Домашна работа 2 - ИС УП

Решенията на задачите да се предадат като отделни .cpp файлове в архив.

Задача 1:

Да се напише програма, която по подаден символен низ с дължина n, репрезентиращ римска цифра (XVI), да го превърне в съответната арабска цифра.

V	-	5
XIV	-	14
XX	-	20

Да се направят коректни валидации, които не позволява на потребителя да въведе невалидна римска цифра, както и да се спази лимита от I до M.

Задача 2:

Нека имаме масив от указатели към масиви с целочислени елементи

с размери n >= 3.

Да се напише функция, която връща най - добре сортирания (във възходящ ред) масив

(При открити два или повече напълно сортирани масива да се върне онзи с най - много елементи)

Да се напише функция, която връща най - разбъркания масив.

(При открити два или повече напълно разбъркани масива да се върне онзи с най - много елементи)

Пример при получаване на масив от указатели към следните масиви. {1, 2, 3} (сортиран на 100%, съдържа 3 елемента), {111, 453, -1} (сортиран на 33%, съдържа 3 елемента), {1, 2, 3, 4} (сортиран на 100%, съдържа 4 елемента).

```
Best sorted array - {1, 2, 3, 4}
Worst sorted array - {111, 453, -1}
```

<u>Задача 3:</u>

Да се напише функция, която приема символен низ, след което премахва всички символи

освен 0, 1, &(логическо и), |(логическо или) и =.След това проверява дали низът е валиден

булев израз. Един булев израз е валиден когато изпълнява условието: <0 или 1> <& или |>

<0 или 1>... <0 или 1> <& или |> <0 или 1> = <0 или 1>..

*За символния низ приемаме, че няма да бъде по - дълъг от 100 символа.

Вход 1

"dsa1&1&s0s=0"

Изход 1

Boolean expression 1&1&0=1 valid!

1&1&0=1 – всичко изглежда коректно (освен сметката)

Вход 2

"dsa1&&1&s1s=0"

Изход 2

Boolean expression 1&&1&0 = 1 is NOT valid!

1&&1&0=1 (две логичски и-та не изглежда доста коректно)

Вход 3

"asd0|1|0=1"

Изход 3

Boolean expression 0|1|0=1 valid!

0|1|0=1

Вход 4

"asd0|1|0dsad"

Изход 4

Boolean expression 0|1|0 is NOT valid!

0|1|0 т

*Бонус 1 – направете програмата, така че да работи с двойни оператори && и ||, т.е

1 & 1 & 0 = 1, да не е валиден израз, а само изрази от вида 1 & 8 1 & 0 = 1 да са.

*Бонус 2 - направете функцията рекурсивна и нека след изчистване на израза между

числата и операторите да има по точно 1 интервал за красота, т.е от първия вход 1&1&0

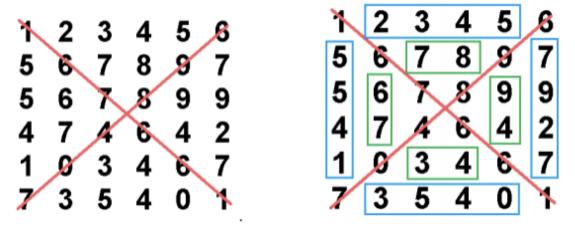
= 1 да стане 1 & 1 & 0 = 1

Задача 4:

Да се напише функция, която приема като аргумент квадратна матрица и нейния размер (размерът трябва да е >= 3), след което за елементите, които НЕ са част от двата й диагонала да се направи следната проверка: Редиците, които се

образуват от тези елементи трябва да са еднакви като горните редици се гледат отляво надясно, а страничните редици трябва да се гледат отгоре надолу.

Пример за разяснение:



За да бъде изпълнена проверката трябва

$$\{2, 3, 4, 5\}, \{5, 5, 4, 1\}, \{7, 9, 2, 7\}, \{3, 5, 4, 0\},$$

да са равни, освен това

също трябва да излезнат равни.

Примерни входове

Вход 1					В	Вход 2				Вход З	
1	2	3	4	5	6	1	2	3	4	5	5 2 4
5	6	7	8	9	7	2	6	6	8	2	212
5	6	7	8	9	9	3	6	7	6	3	4 2 7
4	7	4	6	4	2	4	7	6	6	4	
1	0	3	4	6	7	1	2	3	4	6	
7	3	5	4	0	1						
Изход 1					И	Изход 2				Изход 3	
False					Ti	True				True	

^{*}Бонус - направете програмата, така че размера на матрицата да се задава от клавиатурата и точно толкова памет да се задели за нея.