



УВОД В ПРОГРАМИРАНЕТО – УПРАЖНЕНИЕ №9

30.11.2023

СОФТУЕРНО ИНЖЕНЕРСТВО, ГРУПА 4

АСИСТЕНТ: ЕЛЕНА ТУПАРОВА



ЗА КАКВО ЩЕ СИ ГОВОРИМ ДНЕС?

- Многомерни масиви
- Указатели и указателна аритметика
- Указатели и масиви

КАКВО Е ДВУМЕРЕН МАСИВ?

- На практика това е масив от масиви
- Можем да си го представяме като матрица

```
int matrix1 [2][3]; //матрица с 2 реда и 3 колони
```

```
int matrix2[2][3] = {{ 1, 2, 3 }, { 4, 5, 6}};
```

```
int matrix3[2][3] = {1, 2, 3, 4, 5, 6};
```

```
int matrix4[][3] = {1, 2, 3, 4, 5, 6};
```

ЗАДАЧА 1

- Да се напише програма, която събира матриците $A_{n \times m}$ и $B_{n \times m}$ ($1 \leq n \leq 20$, $1 \leq m \leq 30$).

ЗАДАЧА 2

- Да се напише програма, която намира сумата на елементите на квадратна матрица $A_{n \times n}$ ($1 \leq n \leq 20$), които се намират:
 - по главния диагонал;
 - над главния диагонал.

ЗАДАЧА 3

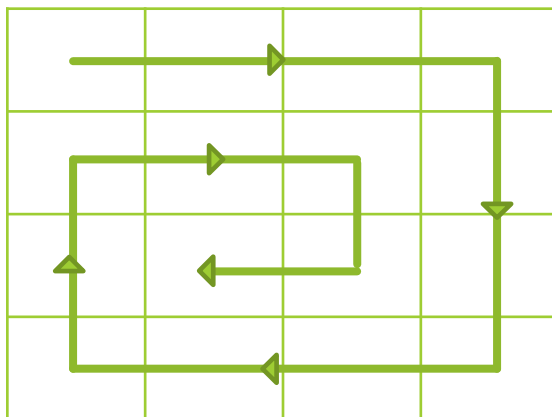
- Да се напише програма, която линеаризира дадена матрица $A_{n \times m}$, т.е. я представя в едномерен масив ред след ред.

ЗАДАЧА 4

- Дадена е реална правоъгълна матрица A с размерност $n \times m$ ($1 \leq n \leq 10$, $1 \leq m \leq 20$). Да се напише програма, която изтрива k -тия стълб на A ($0 \leq k < m$).

ЗАДАЧА 5

- Да се напише програма, която обхожда матрица $A_{n \times n}$ спираловидно и извежда елементите ѝ в този ред.





N-МЕРНИ МАСИВИ

УКАЗАТЕЛИ И РЕФЕРЕНЦИИ

```
int number = 5;
```

```
int& referenceToNumber = number; // друго име на същата променлива  
std::cout << referenceToNumber << std::endl;
```

```
int* pointerToNumber = &number; // пази адреса на променливата number  
std::cout << pointerToNumber << std::endl;
```

```
std::cout << *pointerToNumber << std::endl;
```

УКАЗАТЕЛИ КЪМ КОНСТАНТА != КОНСТАНТНИ УКАЗАТЕЛИ

const int* != int* const

Указател към константа

Сочи към константа, т.е. не може да се променя стойността ѝ
Алтернативен запис - `int const*`

Константен указател

Веднъж свързан с дадена променлива, не може да се „пренасочва“ към друга променлива (като референция)
Може да се променя стойността на самата променлива

УПОТРЕБА

- Промяна на параметри във функции
- Указателна аритметика
- Указатели и масиви
- Указатели и символни низове – следващата седмица

УКАЗАТЕЛНА АРИТМЕТИКА

```
int a = 4;
```

```
int *p = &a;
```

```
p = p + 1; // ще „премести“ напред указателя с една  
„стъпка“, голяма колкото е размерът на типа, към който сочи  
указателят
```

УКАЗАТЕЛИ И МАСИВИ

- Указатели и едномерни масиви

```
int a[100]; // a -> указател към a[0]
```

```
*a == a[0];
```

```
*(a+1) == a[1]; ... *(a+n) == a[n];
```

- Указатели и двумерни масиви

```
int a[10][20]; // a -> указател към първия елемент на едномерния  
масив [a[0], a[1], ... a[9]], а всяко a[i] е указател към a[i][0]
```

```
**a == a[0][0]; *a == a[0];
```

```
a[i][j] == (*(a+i)+j);
```