

Задачи:

1. Да се напише функция **max()**, която приема 2 цели положителни числа и връща по-голямото.

Примерен вход и изход:

max(2, 3) -> 3

max(5, 20) -> 20

max(3, 3) -> 3

2. Да се напише функция, която по подадени реални положителни числа, които съответстват на цена и процент отстъпка, връща като резултат намалената цена.

Примерен изход при вход 100 50 -> 50lv, 30 10 -> 27lv.

3. Да се напише функция **isLetter()**, която проверява дали един символ е буква.

Примерен вход и изход:

isLetter('c') -> true

isLetter('#') -> false

isLetter('K') -> true

4. Да се напише функция **isDigit()**, която проверява дали един символ е цифра.

Примерен вход и изход:

isDigit('3') -> true;

isDigit('F') -> false

5. а) Да се напише функция **pow()**, която приема две цели положителни числа и връща първото на степен второто.

Примерен вход и изход:

pow(2,3) -> 8

pow(7,0) -> 1

pow(4,5) -> 1024

б) Да се напише функция **sqrt**, която приема цяло положително число и намира неговия корен

Примерен вход и изход: 25 -> 5, 81 -> 9. 27 -> -1

6. Да се напише програма, която по подадени 3 положителни числа, проверява дали те могат да бъдат страни на триъгълник. Ако да, да се определи видът му.

Примерен вход и изход: 1 2 3 -> "There is no such triangle!", 3 4 5 -> "Right- angled triangle", 7 8 9 -> "Acute- angled triangle", 6 7 10 -> "Obtuse-angled triangle"

7. Да се напише функция **digitsSum()**, която намира сбора на цифрите на дадено естествено число.

Примерен вход и изход:

digitsSum(10) -> 1

digitsSum(100) -> 1

digitsSum(123456789) -> 45

8. Да се напише функция **DecToBin()** , която принтира двоичния запис на число.

Примерен вход и изход: 10 -> 1010, 15 -> 1111, 20 -> 10100