

Задачи:

1. Да се напише функция **swapValues()**, която приема две цели положителни числа и разменя стойностите им.

Примерен вход и изход: *swapValues(2,3) -> 3,2*

2. Да се напише функция **reverse()**, която приема като аргументи масив и размера му и връща масив, с елементите на подадения в обратен ред:

Примерен вход и изход: *{ 203, 189, 41, 900, 666 } -> { 666, 900, 41, 189, 203 }*

3. Да се напише функция **printMatrix()**, която печата на конзолата елементите на двумерен масив и използва адресна аритметика.

4. Да се напише функция, която по дадени указатели към двумерни масиви от цели числа и техните размери, записва във втория матрицата, получена от транспонирането на матрицата, представена от първия масив.

Примерен вход и изход:

123	147
456	-> 258
789	369

5. Да се напише функция **findDigit()**, която, по даден масив от цели числа (и неговия размер) и цяло число, търси число в масива и ако го намери, връща указател към първото му срещане, в противен случай връща `nullptr`.

Примерен вход и изход:

{16, 74, 32, 95, 28}, 3 -> nullptr;

{16, 74, 32, 95, 28}, 32 -> adress of 32

6. Да се напише функция **digitSort()**, която по даден масив и размера му, ще сортира елементите на масива във възходящ ред според сбора на цифрите им. Ако две числа имат равен сбор на цифрите си, то по-напред ще е по-малкото число.

Примерен вход и изход: *{ 203, 189, 41, 900, 666 } -> { 41, 203, 900, 189, 666 }*

7. Да се напише функция **merge()**, която , по подадени указатели към два сортирани масива от цели числа и съответните им размери и по подаден трети масив , слива

първите два масива в третия , така че полученият масив отново да е сортиран , без да се прилага допълнително сорниране .

Примерен вход и изход:

$\{1, 7, 8\}, \{2, 6, 9\} 3, 3 \rightarrow \{1, 2, 6, 7, 8, 9\}$