

Задачи за подготовка за Контролно 2

Забележка. Всички изброени задачи са давани на второ контролно в курса по УП в предишни години.

Задача 1. Дадена е квадратна матрица от естествени числа с размерност $n \times n, n \in [2; 20]$. Да се напише функция, която сортира редовете на матрицата в низходящ ред според сумата на цифрите във всеки от елементите. Да се напише функция, която проверява дали съществува диагонал (с повече от три елемента), чиито елементи образуват числова редица с общ член $n(n+1)/2$. Забележка. Ако матрицата има диагонал с елементи: 15, 21, 28, 36, то резултатът от изпълнението на програмата трябва да е true, тъй като редицата 15, 21, 28, 36 е получена от общия член при стойности 5, 6, 7 и 8.

Задача 2. Дадена е матрица с големина 5×5 , някои от елементите на която са предварително запълнени със стойности (в интервала от 1 до 9). Играта “Скука” има следните правила – дадената матрица се дозапълва (с въвеждане от клавиатура) с цели числа от интервала $[1, 9]$, като играта се счита за спечелена, ако след запълване на матрицата, сумата на елементите в i -ти диагонал (за всяко $i, i = 1, \dots, 9$) е кратна (дели се без остатък) на номера на диагонала. Диагоналите са номерирани както е показано в таблицата по-долу.

Да се реализират следните аспекти от играта:

- а) подходящо извеждане на текущата матрица на екрана (включително след прочитане на стойност за дадено поле);
- б) четене от клавиатура на координати на поле и стойност на поле, като ако полето не е от предварително зададените, стойността му се променя на текущата;
- в) автоматична проверка за спечелване на играта, след запълване на всички полета на матрицата и извеждане на подходящо съобщение;

Номера на диагоналите				
1	2	3	4	5
2	3	4	5	6
3	4	5	6	7
4	5	6	7	8
5	6	7	8	9

Начална матрица				
8		1		
3			2	
	6			4
		7		
			5	9

Задача 3. Да се напише рекурсивна функция, която намира числото $ху$, по въведени сбор $х+у$ от цифрите $х$ и $у$ и разликата между числата $ух - ху$.

Пример:

Вход: Сбор $х + у = 12$; Разлика $ух - ху = 36$.

Изход: 48

Задача 4. Даден е символен низ с дължина не повече от 100 символа. Да се напише функция, която записва в масив от цели числа всички числа, които могат да се прочетат в низа. Число е последователност от символи, които представляват цифри. Допуска се да има водещ знак (+/-).

Пример:

Вход: "В 9 ч. тръгвам към ФМИ с автобус 94 или 120." Резултат: Масив с елементи 9, 94 и 120.

Задача 5. Да се напише програма, която въвежда от клавиатурата символен низ X с най-много 255 символа. Нека низът s е съставен само от малките латински букви на низа X в реда на срещането им в X , а низът S – само от главните латински букви на низа X в реда на срещането им в X . Програмата да извежда „Yes“, ако низовете s и S се състоят от съответни една на друга букви, например „abc“ и „ABC“. Пример за такъв низ X е низът „abABcC“, но не и низът „aBACbc“. В противен случай програмата да изписва „No“.

Задача 6. Да се напише функция, която получава като параметри матрица от числа във вид на двумерен масив и предикат с един числов аргумент. Функцията да извежда на екрана сумата на числата в редовете и колоните, в които броят на елементите, удовлетворяващи предиката, е максимален за матрицата.

Пример:

Нека е дадена матрицата

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

и предикатът:

```
bool isEvenAndLessThan6(int x) { return x % 2 == 0 && x < 6; }
```

Функцията трябва да извежда на екрана числото 19, което е сумата на реда 1+2+2+4 и на колоната 2+4+4, тъй като и в двете има по три числа, удовлетворяващи предиката.

Да се напише рекурсивна функция, която намира числото $xу$, по въведени сбор $x+y$ от цифрите x и y и разликата между числата $yx - xy$.

Пример:

Вход: Сбор $x + y = 12$; Разлика $yx - xy = 36$.

Изход: 48 **Задача 7.** Дадена е квадратна матрица с размери $n \times n$, $n \in [1, 10]$ която описва лабиринт. Стойност 0 в дадена клетка означава „стена“, стойност 1 означава „свободно място за движение“. Даден е низ съдържащ само буквите E(east), W(west), N(north) и S(south), които указват едностъпкови придвижвания в съответните географски посоки.

Да се напише функция, която проверява дали даденият низ е валиден път от някоя проходима клетка в лабиринта до долния десен ъгъл в лабиринта.