display a figure on console

Output 1

```
> *************
> ****** ******
> *****   *****
> ****     ****
> ***       ***
> **         **
> *           *
> *************
```

Output 2

```
> *************
> *           *
> **         **
> ***       ***
> ****     ****
> *****   *****
> ****** ******
> *************
```

66. Write a program that display a figure on console

Output

```
>       *
>       *
>       *
>       *
> *************
> *************
> *************
> *************
> *************
>       *
>       *
>       *
>       *
```

67. Write a program that display a figure on console

Output

```
>      *********
>      *********
> *******************
>      *********
>      *********
```

68. Write a program that display on console a figure

Output

```
>         *
>  *************
>  *************
> ***************
>  *************
>  *************
>         *
```

69. Write a program that display a figure on console

Output 1

```
>    ***
>  *******
> *********
> *********
>  *******
>    ***
```

Output 2

```
>    ***
>  *     *
> *       *
> *       *
>  *     *
>    ***
```

70. Write a program that display a figure on console

Output 1

```
>    ***      ***      ***
>  *     *  *     *  *     *
> *       **       **       *
> *       **       **       *
```

71. Write a program that display a figure on console

Output

```
> *  ****  ****  ****  ****  *
> *  *  *  *  *  *  *  *  *  *
> *  *  *  *  *  *  *  *  *  *
> ****  ****  ****  ****  ****
```

72. Write a program that display a figure on console

Output 1

```
> *      *      *      *
> *      *      *      *
> *      *      *      *
> *      *      *      *
> **********************
```

Output 2

```
> **********************
> *      *      *      *
> *      *      *      *
> *      *      *      *
> *      *      *      *
```

73. Write a program that display a figure on console

Output 1

```
>    *     *     *     *
>   * *   * *   * *   * *
>  *   * *   * *   * *   *
> *     *     *     *     *
```

Output 2

```
> *     *     *     *     *
>  *   * *   * *   * *   *
>   * *   * *   * *   * *
>    *     *     *     *
```

74. Write a program that display a figure on console

Output 2

```
> *     *     *     *
>  *     *     *     *
>   *     *     *     *
>    *     *     *     *
```

Output 2

```
>    *     *     *     *
>   *     *     *     *
>  *     *     *     *
> *     *     *     *
```

75. Write a program that display a figure on console

Output

```
> *    *    *    *    *    *
>  *   **   **   **   **   *
>   *  * *  * *  * *  * *  *
>    * *  * *  * *  * *  * *
>      *    *    *    *    *
```

76. Write a program that display a figure on console

Output

```
>    ****      ****
>   ******    ******
>   ****************
>    **************
>      **********
>       ********
>        ******
>          **
```

77. Write a program that display a figure on console

Output

```
> ***           ***           ***
>     *             *             *
>      *             *             *
>      *             *             *
>     *             *             *
>     *****************************
>     *             *             *
>      *             *             *
>      *             *             *
>     *             *             *
> ***           ***           ***
```

78. Write a program that display on console expression computing value of π

$$\pi = 4\left(1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \frac{1}{11} + \frac{1}{13} - \frac{1}{15}...\right)$$

Output

```
```

79. Write a program that display on console expression computing **Euler number**

$$e = \frac{1}{1} + \frac{1}{1} + \frac{1}{1 \cdot 2} + \frac{1}{1 \cdot 2 \cdot 3}...$$

Output

```

```

80. Write a program that display on console first 15 **Fibonacci** numbers `https://en.wikipedia.org/wiki/Fibonacci_number`

Output

```
> 1  1  2  3  5  8  13  21  34  55  89  144  233  377  610
```

81. Write a program that print all **Armstrong** numbers between **1** and **1000** `https://en.wikipedia.org/wiki/Narcissistic_number` For example **153** is an armstrong number, because

$$153 = 1^3 + 5^3 + 3^3$$

82. Write a program that print all **Perfect** numbers between **1** and **500** In number theory, a perfect number is a positive integer that is equal to the sum of its positive divisors, excluding the number itself. For instance, 6 has divisors 1, 2 and 3 (excluding itself), and 1 + 2 + 3 = 6, so 6 is a perfect number.

83. Write a program that print all **Strong** numbers between **1** and **1000** Strong number is a number whose sum of all digits' factorial is equal to the number 'n'. Factorial implies when we find the product of all the numbers below that number including that number and is denoted by ! (Exclamation sign), For example: 4! = 4 * 3 * 2 * 1 = 24.

84. Write a program that find **HFC (Highest Common Factor)** of numbers **12** and **42**

85. Write a program that find **LCM (Lowest Common Multiple)** of numbers **12** and **30**

86. Write a program that check **Collatz conjecture**. The Collatz conjecture is a conjecture in mathematics that concerns a sequence defined as follows: start with any positive natural integer **n**. Then each term is obtained from the previous term as follows: if the previous term is even, the next term is one half of the previous term. If the previous term is odd, the next term is 3 times the previous term plus 1. The conjecture is that no matter what value of n, the sequence will always reach 1.

For Example

```
number 11

> 17 -> (11 * 3 + 1)/2
> 26 -> (17 * 3 + 1)/2
> 13 -> 26/2
> 20 -> 20/2
> 10 -> 10/2
> 5  -> (5 * 3 +1)/2
> 8  -> 8/2
> 4  -> 4/2
> 2  -> 2/2
> 1  -> end iteration
```

3

Arrays

1. Write a program that create an empty array of **6** elements and assign values from **1** to **6**

2. Write a program that create an empty array of **4** elements and assign index value of every element to every element an array

3. Write a program that create an array of **5** elements with **float point** type, assign different values and print on the console all array values

4. Write a program that create an array of **integer** elements type with length **5** and assign all elements with different values, after that display all elements of the array on console

5. Write a program that create an empty array of **10** elements and assign with value **5**

6. Write a program that create an empty array of **12** elements and assign with value **1**

7. Write a program that create an array of **8** elements and assign with value **0**, after that assign every even element of the array by value **101**, display all elements on console

8. Write a program that create an empty array of **6** elements and assign first half of array with value **3** and second half **4**

9. Write a program that create an empty array of **12** elements than assign first 4 elements with value **1**, next 4 elements with value **2**, next 4 elements with value **3**, after that display all elements on console

10. Write a program that create an array of **25** elements, and assign with values from **106** to **130**

11. Write a program that iterate through the array and check every element value:

- if value is equals to **5** than print **five**
- if value is equals to **0** than print **zero**
- if value is equal to **3** than print **three**
- if other value that print **undefined**

$$[2, 4, 5, 10, 0, -1, 0, -3, 6, 3, 2, 1, 3]$$

12. Write a program that print on a console every third element in an array

$$[6, 3, 7, 5, 8, 9, 5, 4, 5, 6, 3, 5]$$

For Example

```
print every second element
[1, 2, 3, 4, 0, -1]

> 3, -1
```

13. Write a program that print on a console every third element in an array starting from the end

$$[8, 4, 6, 2, 43, 2, -6, 0, 5, 4, -32, 12]$$

For Example

```
print every second element from the end
[7, 0, -4, 3, 1, 2]

> 3, 7
```

14. Write a program that fill an array from the center in two direction

$$[7, 6, 5, 4, 3, 2, 1, 0, 1, 2, 3, 4, 5, 6, 7]$$

15. Write a program that all elements from the input array copy to new array

$$[3, 54, 7, 3, 2, 6, 4, 3]$$

16. Write a program that copy all odds values of the input array into new array

$$[6, -6, 3, 2, 1, 67, 555, 45, 23, 76, 33, 11, -678]$$

For Example

```
[6, 7, 9, 4]

> [7, 9]
```

17. Write a program that copy all evens values of the input array into new array

$$[5, 7, 4, 4, 3, 2, 78, 45, -4, -3, 22, -100, 0, 12]$$

For Example

```
[1, 3, 6, 31]

> [6]
```

18. Write a program that copy all values that have odd index into new array

$$[7, 5, 4, 6, 12, 45, 98, -10]$$

For Example

```
[6, 4, 3, 1]

> [4, 1]
```

19. Write a program that copy all values that have even index into new array

$$[7, 3, -4, -1, 0, 4, 75, -90]$$

For Example

```
[-1, 0, 3, 99]

> [-1, 3]
```

20. Write a program that merge two arrays into new one (merging elements must be done one by one)

Input:

```
array one - [6, 34, 234 ,65, 0, -54, -3, 6, 3, 5]
array two - [6, 4, 3, 6, 4, -4]
```

Example

```
Input:  array one - [1, 2, 3], array two - [4, 5]
Output:  [1, 4, 2, 5, 3]
```

21. Write a program that print all elements on a console from the array of strings $[''array'', ''string'', ''programming'']$

22. Write a program that create a two-dimensional array with size **2 x 10** (two arrays by 10 elements), and assign values from **1** to **20**

23. Write a program that all elements from **two-dimensional** array copy into **one-dimensional** array

$$[$$

$$[4, 6, 8, 6, 4, 3, 6, 33, 6, 4, 2],$$

$$[5, 8, 7]$$

$$]$$

24. Write a program that all elements from **two-dimensional** string array copy into one-dimensional string array

$$[[''Hello'', '', world!'', ''I'', '''m''], [''a'', ''programmer'']]$$

25. Write a program that create an array of **15** elements and assign **random** integer numbers (use a random function)

26. Write a program that create an array with random size and assign all elements by **-1**

27. Write a program that write number **111** in binary representation (use boolean array), and print on console all array's numbers

For Example

```
number 5
[true, false, true]

> [1, 0, 1]
```

28. Write a program that print all array elements from the beginning to value **0** if it presents (program must interrupt when value of the array is equal to **0**)

$$[6, 5, 7, -5, 4, 6, 5, 0, 5, 6, 7]$$

$$[0, 3, 5, 4, 2, 4, -6, -100, 8]$$

$$[5, 4, 6, -5, -4, 3, 1]$$

29. Write a program that change the sign of every elements in the array by opposite

$$[0, 0, -123, 43, 6, 8, -999, 1000, 9877, -234]$$

For Example

```
[6, -4, 8, 0, -4, 3, -6]

> [-6, 4, -8, 0, 4, -3, 6]
```

30. Write a program that generate the array by using only loop

$$[1, 11, 111, 1111, 11111, 111111, 1111111, 11111111]$$

31. Write a program that sum all elements in the array

$$[0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1]$$

32. Write a program that do operation (+, -, /, *) with elements two arrays (round up to 2)

$$[5, 7, 8, 5, 3, 6, 8]$$

$$[67, 43, 2, 3, 4444, 3, 22]$$

For example

```
Input
[1, 2, 3]
[2, 7, 9]

Output

> operation -
> -1, -5, -6
> operation +
> 3, 9, 12
> operation *
> 2, 14, 27
> operation /
> 0.5, 0.29, 0.33
```

33. Write a program that count how many letter symbols in the array

$$['4',' 1',' a',' 1',' n',' y',' p',' 1',' 9',' p']$$

For example

```
array => [’l’, ’f’, ’r’, ’4’, ’1’]

> 3
```

34. Write a program that count how many symbols **y** in the array

$$['d,' f',' 3',' 1',' y',' r',' i',' y',' y',' u']$$

For example

```
array => [’a’, ’b’, ’y’, ’6’, ’y’]

> 2
```

35. Write a program that count evens and odds values in array

$$[234, 3457, 657, 456, 89, 234, 2, 45243, 5, 423, 23, 5, 423, 52]$$

For example

```
array = [5, 4, 9]
        > evens elements
        > 1
        > odds elements
        > 2
```

36. Write a program that calculate and display an arithmetic mean of array elements

$$[4, 8, 90, 0, 40, 32, 64, 22, 76]$$

For example

```
array = [2, 10, 12] => (2 + 8 + 12)/3

> 8
```

37. Write a program that calculate and an arithmetic mean for only positive elements

$$[6, 5, 3, -44, -5, 0, 5, -11, 4, 0, 3, 4, -54, -43]$$

For example

```
array = [-1, 0, 5, 4] => (5 + 4)/2

> 4.5
```

38. Write a program that display elements between **min** and **max** values of an input array

$$[6, 54, 5, 34, 43, 54, -9, 54, 3, 54, 5, 3456, 3, 5, 4, 3]$$

For example

```
[6, 5, 3, 8, 7, 6, 1, 54, 23, 4, 456, 4, 3, 5]
    min = 1
    max = 456
>  54, 23, 4
```

39. Write a program that calculate and display multiplication of all array elements that have positive value

$$[34, -56, 45, 0, -66, 34, 2, 1, -67, -999, -1000, -4]$$

For example

```
array = [9, -4, 5, 7, 0, -99, 4, -100] => 9 * 6 * 7 * 4

        > 1512
```

40. Write a program that calculate and display multiplication of all array elements that have negative value

$$[5, -7, -3, 5, 6, -99, -34, 0, 34, 2, 7, 89, 3, 4, -1, 5, -1, -2]$$

For example

```
array = [-1, 5, -6, 45, -10] => 1 * 6 * 10

        > 60
```

41. Write a program that calculate and display multiplication of all array elements

$$[3, 4, 5, 6, 3, 2, 5, 7, 8]$$

For example

```
array = [1, 3, 4] => 1 * 3 * 4

        > 12
```

42. Write a program that calculate sum of only negative array values that locate near

$$[5, 4, 67, -5, 34, -6, -5, 6, -5, 7, -8, -5, -4]$$

For example

```
array = [6, 5, 7, -5, -4, 7, 0, -1, 5]

> -9
```

43. Write a program that calculate sum of only negative array values

$$[6, 5, 4, -7, -3, 3, 5, 2, 6, 8, -4, -7, -1]$$

For example

```
array = [-1, 4, -2]

        > -3
```

44. Write a program that display sum of every unpaired element in the array

$$[2, 4, 5, 76, 3, 4, 6, 6, 3, 4, 2, 43, 234, 342, 2, 3, 45, 23, 46]$$

For example

```
array = [2, 3, 2, 1] => 2 + 2 = 4

        > 4
```

45. Write a program that calculate and display sum of every second element in an array

$$[5, 4, 7, 8, 45, 123, 4, 78, 5, 6, 7, 8, 9, 0, 23, 4, 34]$$

For example

```
array = [6, 5, 7, 3] => 5 + 3 = 8

        > 8
```

46. Write a program that calculate and display sum of all array elements

$$[5, 4, 6, 8, 3, 23, 5, 67, 45, 4, 24, 6, 34]$$

For example

```
array = [2, 5, 6] => 2 + 5 + 6

         > 13
```

47. Write a program that find all numbers from **101** to **999** whose square root is integer without remainder

For Example

```
from 1 to 10
> 1 4 9
```

48. Write a program that create and display a new array from evens elements of array

$$[234, 4, 5, 7, 4, 5, 3, 7, 9, 5]$$

For example

```
array = [9, 113, 65, 38]

         > [113, 38]
```

49. Write a program that sort array in ascending way

$$[5, 4, 6, 8, 3, 23, 5, 67, 45, 4, 24, 6, 34]$$

For example

```
array = [5, 4, 7, 1]

         > [1, 4, 5, 7]
```

50. Write a program that find minimum value (sign of number does not matter)

$$[4, 5, -23, -1234, 456, 7, -44, -1, 45, 23, 23, 1, 2]$$

For example 1

```
array = [4, 6, -2, -7]

        > -2
```

For example 2

```
array = [3, 6, 1, -7]

        > 1
```

51. Write a program that display on a console how many negative numbers divide by **2** without a remainder

$$-4, 5, -22, -1, 45, -99, -98, -55, 5, 43, 7, 64, 77, 107$$

For example

```
array = [4, -2, -3, 3, 7] => -2

        > 1
```

52. Write a program that display on a console those elements of array that belongs to the interval $[16, 300]$

$$[2, 4, 15, 17, 45, 23, 5, 47, 543, 234, 5345, 23, 1]$$

For example

```
array = [5, 4, 6, 3, 7]
        interval [6, 8]

        > 6 7
```

53. Write a program that display on a console those elements of the array that does not belong to the interval $[15, 35]$

$$[3, 4, 7, 16, 20, 3, 23, 30, 29, 4, 6, 8, 56, 32, 213, 53, 423, 7, 324, 442]$$

For example

```
array = [5, 8, 9, 1]
        interval [6, 8]

        > 5 9 1
```

54. Write a program that find min and max element in array

$$[3, 6, 5, 7, 34, 99, 6, 8, 32, -20, 45, -100]$$

For example

```
array = [-4, 2, 123, 4, 3]

> Min value = -4
> Max value = 123
```

55. Write a program that find the second **max** value of the array

$$[6, 34, 654, 76, 6723, 4, 5, 4, 43, 42, 6, 67, 2, 4]$$

For example

```
array = [5, 8 ,4, 5, 3, 7]

        > 7
```

56. Write a program that find the second **min** value of the array

$$[5, 7, 5, 0, 6, 4, -6, -4, 5]$$

For example

```
array = [-3, 4, 6, 5, 4, -3, -1, 7, -4, -7]

        > -4
```

57. Write a program that find in array prime numbers

$$[5, 4, 5, 7, 3, 8, 34, 234, 17, 19, 21, 44, 55, 67, 99, 100]$$

For example

```
array = [1, 6, 7, 10, 11]

> 1, 7, 11
```

58. Write a program that check if an array contains the value **5**

$$[6, 4, 3, 9, 10, 12, -45]$$

$$[7, 4, 5, 0, -39, 23]$$

59. Write a program that check if an array contains the values **55** and **45**

$$[5, 7, 5, 55, 4, 0, -9]$$

$$[-10, 44, 55, 3, -54, 45]$$

60. Write a program that check if **44** contains in array or not (for searching please use binary search)

```
array1 = [5, 2, 1, 7, 9, 10, 1001, 44, 3445, 2222, 1234567, 987, 3]
array2 = [5, 8, 3, 8, 4, 3, 6, 444, 66, 4, 66, 55]
```

For example

```
array 1 = [1, 44]
        array 2 = [4, 6]

        > true
        > false
```

61. Write a program that check arrays for identity (two arrays are identical if they contain the same amount of elements, and all values of elements are identical)

```
array1 = [9, 8, 7, 6, 5]
array2 = [9, 8, 7, 6, 5, 4, 3, 2, 1]
array3 = [9, 8, 7, 6, 5]
```

Need check identity for pairs array1 - array2, array2 - array3, array1 - array3

For example

```
array 1 = [1, 7]
array 2 = [4, 6]
array 3 = [1, 7]

> Identity array1 - array2
> false
> Identity array2 - array3
> false
> Identity array1 - array3
> true
```

62. Write a program that check how many time digit **5** presents in the array only in negative elements

$$[123, 4, -4, 5, -7, -999, -7, -11, -777, -77, -43, 45, 67, 999, 234, 173, 761]$$

For example

```
array = [1, 55, 6, 5, 7, -5, -10, -15]

> 2
```

63. Write a program that check how many times number **7** presents in an array

$$[4, 23, 7, 34, 17, 23, 77, 3243, 34, 23, 5, 777, 324, 12, 167, 234, 43]$$

For example

```
array = [2, 7, 5, 66, 12, 77, 55]

> 3
```

64. Write a program that check if an array contains unique element (or several) or not

$$[1, 2, 4, 5, 3, 1, 5, 10, 4, 2]$$

$$[5, 4, 5, 6, 7, 4, 6]$$

$$[5, 4, 6, 7]$$

For example

```
array = [1, 2, 3, 1]

        > 2, 3
```

For example

```
array = [1, 2, 3, 2, 3]

        > 1
```

For example

```
array = [1, 2, 1, 2, 3, 3]

        > no unique elements
```

65. Write a program that use the array to print on console (if number positive print '+' number's time, if number is negative print '-' number's time, if 0 just print nothing)

$$[5, -3, 0, 5, -2, 0, 0, -10, 12, -5, -7, 11, 0, 0, 4, -1]$$

For example

```
array = [1, 0, -2, 4, -1]

        > +
        >
        > --
        > ++++
        > -
```

66. Write a program that check if in the input array presents values **10** and **11**, if those values are present print on console **true**, if not print **false**

$$[4, 2356, 4, 32, 6, 578, 34, 7, 35, 547, 345, 6, 24, 10, 523, 5, 324]$$

$$[234, 54, 23, 411, 33, 11, 3, 21, 5, 57, 45, 7, 10, 3, 234, 5, 53, 45]$$

For example 1

```
Check for numbers 2 and 5
        [234, 43, 6, 34, 5, 42]

        > false
```

For example 2

```
Check for numbers 2 and 5
array = [32, 67, 4, 4, 2, 5, 6, 3]

        > true
```

67. Write a program that find the equal numbers pair in two numbers (**3459** and **46798**)

For Example

```
123 and 4352
> 2
> 3
```

68. Write a program that determine two elements in an array of integer whose sum is closest to zero

$$[5, 7, -6, -3, 0, 6, 7, 23, 23, 2, 100, -101, 5, 6, 7, 3]$$

69. Write a program that find two elements in a sorted array whose sum is **8**

$$[1, 3, 4, 4, 5, 1]$$

For Example

```
[1, 3, 4, 5]
> 3 and 5
```

70. Write a program that find quantity of different values in an array

$$[2, 6, 4, 6, 1, 4, 5, 4, 7, 8]$$

For Example

```
[1, 6, 1, 6, 5, 2]
> 2 (5 and 2)
```

71. Write a program that find repetitive values in an array

$$[33, 4, 3, 2, 5, 6, 4, 3, 3, 11, 4, 5, 7, 9, 123, 11, 12, 65]$$

For Example

```
[6, 4, 3, 6, 4, 8, 5, 3, 4, 1, 2]
> 6, 4, 3
```

72. Write a program that find **max** value in an array that has **max** index

$$[6, 4, 5, 65, 234, 6, 34, 6, 5, 4]$$

$$[2, 6, 4, 6, 1, 4, 5, 4, 7, 8]$$

For Example

```
[9, 4, 6, 2]
> max value 6 with index 2
```

73. Write a program that find the most often value in an array

$$[45, 45, 23, 45, 33, 31, 775, 789, -34, 43, 43, 45, 46, 45, 32, 45]$$

For Example

```
[1, 4, 5, 6, 2, 1, -6, 1, 4, 3, 2, 1, 1, 0, 1]
> 1
```

74. Write a program that convert a number **167** into words

For Example

```
number 25

> two five
```

75. Write a program that decompose a number **14367**

For Example

```
number 125

decomposed number
> 100 20 5
```

76. Write a program that convert a decompose **3459** into **romanian numbers**

For Example

```
number 238

decomposed romanian numbers
> II III VIII
```

77. Write a program that create a number from the array elements

$$[4, 5, 6, 7, 8, 3, 5, 7, 9]$$

For example

```
[1, 5, 7]

> 157
```

78. Write a program that create a number from all elements in array (glue all elements) $[4, 156, 43, 6, 8943]$

For example

```
array = [5, 6, 57]

        > 5657
```

79. Write a program that generate 30 elements of the array by following below principe

$$[1, 2, 3, 6, 12, 24, 48...]$$

80. Write a program that generate the array by following below principe

$$[1, 2, 12, 212, 12212, 21212212]$$

81. Write a program that insert value **5** into an array after index **2**

$$[4, 6, 4, 7, 3, 7, 4, 6, 5, 7, 3]$$

For example

```
insert value 3 after index 2
    [3, 5, 6, 8, 3]
    > [3, 5, 6, 3, 8, 3]
```

82. Write a program that insert value **1** into an array after every value **3** int the array (if it presents)

$$[4, 6, 2, 4, 2, 3, 6, 1, 2, 3, 6, 7, 9, 4, 6, 8, 4, 2, 3, 4, 3, 56, 3]$$

For example

```
[4, 5, 3, 7, 3]

> [4, 5, 3, 1, 7, 3, 1]
```

83. Write a program that insert value **9** into an array after and before every value **4** in the array (if it is presented in an input array)

$$[5, 7, 8, 4, 8, 234, 6, 35, 234, 4, 6, 7, 9, 6, 6, 4, 5, 2, 5, 7, 3, 2, 4, 1, 3, 7]$$

For example 1

```
[1, 5, 4, 7, 4]

> [1, 5, 9, 4, 9, 7, 9, 4, 9]
```

For example 2

```
[9, 8, 4, 4, 4]

> [9, 8, 9, 4, 9, 9, 4, 9, 9, 4, 9]
```

84. Write a program that insert value **101** after every element in the input array

$$[6, 4, 6, 4, 2, 34, 6, 4, 2, 3, 4, 32]$$

For example

```
[1, 2, 3, 4]

> [1, 101, 2, 101, 3, 101, 4, 101]
```

85. Write a program that insert the same value into the array after current value if it more than **30**

$$[3, 99, 0, -4, 11, 31, 21, 790]$$

For example

```
[61, -2, 13, 44] if more than 20

> [61, 61, -2, 13, 44, 44]
```

86. Write a program that insert the square root of even value after the current value (fractional part must be ignored)

$$[65, 55, 44, 32, -1, 0, 43, 8, 33, 44, 76]$$

For example

```
[1, 2, 4, 7, 9, 11, 12]

> [1, 2, 1, 4, 2, 9, 11, 12, 3]
```

87. Write a program that insert every odd value in an array to the end of an array

$$[55, 34, 33, -5, -10, 1234, 85, 4, 3, 6, 7, 8, 34, 56]$$

For example

```
[5, 4, 3, 1, 3, 4, 5, 11]

> [5, 4, 3, 1, 3, 4, 5, 11, 5, 3, 1, 3, 5, 11]
```

88. Write a program that insert value **0** befo§re and after the middle element of an array. If an array does no have the middle element, then insert value **0** between to parts of an array.

$$[5, 4, 3, 76, 44, -2, 4, -101, 99]$$

$$[6, 7, 8, 45, 43, -500, 123, 4323]$$

For example

```
[1, 2, 3, 4, 5]

[1, 2, 0, 3, 0, 4, 5]
```

For example

```
[1, 2, 3, 4]

[1, 2, 0, 3, 5]
```

89. Write a program that delete first half of an input array

$$[345, 34, 2, 7, 5, 0, 43, 23, 4, 6, 78, 54]$$

For example

```
[1, 2, 3, 4]

> [3, 4]
```

90. Write a program that delete from an inout array those elements that value are divided by 3 without a remainder

For example

```
[1, 2, 3, 4, 9, 6, 7, 3]

> [1, 2, 4, 7]
```

91. Write a program that delete all values from an input array that have odd index except two first elements

$$[4, 3, 5, 7, 3, 2, 4, 3, 23, 432, 24, 543, 34, 0, -44]$$

For example

```
[3, 4, 5, 6, 7, 8]

> [3, 4, 5, 7]
```

92. Write a program that delete first three elements from an input array

$$[]$$

For example

```
[1, 2, 3, 4, 5]

> [4, 5]
```

93. Write a program that delete last five elements from an input array

$$[65, -43, 65, 78, 3, 4, 8, -87, 0, 43, 34, 986]$$

For example

```
[5, 3, 5, 7, 3, 435, 54, 5, 87, 12, 123, 7, 6, 5]

> [5, 3, 5, 7, 3, 435, 54, 5, 87]
```

94. Write a program that delete elements on the left and on the right from element that is located on the index 3 in an input array

$$[5, 4, 6, 3, 33, -43, 0, 32]$$

For example

```
[7, 6, 5, 4, 3, 2, 1]

> [7, 6, 4, 2, 1]
```

95. Write a program that delete all prime numbers from an input array

$$[2, 4, 5, 6, 7, 23, 31, 45, 64, 3, 1, 5]$$

For example

```
[1, 2, 3, 4, 5, 6, 22, 23]

> [2, 4, 6, 22]
```

96. Write a program that delete all numbers from the second array that presents in the first array

$$[1, 2, 3, 4, 5]$$

$$[5, 4, 2, 3, 3, 1, 3, 64, 34, 23, 44, 5, 4, 3, 0, 3, 2, 22]$$

For example

```
[3, 6, 4, 1] (first array)
[1, 4, 5, 8] (second array)

> [5, 8]
```

97. Write a program that delete that max value from an input array (operation must be applied three times)

$$[555555, 4, 6, 4, 34, 21, 65443, 0, 54, 23, 12]$$

For example

```
[8, 6, 4, 3, 5, 66, 43]

> [8, 6, 4, 3, 5, 43]
> [8, 6, 4, 3, 5]
> [6, 4, 3, 5]
```

98. Write a program that delete every second element in an array

$$[6, 5, 4, 2, 3, 4, 5, 2, 34, 432, 0, -432, 43, 433]$$

For example

```
[3, 7, 5, 3]

> [3, 5]
```

99. Write a program that delete element in the array by index **5**

$$[5, 7, 4, 4, 4, 6, 8, 4, 3, 4]$$

100. Write a program that delete from the array value **12** if it presents

$$[5, 34, 7, 6, 4, 7, 12, 3, 23, 5, 65, 4]$$

101. Write a program that delete from the array all numbers that more equal than 10

$$[43, 1, 3, 5, 2, 5, 8, 89, 65, 2, 32, 12, 3]$$

For example

```
[34, 3, 5, 6, 21, 44, 11]

> [3, 5, 6]
```

102. Write a program that delete from the array all numbers that less equal than **25**

$$[4, 67, 7, 234, 6, 4, 6, 76, 34, 5, 87, 234, 4, 2, 3, 6, 3, 56, 34]$$

For example

```
[5, 4, 26, 654, 25]

> [26, 654]
```

103. Write a program that delete from the array all negative numbers

$$[6, 4, 7, -5, 3, -44, 3, 7, 8, -2, -123, 7, -77, 43, -3, -3, -1, 0, 5]$$

For example

```
[4, -3, 5, 9, -5, 5]

> [4, 5, 9, 5]
```

104. Write a program that delete from the array all positive numbers and **0**

$$[6, -5, 0, 3, 4, -5, -4, 0, 3, 3, 0, 0, -5, 0]$$

For example

```
[0, 1, 2, -1, 0, 0, -3, 0, -4, 9, 6]

> [1, 2, 9, 6]
```

105. Write a program that delete the third negative element in the array

$$[-6, 5, 4, -4, -123, -9, 5, -4, 23, 0, -34, -12345, 90]$$

For example

```
[-1, 5, -2, 4, -1, 6, -7]

> [-1, 5, -2, 4, 6, -7]
```

106. Write a program that delete from the array elements from index **4** to **10**

$$[6, 4, 67, 8, 45, 23, 7, -5, -234, 0, 0, 4, 2, 0, -324, 3, 5, 7, 5, 3]$$

For example

```
[6, 4, 78, 5, 3, 0, 56] delete from index 1 to 3

> [6, 3, 0, 56]
```

107. Write a program that delete all duplicated elements from the array

$$[5, 4, 5, 6, 7, 5, 7, 3, 4, 5, 7, 1, 1, 2, 3, 1, 3, 4, 5, 1]$$

For example

```
[2, 4, 1, 2, 4, 2, 5, 1]

> [5]
```

108. Write a program that check an array by pairs elements, if a pair sum less than **20** than delete larger value in a pair

$$[3, 41, 0, 5, -10, 121, -11, 4, 7, 9, 0, 237]$$

For example

```
[1, 2, 15, 21, -10, 123] (delete 2)

> [1, 15, 21, -10, 123]
```

109. Write a program that delete in an input array those elements which contain digit **1**

$$[3, 7, 12, 0, -5, 4, 10, -101, 1, 0, 5, 4, 11]$$

For example

```
[1, 2, 3, 11]

> [2, 3]
```

110. Write a program that copy all elements into two-dimensional array in specific way

For Example

```
[6, 12, 65, -5, 1]
[
    [6],
    [12],
    [65],
    [-5],
    [1]
]
```

111. Write a program that from the given array create two arrays, the first array contain elements from first half of the input array, and the second array second half

$$[54, 234, 6, 2, 4, 1, 5, 7, 324, 2, 56, 2, 43, 32]$$

For example

```
array = [3, 4, 5, 1]

        > [3, 4]
        > [5, 1]
```

112. Write a program that move to new array only those elements that arithmetic mean of input array elements is less than element value

$$[4, 456, 35, 43, 346, 234, 34, 74, 775, 234, 55, 45]$$

For example

```
array = [1, 2, 6, 4, 6]
arithmetic mean = (1 + 2 + 6 + 4 + 6)/ 5 = 3.8

> [1, 2]
```

113. Write a program that glue two arrays of symbols into one

$$['a','v','t','g','y','d']$$

$$['w','g','q','a','k','g','f','v']$$

For example

```
array 1 => [’a’, ’b’]
        array 2 => [’c’, ’d’]

        > [’a, ’b’, ’c’, ’d’]
```

114. Write a program that create and display a new array from odds elements of array

$$[6, 8, 44, 55, 23, 7, 6, 4, 9, 1, 12, 12]$$

For example

```
array = [7, 5, 4, 1]

        > [7, 4]
```

115. Write a program that from two arrays create one array with elements that presents in first and second array (sets intersections)

```
array 1 => [345, 3, 45, 4, 5, 1, 55, 45, 54, 33, 34, 49]
array 2 => [4100, 3, 6, 7, 54, 49, 1]
```

For example

```
array 1 => [1, 2, 3, 4]
        array 2 => [3, 4, 5, 6, 8]
        > [3, 4]
```

116. Write a program that convert input array in a way that every number in the array split as simple numbers (from 0 to 9 for example **21** to **2 and 1**, **123** to **1 and 2 and 3**) and put all into new array

$$[1, 22, 345, 234523456, 34532, 854, 456546, 77543, 2345234, 2345, 3245, 2, 47376, 345]$$

For example

```
array = [1, 234, 5678, 4, 33]
          > [1, 2, 3, 4, 5, 6, 7, 4, 3, 3]
```

117. Write a program that create the new array from input array in a way, that new array must contain only unique elements from input array

$$[1, 1, 2, 3, 4, 4, 5, 5, 5, 5, 6, 7, 8, 9, 0, 0]$$

For example

```
array = [4, 4, 3, 8, 3, 10]

> [8, 10]
```

118. Write a program that create array from several arrays and sort elements in asc order

$$Array\ 1 = [3, 6, 4, 43, 66, 55, 32, 98]$$
$$Array\ 2 = [123, 6789, 55, 443]$$
$$Array\ 3 = [1, 2, 3]$$
$$Array\ 4 = [999, 456, 43, 678]$$

For example

```
array 1 = [4, 3, 8]
array 1 = [11, 9]

> [3, 4, 8, 9, 11]
```

119. Write a program that convert one dimensional array into two dimension array in a way that every element from input array will be single element in second array

$$[4, 5, 6, 34, 435, 32, 4, 6]$$

For example

```
array = [1, 2, 3]

> [[1], [2], [3]]
```

120. Write a program that divide all elements into the array by the **min** value of array (round values up to 2)

$$[6, 7, 3, 4, 6, 34, 24, 64, 87, 234, 78, 934, 324]$$

For example

```
array = [7 , 5 , 3] => 7/3 , 5/3 , 3/3

        > [2.33 , 1.67 , 1]
```

121. Write a program that divide all elements into the array by the **max** value of array (round values up to 3)

$$[5, 32, 6, 4, 2, 35, 7, 79, 43, 5, 432, 7, 5]$$

For example

```
array = [5 , 7 , 4 , 23 , 4] => 5/23 , 7/23 , 4/23 , 23/23 , 4/23

        > [0.213 , 0.304 , 0.174 , 1 , 0.174]
```

122. Write a program that copy to a new array only negative numbers

$$[6, 5, 3, -7, -4, 3, 6, 8, 5, 3, 6, 89, -2, 5, -1, 5, -1234, 6, 8]$$

For example

```
[6 , 3 , -6 , 3 , 4 , -8]

8]
```

123. Write a program that square root every element of the array, and round to 3 numbers after point

$$[5, 34, 778, 43, 2345, 7, 8, 9, 345, 32, 12, 146, 76, 37, 97]$$

For example

```
array = [3 , 1 , 4 , 5]

        > [1.732 , 1 , 2 , 2.236]
```

124. Write a program that iterate from the second and to the penultimate elements in the array and in every iteration check if en element on the left more equal than an element on the right than put a current element into a new array, if not skip

$$[2, 5, 1, -5, 9, 7, 3, 12, 0, -2]$$

For example

```
[1, 3, 4, 2, -3, 10, 3]
```

125. Write a program that convert all array elements into **char** type and print on the console

$$[72, 69, 76, 76, 79]$$

126. Write a program that split input array into three other arrays

$$[6, 4, 5, 2, 43, 543, 23, -5, 2432, 56, 2, 0, 3, 21, 345, 76, 2, 543, -589, 32, 23]$$

For Example

```
[1, 2, 3, 4, 5, 6, 7, 8, 9]

 7]
> [2, 5, 8]
> [3, 6, 9]
```

127. Write a program that do for every element in an array

- if value more than 0 replace value by **1**
- if value less than 0 replace value by **-1**
- of value is 0 replace value by **2**

$$[6, 5, -4, 6 - 32, 65, 0, 8, 7, -65, 4, 3, 1, 5, -100, 0, 1]$$

128. Write a program that mirror left side arrays elements

$$[6, -4, 6, 4, 7, 34, 3, 4, 101]$$

For example

```
[6 , 5 , 4]
 6]
```

129. Write a program that swap elements in the array if they difference more equal **5**

$$[6, 5, 11, 4, 53, 54, 23, 5, 4, 6, 4, -1, -44, 5]$$

For example

```
[4 , 23 , 3 , 4 , 5 , 14]
, 3 , 4 , 14 , 5]
```

130. Write a program that swap only negative elements which difference more than **4**

$$[2, 6, -8, 33, -5, 0, 4, -10, 7, -13, 11]$$

For example

```
[4 , -1 , 5 , -6 , 9 , -5]
, 5 , -5 , 9 , -1]
```

131. Write a program that modify elements in input array in a way, that if value of element id divided by 2 without a reminder than replace that value by 1, otherwise by 0

$$[2345, 2346, 56, 7, 45, 32, 5, 234, 5, 23, 45, 2543, 6, 3, 45, 4568, 34, 56, 34, 56]$$

For example

```
array = [4 , 3]

          > [1 , 0]
```

132. Write a program that change all negative numbers in the array to positive

$$[5, -7, 4, -1, 0, 7, 12, -66, -3, 8, 2, -7, -19]$$

For example

```
array = [4, -2, 3]
        > [4, 2, 3]
```

133. Write a program that change all negative numbers in the array to positive and all positive numbers to negative

$$[5, 8, 33, -6, -3, 7, 0, -6, 45, -4, -1, 9, -8]$$

For example

```
array = [-1, 9, 4]
        > [1, -9, -4]
```

134. Write a program that change all elements in the array to uppercase

$$['a','A','b','s','q','i','p','r','m']$$

For example

```
array => [’y’, ’6’, ’y’, ’g’, ’t’]

        > [’Y’, ’6’, ’Y’, ’G’, ’T’]
```

135. Write a program that change all elements in the array to lowercase

$$['a','A','G','3','X','c','V','M','F','d']$$

For example

```
array => [’T’, ’O’, ’r’, ’F’, ’P’]

        > [’t’, ’O’, ’r’, ’f’, ’p’]
```

136. Write a program that revers all elements in array

$$[5, 4, 6, 2, 3, 4, 5, 7, 8, 9, 12, 56, 56, 1, 3, 4]$$

For example

```
array = [4, 3, 6, 8]

        > [8, 6, 3, 4]
```

137. Write a program that sort array in descending way

$$[345, 6, 4, 3, 8, 0, 4, 4, 5546, 43]$$

For example

```
array = [2, 3, 7, 4]

        > [7, 4, 3, 2]
```

138. Write a program that swap **min** and **max** values into the input array

$$[5, 7, 34, 54, -5, 43, 50, 4, 42, 7, 3, 3, 2, 1, 55, 9]$$

For example

```
array = [5, 8, 4, 3]

        > [5, 3, 4, 8]
```

139. Write a program that swap elements in two-dimensional array

```
array = [
    [4, 7, 8, 5, 4, 3, 5, 2],
    [4, 6, 8, 3, 2, 6, 3, 3]
]
```

For example

```
array = [ [1, 2], [3, 4] ]

        > [ [3, 4], [1, 2] ]
```

140. Write a program that swap elements between two arrays in special way. Swap paired, a first element in the first array swap with a second element in the second array, a second element in the first array swap with a first element in the second array.

$$[5, 4, 6, 8, 9, 3, 4]$$

$$[6, 7, 8, 9, 4, 5, 3]$$

For example

```
array 1 = [1, 2, 3, 4]
        array 2 = [5, 6, 7, 8]
        > [6, 5, 8, 7]
        > [2, 1, 4, 3]
```

141. Write a program that in the given array sort (desc) only first half of the array

$$[6, 8, 3, 4, -6, 0, 34, 101, 1001, 3, 5, 5, 23, 432]$$

142. Write a program that in the given array sort (asc) only second half of the array

$$[2, 1, 12, 4, 6, 3, 77, 4, 5, 2, 8, 1]$$

For example

```
array = [5, 2, 4, 4100, 44, 100]

        > [5, 2, 4, 44, 100, 4100]
```

143. Write a program that swap pair elements in the array only if two numbers are even

$$[6, 5, 4, 8, 3, 1, 8, 10, 9, 4, 34, 2]$$

For example

```
array = [1, 2, 8, 4, 4, 3]

        > [1, 2, 4, 8, 4, 3]
```

144. Write a program that switch pair elements in the array in a way (fist switch with last, second switch with penultimate etc.)

$$[4, 4, 7, 3, 54, 88, 32, 65, 43, 23]$$

For example

```
array = [45 , 43 , 21 , 3 , 54 , 65]

        > [65 , 54 , 3 , 21 , 43 , 45]
```

145. Write a program that switch pair elements in the array

$$[5, 65, 3, 5, 3, 6, 7, 9, 12, 544, 86, 12]$$

For example

```
array = [3 , 5 , 99 , 34 , 101 , 1]

        > [5 , 3 , 34 , 99 , 1 , 101]
```

146. Write a program that squared every element of the array

$$[4, 5, 7, 4, 33, 66, 54, 3, 3, 7, 8, 0, 24, 5, 43]$$

For example

```
array = [5 , 1 , 5 , 6]

        > [25 , 1 , 25 , 36]
```

147. Write a program that subtract **4** and multiply by **2** every element in the array

$$[56, 456, 12, 34, 3, 5, 7, 90, -50, -3]$$

148. Write a program that divide by **4** those elements in the input array that have remainder **0** with division by **2**

$$[5, 1, 3, -7, 24, 48, 3, 7, 9, 7, -21, -11, 0, 4]$$

For example

```
[3, 8, 1, 12, 9]
 1, 3, 9]
```

149. Write a program that shift all elements by **1** position to the right

$$[12, 4, -5, -2, 0, 4, 6, 2, -55, 43, 7, 32, 4, 3]$$

For example

```
[1, 2, 3, 4]
 2, 3]
```

150. Write a program that shift all elements by **1** position to the left

$$[6, 4, 8, 6, 5, -2, 4, 5, 55, 23, 7, 3, 1]$$

For example

```
[4, 5, 6, 7]
> [5, 6, 7, 4]
```

151. Write a program that replace the second half of an array by the first one

$$[65, 34, -6, 0, 43, -7, 2, 4, 65, 345, 432, 54, 13, 43]$$

For example

```
[4, 5, 6, 7, 8, 5]
> [0, 0, 0, 4, 5, 6]
```

152. Write a program that multiply by **3** every element in the array that divide by **2** without remainder

$$[5, 6, 4, 11, 2, 12, 5, 6, 8, -43, 0, 34, 13]$$

153. Write a program that multiply by **2** every element in the array

$$[5, 7, 4, 7, 4, 5, 7, 5, 4, 3, 4, 5]$$

154. Write a program that add **2** to every element in the array that value more equals to **10**

$$[5, 6, 33, 4, 6, 1, 11, 3, 15, 4, 6]$$

155. Write a program that subtract **11** from every element in the array

$$[5, 7, 32, 57, 3, 34, 54, 7, 568, 23, 5, 3, 4, 5, 54]$$

156. Write a program that add **21** to every element in the array

$$[6, 8, 5, 5, 7, 7, 453, 3, 5, 34, 23, 6, 5]$$

157. Write a program that change every element in an array by remaining of division **3**

$$[6, 5, 6, 7, 8, 9, 4, 32, 33, 55, 76, 123, 333]$$

158. Write a program that mirror two-dimensional array vertically

```
array = [
    [1],
    [4, 6],
    [5, 4, 2],
    [9, 5, 4, 7],
    [0, 4, 3, 4, 3],
    [4, 2, 1, 5, 6, 7],
    [5, 7, 8, 4, 2, 2, 3]
]
```

For example

```
[
    [3]
    [6, 7]
    [3, 9, 8]
]

[
    [3]
    [7, 6]
    [8, 9, 3]
]
```

159. Write a program that adds two matrices

```
Matrix A
[
    [5, 4, 6, 7, 4, 4],
    [7, 4, 2, 3, 2, 1],
    [8, 33, 21, 4, 0, -3],
    [-50, 3, 2, 41, 3, 5],
    [4, 3, 1, 3, 2, 3],
    [5, 4, 7, 9, 3, 5]
]
Matrix B
[
    [4, 3, 5, 2, 4, 6],
    [-9, 0, 4, 20, 4, 3],
    [-10, 34, 2, 3, 55, 4],
    [3, 5, 7, 4, -1, 5],
    [5, 5, 4, 2, 6, 3],
    [8, 3, 5, 3, 2, 4]
]
```

For example

```
Matrix A
        [
            [1, 2],
            [3, 4]
        ]
        Matrix B
        [
            [5, 6],
            [7, 8]
        ]

        Matrix C
        [
            [1 + 5, 2 + 6],
            [3 + 7, 4 + 8]
        ]
```

160. Write a program that subtract two matrices

```
Matrix A
[
    [7, 5, 34, 3, 5, 3],
    [-6, 4, 7 ,-5, 4, 3],
    [45, 23, 6, 4, 3, 5],
    [0, 6, 0, 5 , -3, 54],
    [6, 8, 34, 5, 34, -2],
    [54, 7, 4, 6, 45 ,4]
]
Matrix B
[
    [6, 7, 4, 7 , 2, 4],
    [0, 0, 0, -1, 2, 32],
    [-6, 4, 2, 6, 4, 2],
    [7, -5, -3, 4, 7, 34],
    [0, 5, 4, 3, 0, 43],
    [6, 4, 7, 4, 6, 2]
]
```

For example

```
Matrix A
        [
            [1, 2],
            [3, 4]
        ]
        Matrix B
        [
            [5, 6],
            [7, 8]
        ]

        Matrix C
        [
            [1 - 5, 2 - 6],
            [3 - 7, 4 - 8]
        ]
```

161. Write a program that perform scalar matrix multiplication by **3**

```
Matrix A
[
    [6, 7, 6],
    [0, -5, 43],
    [-5, 7, 5],
]
```

For example

```
Matrix A
        [
            [1, 2],
            [3, 4]
        ]

        multiplier : 2

        Scalar Matrix
        [
            [1 * 2, 2 * 2],
            [3 * 2, 4 * 2]
        ]
```

162. Write a program that check if matrices are equal

```
Matrix A
[
    [6, 4, 3],
    [8, 3, 2],
    [8, 4, 3],
]
Matrix B
[
    [6, 4, 3],
    [8, 3, 2],
    [8, 4, 3],
]
Matrix C
[
    [8, -5, 3],
    [0, 5, 1],
    [-1, 4, 5],
]
```

For example

```
Matrix A
        [
            [1, 2],
            [3, 4]
        ]
        Matrix B
        [
            [1, 2],
            [3, 4]
        ]
        Matrix C
        [
            [5, 6],
            [7, 8]
        ]

        Matrices A and B are equals
        Matrices A and C are not equals
```

163. Write a program that interchange matrix diagonals

```
Matrix
[
    [6, 4, 3, 2],
    [3, 7, 4, 6],
    [7, 8, 5, 4],
    [0, 5, 4, 3]
]
```

For example

```
Matrix
        [
            [1, 2],
            [4, 5]
        ]

        Matrix with interchanged diagonals
        [
            [2, 1],
            [5, 4]
        ]
```

164. Write a program that find sum of each row and columns of a matrix

```
Matrix
[
    [6, 4, 5, 7, 3],
    [7, 6, 3, -3, 0],
    [4, 3, 5, -4, 2]
]
```

For example

```
Matrix
        [
            [1, 2, 3],
            [4, 5, 6]
        ]

        > Sum of row 1: 1 + 2 + 3 = 6
        > Sum of row 2: 4 + 5 + 6 = 15
        > Sum of column 1: 1 + 4 = 5
        > Sum of column 2: 2 + 5 = 7
        > Sum of column 3: 3 + 6 = 9
```

165. Write a program that multiplicates two matrixes

```
Matrix A
[
    [-5, 4, 0],
    [101, 22, 10]
]
Matrix B
[
    [1111, -99],
    [6, 8],
    [4, 3]
]
```

For example

```
Matrix A
        [
            [1, 2, 6],
            [3, 4, 7]
        ]
        Matrix B
        [
            [1, 2],
            [3, 8],
            [5, 7]
        ]

        Matrix C
        [
            [1 * 1 + 2 * 3 + 6 * 5,
1 * 2 + 2 * 8 + 6 * 7],
            [3 * 1 + 4 * 3 + 7 * 5,
3 * 2 + 4 * 8 + 7 * 7]
        ]
```

166. Write a program that calculate multiplication of 4 cube diagonals (3-dimensional input array)

```
[
    [
        [2, 2, 9, 7, 9],
        [9, 7, 1, 8, 5],
        [7, 3, 2, 5, 2],
        [8, 2, 8, 3, 8],
        [1, 5, 3, 4, 4],
    ],
    [
        [2, 4, 3, 1, 2],
        [6, 4, 1, 4, 5],
        [9, 3, 2, 3, 4],
        [2, 5, 9, 7, 3],
        [3, 8, 2, 1, 2],
    ],
    [
        [2, 3, 3, 2, 9],
        [9, 4, 1, 3, 6],
        [7, 3, 2, 4, 2],
        [0, 7, 1, 5, 8],
        [9, 5, 2, 3, 1],
    ],
    [
        [7, 5, 3, 4, 2],
        [5, 3, 3, 4, 2],
        [8, 6, 9, 8, 3],
        [4, 3, 2, 1, 3],
        [2, 1, 5, 6, 7]
    ],
    [
        [5, 7, 5, 4, 2],
        [4, 3, 5, 6, 4],
        [3, 2, 5, 6, 5],
        [2, 1, 3, 1, 5],
        [9, 8, 4, 3, 2]
    ]
]
```

For example

```
[
    [
        [1, 2, 3],
        [4, 5, 6],
        [7, 8, 9]
    ],
    [
        [10, 11, 12],
        [13, 14, 15],
        [16, 17, 18]
    ],
    [
        [19, 20, 21],
        [22, 23, 24],
        [25, 26, 27]
    ]
]

diagonal  1 * 14 * 27 = 378
 diagonal 3 * 14 * 25 = 1050
diagonal  19 * 14 * 9 = 2394
 diagonal 21 * 14 * 7 = 2058
```

167. Write a program that check if two-dimensional array (matrix in our case) is an upper triangular matrix

```
array 1 = [
    [6, 7, 8, 4,  3]
    [0, 3, 2, 11, 44]
    [0, 0, 8, 4,  56]
    [0, 0, 0, 9,  1]
    [0, 0, 0, 0,  2]
]

array 2 = [
    [7, 7, 0, 4,  6]
    [0, 2, 2, 11, 4]
    [0, 0, 1, 4,  -5]
    [0, 4, 0, -6,  3]
    [1, 0, 0, 0,  -99]
]
```

For example

```
upper triangular
        [
            [1, 2, 3]
            [0, 4, 5]
            [0, 0, 6]
        ]

        not upper triangular
        [
            [1, 2, 3]
            [4, 5, 6]
            [0, 0, 7]
        ]
```

168. Write a program that check if two-dimensional array (matrix in our case) is an lower triangular matrix

```
array 1 = [
    [6, 0, 0, 0, 0]
    [8, 3, 0, 0, 0
        [4, 0, 8, 4, 0]
        [0, 5, 6, 9, 1]
        [2, 3, 6, 7, 2]
    ]

    array 2 = [
        [7, 0, 0, 0, 0]
        [8, 2, 0, 0, 0]
        [2, 5, 1, 0, 0]
        [1, 4, 6, 6, 3]
        [1, 4, 3, 5, 9]
    ]
```

For example

```
lower triangular
        [
            [1, 0, 0]
            [5, 4, 0]
            [4, 6, 6]
        ]

        not lower triangular
        [
            [1, 0, 3]
            [4, 5, 0]
            [3, 1, 7]
        ]
```

169. Write a program that find determinant of a matrix

```
matrix 1
[
    [32, 76],
    [444, 44]
]

matrix 2
[
    [54, 3, 4],
    [65, 2, 43],
    [78, 4, 2]
]
```

For example

```
matrix A
        [
            [a, b],
            [c, d]
        ]

        det(A) = a * d - c * b

        matrix B
        [
            [a, b, c],
            [d, e, f],
            [g, h, i]
        ]

        det(B) = a * det($\begin{bmatrix} e & f \\ h & i \end{bmatrix}$) - b * det($\begin{bmatrix} d & g \\ f & i \end{bmatrix}$) + c * det($\begin{bmatrix} d & e \\ g & h \end{bmatrix}$)
```

170. Write a program that copy all elements from the input array into two-dimensional array (in matrix diagonals)

$$[1, 2, 3, 4, 5, 6, 7, 8, 9]$$

For example

```
array = [1, 2, 4]

        matrix 1
        [
            [1, 0, 0],
            [0, 2, 0],
            [0, 0, 3]
        ]

        matrix 2
        [
            [0, 0, 1],
            [0, 2, 0],
            [3, 0, 0]
        ]
```

171. Write a program that transform the input array into matrix 4x4 (two-dimension array) in a specific way. All elements must fill a triangle matrix, if elements into array more than elements in a triangle they must be omitted.

$$[34, 54, 65, 234, 676, 86, 87, 3, 4, 5, 8, 456, 456]$$

For example

```
array = [1, 2, 3, 4, 5, 6, 7, 8, 9]

        matrix 3x3

        [
            [1, 2, 3],
            [0, 4, 5],
            [0, 0, 6]
        ]
        elements 7, 8, 9 omitted

        matrix 4x4

        [
            [1, 2, 3, 4],
            [0, 5, 7, 7],
            [0, 0, 8, 9],
            [0, 0, 0, 0]
        ]
        if elements in the array less than in
        triangle they must full by 0
```

172. Write a program that from the given two-dimensional array with the same numbers of elements (aka matrix) make multiplication of two matrix diagonals

```
[
    [5, 4, 3, 1, 9],
    [9, 4, 1, 4, 6],
    [7, 3, 2, 5, 5],
    [3, 2, 8, 7, 8],
    [9, 5, 6, 1, 2],
]
```

For example

```
[
    [4, 2, 3]
    [3, 8, 4]
    [5, 1, 6]
]

diagonal 4 * 8 * 6

 diagonal 3 * 8 * 5
```

173. Write a program check if matrix a sparse or not (a sparse matrix is a matrix in which more the a half elements have value **0**)

```
Matrix A
[
    [4, 0, 9, 0],
    [0, 0, 0, 1],
    [0, 0, 9, 0],
    [4, 0, 0, 1]
]
Matrix B
[
    [1, 3, 2, 0],
    [0, 1, 2, 1],
    [0, 3, 9, 0],
    [6, 5, 2, 1]
]
```

For example

```
Matrix A (is sparse matrix)
        [
            [0, 0],
            [0, 9]
        ]

        Matrix B (not a sparse matrix)
        [
            [1, 0],
            [0, 9]
        ]
```

174. Write a program check if matrix is identity (Identity matrix is a special square matrix whose main diagonal elements is equal to 1 and other elements are 0. Identity matrix is also known as unit matrix.)

```
Matrix A
[
    [1, 0, 0, 0],
    [0, 1, 0, 0],
    [0, 0, 1, 0],
    [0, 0, 0, 1]
]
Matrix B
[
    [1, 0, 0, 0],
    [0, 1, 0, 0],
    [0, 0, 1, 0],
    [0, 1, 0, 1]
]
```

For example

```
Matrix A (an identity matrix)
        [
            [1, 0],
            [0, 1]
        ]

        Matrix B (not an identity matrix)
        [
            [1, 2],
            [0, 1]
        ]
```

175. Write a program that find transpose of a matrix (Transpose of a matrix A is defined as converting all rows into columns and columns into rows)

```
Matrix A
[
    [1, 2, 3, 4],
    [5, 6, 7, 8],
    [4, 7, 4, 6],
    [5, 4, 3, 3]
]
Matrix B
```

```
[
    [3, 2, 3, 7],
    [2, 9, 1, 3],
    [2, 9, 5, 3],
    [4, 8, 1, 1]
]
```

For example

```
Matrix A (an identity matrix)
        [
            [1, 2, 3],
            [4, 5, 6],
            [7, 8, 9]
        ]

        transpose of Matrix A
        [
            [1, 4, 7],
            [2, 5, 8],
            [3, 6, 9]
        ]
```

176. Write a program that find sum of lower triangular matrix

```
Matrix A
[
    [1, 0, 0, 0],
    [-4, 6, 0, 0],
    [11, -1, 4, 0],
    [7, 4, 3, 2]
]
Matrix B
[
    [3, 0, 0, 0],
    [10, 9, 0, 0],
    [-9, 21, 5, 0],
    [7, 2, -4, 1]
]
```

For example

```
Matrix A
        [
            [1, 0, 0],
            [4, 5, 0],
            [7, 8, 9]
        ]

        > sum = 4 + 8 + 7 = 19
```

177. Write a program that find sum of upper triangular matrix

```
Matrix A
[
    [2, -1, 789, 33],
    [0, 9, 22, 44],
    [0, 0, 1, 123],
    [0, 0, 0, 4]
]
Matrix B
[
    [6, 3, 76, 456],
    [0, 33, -55, 23],
    [0, 0, 32, 89],
    [0, 0, 0, 123]
]
```

For example

```
Matrix A
[
    [1, 2, 3],
    [0, 5, 6],
    [0, 0, 9]
]

> sum = 2 + 3 + 6 = 11
```

178. Write a program that find sort(asc) all matrix columns

```
Matrix A
[
    [6, 5, 5, 5, 3],
    [-4, 4, 1, 8, 0],
    [11, 5, 6, -8, -1],
    [6, 7, 8, 3, 2]
]
```

For example

```
Matrix A
[
    [4, 3],
    [1, 2],
]

Matrix A (sorted columns)
[
    [1, 2],
    [4, 3],
]
```

179. Write a program that check Syracuse hypothesis for numbers from **20** to **30**. Let's take any natural number. If it is even, we divide it in half, if it is odd, we multiply by 3, add 1 and divide it in half. Let's repeat these actions with the newly received number. The hypothesis says that regardless of the choice of the first number, sooner or later we will get 1.

180. Write a program that print on the console "Hello World" with using Morse's ABC

181. Write a program that find prime numbers from **1** to **100** by using algorithm **Sieve of Eratosphenus**

4

Strings

1. Write a program that concatenate several strings into one and display the result on console

Input
”A crow will ”
” never ”
”be a falcon”

For example

```
Input
"Hello,"
"world!"


> Hello, world!
```

2. Write a program that generate **1 000 000** strings (put them into an array), every string must randomly generated (from **10** to **1000** randomly symbols). Than a program must count symbols in all strings.

3. Write a program that count how many letters contains in string

”*A friendly word is better than a heavy cake*”

For example

```
string => "hello"

> 5
```

4. Write a program that count how many **’a’** symbols contains in a string

”*A hungry wolf is stronger than a satisfied dog*”

For example

```
string => "transitive word"

> 1
```

5. Write a program that convert string into an array of symbols

”*Borrowed bread lies heavy on the stomach*”

For example

```
string => "Training"

> [’T’, ’r’, ’a’, ’i’, ’n’, ’i’, ’n’, ’g’]
```

6. Write a program that convert all symbols in string to upper case

”*Deficiencies come by the kilo and go by the gram*”

For example

```
string => "Programming engineer"

> "PROGRAMMING ENGINEER"
```

7. Write a program that create and display an array, that contain indexes of symbol ’o’ in string

”*Fools love not the wise, drunkards love not the sober.*”

For example

```
string => "I know nothing except the fact of my ignorance"

> [5, 9, 32, 41]
```

8. Write a program that count words in sentence

”*All programming is maintenance programming,*

because you are rarely writing original code.” (*DaveThomas*)

9. Write a program split string-sentence by space delimeter into an array of words

”*He who licks knives will soon cut his tongue*”

For example

```
string => "He is guilty who is not at home"

> ["He", "is", "guilty", "who", "is", "not", "at", "home"]
```

10. Write a program that convert all symbols in string into ASCII symbols and calculate sum of all values

”*Keep fire away from straw*”

For example

```
string => "ABc"

’A’ -> 65
’B’ -> 66
’c’ -> 99
65 + 66 + 99 = 230

> 230
```

11. Write a program that validate algebraic equation (use simple validation, just count opened and closed parentheses)

$$(a+b)*(b+c)=0$$

$$(a+b)*(b+c)-(a+4+r))=0$$

For example

```
string => "(a + b) = 0"
> true
string => "(a + b) + c) = 0"
> false
```

12. Write a program that find the shortest and the longest word in a sentence

”*No matter how hard you try the bull will never give milk*”

For example

```
string => "No cook ever died of starvation"
> No
> starvation
```

13. Write a program that check how many numeric symbols contains in a string

”*iuqheiuehrg7jh34hj234jh5y287b23jhbhj34thj2b34thj*”

For example

```
string => "asdfj345jkjh2k2" => 34522 => 5
> 5
```

14. Write a program that count how many uppercase symbols in a string

”*sdgHjhgJHGHJggSLsjLKSkjhkjhSjhkSkhsk*”

For example

```
string => "abcEljkR"
> 2
```

15. Write a program that count how many lowercase symbols in a string

”*JHGJHGhjgHGUYGUYSJJKhjhkjUIUIHJhjhKJHGK*”

For example

```
string => "ABCdfKILFp"
> 3
```

16. Write a program that count how many vowels in a string

”*asdjfhwuqyetrnzxbcvcpoxincumbniuoqwyenzxcbmzvnbcqtyrenbvzcpoiedsfmngb*”

(in english alphabet there are 6 vowels letters - a, e, i, o, u, y)

For example

```
string => "qwyen"
> 2
```

17. Write a program that count how many consonants in a string

”*sdkfjgskjdfnvjksdfngjkasdafkbvkamsndiugkhbahjdfdbncva*”

For example

```
string => "jkasd"
> 4
```

18. Write a program that detect if string is a palindrome

”*Anna*”

”*Kayak*”

”*Racecar*”

”*git*”

”*maven*”

For example

```
string => "Civic" (a palindrome)
string => "programming" (not a palindrome)
```

19. Write a program that insert **1** after each letter (spaces must be ignored)

”*I want to become a programmer*”

For example

```
string => "Programming is fun"
> "P1r1o1g1r1a1m1m1i1n1g1 i1s1 f1u1n1"
```

20. Write a program that replace all occurrences of ’a’ character with string ”test”

You can′t unscramble a scrambled egg

21. Write a program that insert a dot after each word

”*A bird in hand is worth two in the bush*”

For example

```
string => "Hello world"
> "Hello. world."
```

22. Write a program that revers all symbols in string

”*Flies will not land on a boiling pot*”

For example

```
string => "String"

> "gnirtS"
```

23. Write a program that revers symbols in every word

A chain is only as strong as its weakest link

For example

```
string => "Hello world"

> "dlrow olleH"
```

24. Write a program that swap symbols group of 2 in string

”*I was really too honest a man to be a politician and live*”

For example

```
string => "Envy is the ulcer of the soul"

> "nEyv si hte luecr of hte oslu"
```

25. Write a program that convert input string into two strings. First string must contain only odds symbols of input string and second - evens.

”*wejhbnpolkdshjnjwe*”

For example

```
string => "abcd"

> "ac"
> "bd"
```

26. Write a program that take fist symbol of every word in sentence and add to new string

”*Love tells us many things that are not so*”

For example

```
string => "Love will find a way"
> "Lwfaw"
```

27. Write a program that replace all spaces in a string by symbol **’-’**

”*Laziness is your best friend. Never do twice what you can automate once*”

28. Write a program that replace the second occurrence of **’o’** symbol in a string by symbol **’!’**

”*Debugging becomes significantly easier if you first admit that you are the problem*”

29. Write a program that convert the fist symbol in string to upper case for every word

”*Fire begins with sparks*”

For example

```
string => "Software developer"

> "Software Developer"
```

30. Write a program that convert the fist and the last symbol in string to upper case for every word

”*Drunkards know no danger*”

For example

```
string => "Big mind"

> "BiG MinD"
```

31. Write a program that convert the second and the last but one symbol in string to upper case for every word

”*Black souls wear white shirts*”

For example

```
string => "I cannot teach anybody anything. I can only make them think

> "I cAnnOt tEaCh aNyboDy aNythiNg. I can oNLy mAKe tHEm tHiNk"
```

32. Write a program that remove every second character from the string

Absence makes the heart grow fonder

33. Write a program that keep in a string only vowel characters

A cat has nine lives

34. Write a program that remove extra spaces in string

If debugging   is the process of   removing software   bugs, then   programming must

be the   process   of putting   them   in”

(Edsger Dijkstra)

35. Write a program that remove trailing and leading white spaces in a string

What is the bigger than the universe?

36. Write a program that remove all duplicate characters except first

”*Always put your best foot forward*”

For example

```
string => "Hello world"

> "Helo wrd"
```

37. Write a program that remove the last occurrence of character ’a’ from a string

You show me the man and I′ll show you the rule

38. Write a program that find from which index begin substring **”for”** in the string

”*performance*”

For example

```
string => "java" (substring "va")
> 2 (substring "va" begin in string by index 2)
```

39. Write a program that find the longest word in a sentence and sort all symbols of the word in asc using ASCII codes

”*Facebook wasn′t built in a day*”

40. Write a program that find index of the first occurrence of character **'f'** in string

”*A programming language is for thinking about programs, not for expressing programs you′ve already thought of. It should be a pencil, not a pen*”

(Paul Graham)

41. Write a program that find index of the last occurrence of character **'p'** in string

”*Correctness is clearly the prime quality. If a system does not do what it is supposed to do, then everything else about it matters little*”

(Bertrand Meyer)

42. Write a program to find the most frequent character in a given string.

You can′t make an omelette without breaking eggs

43. Write a program that find the second most frequent character in a given string.

```
"containing"
```