

Упражнение върху масиви и низове

# Задача 0:

Напишете програма с името:

- a) **Task0a\_CreateArray**, която да създава масив от тип **int** и да го инициализира със стойностите 5, 9, 11, 3, 6, 4, 7. Отпечатайте стойностите на екрана, всяка на нов ред.
- b) **Task0b\_ReadArray**, която да създава масив от тип **int** и да го инициализира със стойности, въведени от стандартния вход. На първия ред на стандартния вход ще бъде въведено число N, което ще указва броя елементи, които ще бъдат въведени след това. На втория ред на стандартния вход ще бъдат въведени N на брой числа, разделени с интервал. Нека създаденият масив да има големина точно N. Отпечатайте въведените числа, разделени със запетая, на стандартния изход.
- c) **Task0c\_PrintString**, която да чете от стандартния вход един символен низ и да го отпечата на екрана, като всяка буква е на нов ред.
- d) **Task0d\_Print2dArray**, която да отпечата на стандартния изход двуизмерен масив 4x4, инициализиран със стойностите на числата от 1 до 16.

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

# Задача 1: PrintMatrix

Попълнете и принтирайте на екрана матрица NxN с числата от 1 до  $N^2$ , използвайки цикли.

Числото N е въведено от стандартния вход, а на изхода трябва да се принтира двумерния масив, попълнен в следния формат:

a)

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

b)

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

c)

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

d) \*

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

Програмите да се казват съответно **Task1a\_PrintMatrix**, **Task1b\_PrintMatrix**, **Task1c\_PrintMatrix**, **Task1d\_PrintMatrix**.

## Задача 2:

По въведено N и N на брой числа, напишете програма:

- a) **Task2a\_PrintLargestNumber**, която да изведе най-голямото число.

Вход:	Изход:
11 21 8 11 -1 -3 7 4 19 5 -6 0	21

- b) **Task2b\_PrintSortedNumbers**, която да изведе числата наредени по големина.

Вход:	Изход:
11 21 8 11 -1 -3 7 4 19 5 -6 0	-6 -3 -1 0 4 5 7 8 11 19 21

- c) **Task2c\_PrintClusterCount**, която да изведе броя на клъстерите (клъстер е последователност от елементи с един или повече идентични елемента).

Вход:	Изход:
14 1 1 2 3 3 5 6 7 9 9 9 10 10	4

- d) **Task2d\_PrintSplitIndex**, която да изведе позицията (индекса) в масива, където сбора на всички числа отляво до тази позиция (включително) е равен на сбора на елементите отдясно (изключвайки тази позиция). Ако няма такава, програмата да изведе "NO".

Вход:	Изход:
6 1 2 3 4 9 1	3

- e) **Task2e\_LongestIncreasingSubsequence**, която да изведе най-дългата растяща подредица от последователни елементи.

Вход:	Изход:
10 9 8 1 2 6 11 4 8 9 3	1 2 6 11

- f) **Task2f\_LargestSumOfKSubsequence**, която да изведе подредицата от последователни елементи с дължина K и с най-голяма сума. K също се чете от стандартния вход след N и преди редицата от N елемента.

Вход:	Изход:
10 5 9 8 1 2 6 11 4 8 9 3	6 11 4 8 9

g) \* **Task2g\_LargestSumSubsequence**, която да изведе подредицата с най-голяма сума.

Вход:	Изход:
12 4 7 -3 -11 2 9 -4 5 6 -8 0 5	2 9 -4 5 6

h) \* **Task2h\_PrintOddFrequencyNumber**, която да изведе „самичкото“ число, ако допуснем че в редицата всяко число ще се среща четен брой пъти, освен едно, което ще се среща нечетен брой пъти.

Вход:	Изход:
15 4 6 4 -3 5 2 5 4 5 6 -3 2 5 4 6	6

i) \*\* **Task2i\_FindKInSorted**, която да изведе индекса на число K, ако е в подредицата или да отпечата “NO”, ако го няма. Предположете че редицата е подредена по големина във възходящ ред и че K също се чете от стандартния вход след N и преди редицата от N елемента.. Опитайте се да направите програмата така, че дори и в най-лошия случай да намери отговора без да обхожда всички елементи.

Вход:	Изход:
12 4 -11 -8 -4 -3 0 2 4 5 5 6 7 9 4	6
9 11 1 2 3 4 5 6 7 8 9	NO

## Задача 3:

По въведен низ, напишете програмата:

- а) **Task3a\_IsPalindrome**, която проверява и извежда true или false, ако низът е палиндром (чете се от ляво надясно и от дясно наляво по един и същ начин)

Вход:	Изход:
alabala	true
abbaabba	true

- б) **Task3b\_PrintLetterCount**, която извежда броя на срещане на всяка буква без значение малки и големи. Предположете че в низа има само латински букви.

Вход:	Изход:
Programming	a(1) g(2) i(1) m(2) n(1) o(1) p(1) r(2)

- в) **Task3c\_CaesarCipher**, която кодира или декодира низа и изпечатва резултата на екрана. При кодиране всяка буква се заменя със следващата в азбуката, като 'z' се заменя с 'a'. Програмата кодира, ако на реда следващ низа е въведено "encode" и декодира, ако е въведено "decode".

Вход:	Изход:
I love programming. encode	J mpwf qsphsbnnjoh.
J ibuf tusjoht opx. decode	I hate strings now.

- г) **Task3d\_BracketMatching**, която отпечатва true, ако скобите са поставени правилно и false в обратния случай. Предположете че низа се състои само от следните символи "()[]{}".

Вход:	Изход:
(([[{({})}]]))	false
([]({}){[]})	true

- д) **Task3e\_ConvertibleStrings**, която проверява дали чрез задраскване на букви може да се получи друг низ, който е въведен от стандартния вход на следващия ред.

Вход:	Изход:
I love orange t-shirts I love oranges	true

- f) **Task3f\_LongestCommonSubstring**, която извежда най-дългата символна подредица, която е обща за въведения низ и още един, въведен на следващия ред на стандартния вход.

Вход:	Изход:
ahasjaaakxmza ahaooxmaaakxs	aaakx

## Задача 4: PersonCharacteristics

Напишете програма **Task4\_PersonCharacteristics**, като за основа ползвате задачата **PersonCharacteristics** от предния път. Добавете:

- Променлива **gender** от тип **char**, в която да се пази, въведен от клавиатурата пола на лицето – М за мъж и F за жена
- Масив, в който да се пазят, въведени от клавиатурата 4 оценки по шестобалната система (десетични числа от 2 до 6)

Програмата чете от стандартния вход число N, след което N на брой реда в следния формат:

<първо\_име>;<последно\_име>;<пол>;<година\_на\_раждане>;<тегло>;<височина>;<професия>;<оценка1>;<оценка2>;<оценка3>;  
<оценка4>

Изхода на програмата трябва да бъде N на брой съобщения във формата:

<първо име> <последно име> **is** <години> **years old**. <His/Her> **weight is** <тегло> **and** <he/she> **is** <височина> **cm tall**. <He/She> **is a**  
<професия> **with an average grade of** <средна оценка>. [<първо име> <последно име> **is under-aged**.]

Като местоименията *he/she/his/her* се слагат според пола, средната оценка се пресмята, теглото се отпечатва с 1 знак след десетичната запетая, а оценката с 3.

*Пример:*

Вход:
3 Peter;Petrov;M;1985;89.9;184;miner;4.5;4;3.75;4.25 Ivan;Ivanov;M;2000;71.2;177;student;5;4;6;5 Mariya;Atanasova;F;1992;51.6;160;secretary;6;6;5.75;5.5
Изход:
Peter Petrov is 31 years old. He was born in 1985. His weight is 89.9 and he is 184 cm tall. He is a miner with an average grade of 4.125. Ivan Ivanov is 16 years old. He was born in 2000. His weight is 71.2 and he is 177 cm tall. He is a student with an average grade of 5.000. Ivan Ivanov is under-aged. Mariya Atanasova is 24 years old. She was born in 1992. Her weight is 51.6 and she is 160 cm tall. She is a secretary with an average grade of 5.813.