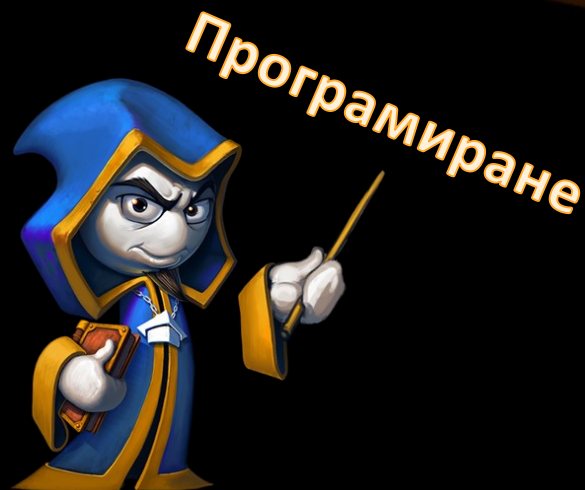


Многомерни масиви

Масиви с повече размерности



```
int[,] intMatrix =  
{  
    {2, 8, 3, 5},  
    {7, 9, 0, 3},  
};
```



Учителски екип

Обучение за ИТ кариера

<https://it-kariera.mon.bg/e-learning/>

<https://github.com/BG-IT-Edu/School-Programming/tree/main/Courses/Applied-Programmer/Programming-Fundamentals>

Съдържание

1. Какво е многомерен масив?
2. Двумерни масиви – матрици
3. Многомерни масиви – размерности, индексирание



Какво е многомерен масив?

- **Двумерен масив** = таблица;
 - Всеки елемент се идентифицира чрез две измерения – номер на реда и номер на колоната в таблицата
- **Многомерен масив** = Аналогично можем да имаме допълнителни измерения в масива. В този случай е удобно да си представяме масив, в който всеки елемент е масив с по-ниско измерение:
 - Двумерен масив = масив, чиито елементи са едномерен масив
 - Тримерен масив = масив, чиито елементи са двумерни масиви
 - Четиримерен масив = масив, чиито елементи са тримерни масиви
- Основните правила, които важат за едномерни масиви важат и за многомерни

Двумерен масив – матрица

- Има `rows` x `columns` на брой елементи, където `rows` е брой на редовете, а `columns` – на колоните
- Размера на масива е постоянен по всяко негово измерение – не се променя
- Елементите са от **един и същ** тип
- Елементите във всяко измерение са номерирани с два индекса:
 - Ред – от 0 до **`rows-1`**
 - Колона – от 0 до **`columns-1`**

Деклариране на многомерен масив

- Досега едномерен масив от цели числа декларирахме чрез `int[]`, двумерен масив бихме декларирали по следния начин:

```
int[,] twoDimentionalArray;
```

- Аналогично тримерен масив бихме декларирали така:

```
int[, ,] threeDimentionalArray;
```

- Няма теоретично ограничение за броя на размерностите на масив, но в практиката масиви с повече от 2 размерности са рядко срещани

Деклариране и заделяне

- Отбелязването на променливата като многомерен масив само по себе си не заделя памет за неговите елементи. За целта използваме **new**:

```
int[,] intMatrix = new int[3, 4];
```

Двумерен масив от цели числа с 3 реда и 4 колони на ред

```
float[, ,] floatCube = new float[5, 5, 5];
```

Тримерен масив от 5 елемента, като всеки е двумерен масив с 5 реда и 5 колони

Инициализация на двумерен масив

- Както при едномерените масиви можем да зададем стойности на многомерния масив веднага след деклариране:

```
int[,] intMatrix =  
{  
    {2, 8, 3, 5},  
    {7, 9, 0, 3},  
};  
//двумерен масив 2 x 4 (2 реда, 4 колони)
```

Достъп до елементите на многомерен масив

- Както при едномерните масиви, така и при многомерните всички индекси започват от 0. Разликата е, че тук индексите са повече от 1.
- Ето как да достъпим елементите на примера от предния слайд:

```
intMatrix[0, 0]
```

```
intMatrix[0, 1]
```

```
intMatrix[0, 2]
```

```
intMatrix[0, 3]
```

```
intMatrix[1, 0]
```

```
intMatrix[1, 1]
```

```
intMatrix[1, 2]
```

```
intMatrix[1, 3]
```

- Индексите се отделят със запетай!

Дължина на многомерен масив

- Всяка размерност на многомерен масив може да има различна дължина спрямо останалите. Поради тази причина всяка размерност се номерира по сходен начин на индексите.
- За да разберем колко реда има двумерния масив от примера:

```
intMatrix.GetLength(0);
```

- А за да разберем колко колони има:

```
intMatrix.GetLength(1);
```

Пример: Отпечатване на матрица

```
int[, ] intMatrix =  
{  
    {2, 8, 3, 5},  
    {7, 9, 0, 3},  
};
```

```
for(int row = 0; row < intMatrix.GetLength(0); row++)  
{
```

Брой редове

Брой колони

```
    for(int col = 0; col < intMatrix.GetLength(1); col++)  
    {
```

```
        Console.Write(intMatrix[row, col]+" ");
```

Изпечатване на
елемента

```
    }
```

```
    Console.WriteLine();
```

Прехвърляме
се на нов ред

```
}
```

Пример: Вход/Изход на матрица

```
int rows = int.Parse(Console.ReadLine()); //въвеждаме брой редове
int cols = int.Parse(Console.ReadLine()); //въвеждаме брой колони
int[,] matrix = new int[rows, cols]; //заделяме съответния брой
елементи

for(int row = 0; row < rows; row++)
{
    for(int col = 0; col < cols; col++)
    {
        Console.Write("matrix[{0},{1}] = ", row, col);
        matrix[row, col] = int.Parse(Console.ReadLine());
    }
}
//TODO: отпечатваме елементите на масива като в предния пример
```

Задача: Средноаритметично по редове

- Напишете програма, която обработва двумерен масив.
 - Изведете го като в края на всеки ред добавете средноаритметичното от реда.
 - Всеки елемент да заема 8 позиции

Тествайте в Judge: <https://judge.softuni.bg/Contests/2667>

Пример: Средноаритметично по редове

2
4
1
2
3
5
8
6
9
4



| | | | | |
|---|---|---|---|------|
| 1 | 2 | 3 | 5 | 2.75 |
| 8 | 6 | 9 | 4 | 6.75 |

Тествайте в Judge: <https://judge.softuni.bg/Contests/2667>

Решение: Средноаритметично по редове

```
// TODO: Въведете матрицата
for (int row = 0; row < rows; row++) {
    double avg = 0;
    for (int col = 0; col < cols; col++) {
        Console.Write("{0, 8}", matrix[row, col]);
        avg += matrix[row, col];
    }
    avg = avg / cols;
    Console.WriteLine("{0, 8}", avg);
}
```

В началото на
всеки ред,
зануляваме

Извеждаме елемента,
заемащ 8 позиции

Изчисляваме
средноаритметично
и го извеждаме

Тествайте в Judge: <https://judge.softuni.bg/Contests/2667>

Какво научихме този час?

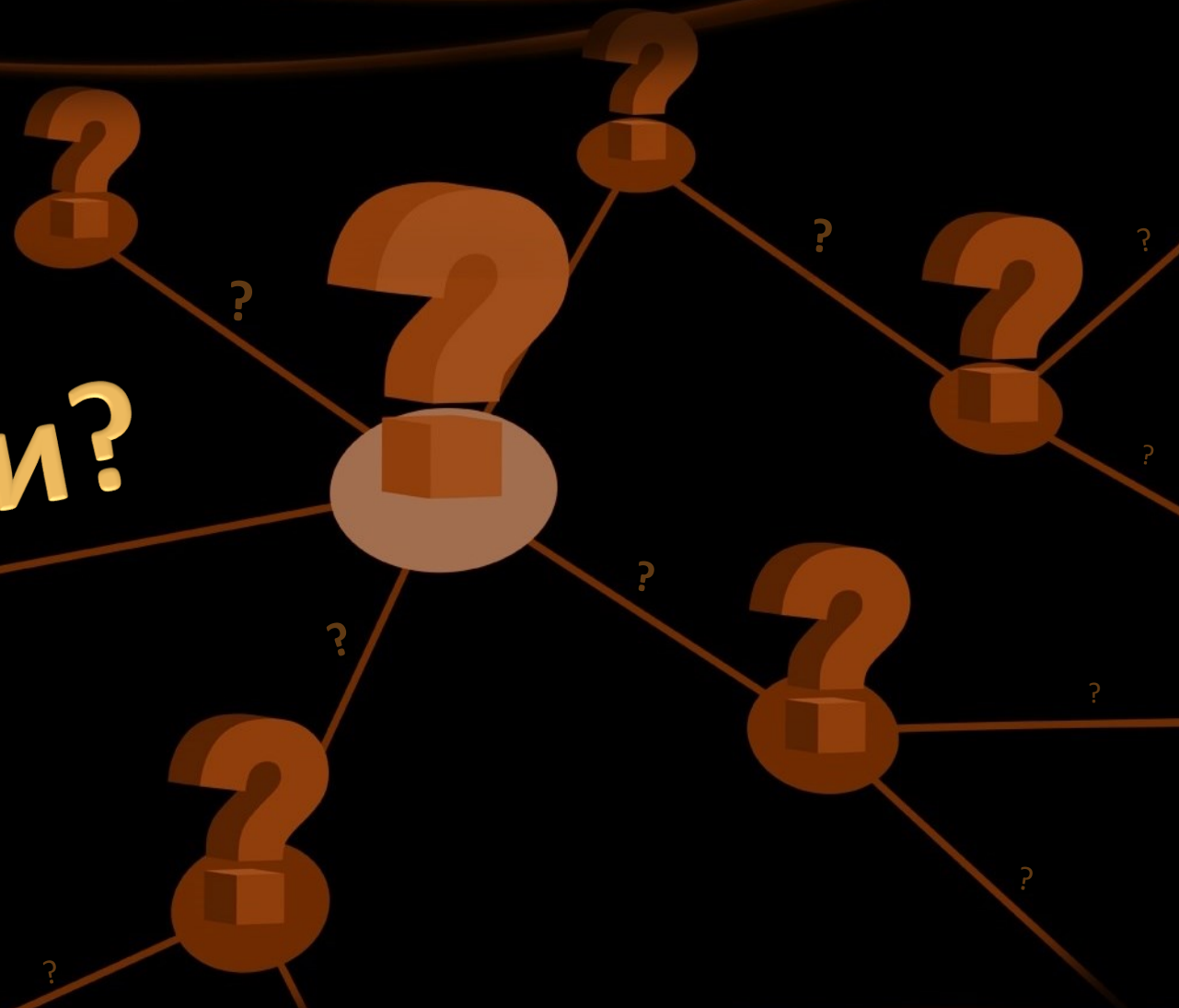
- Многомерните масиви са масиви, чиито елементи са също (многомерни) масиви
- Многомерните масиви разполагат с повече от една размерност и с толкова индекси, колкото размерности имат
- Памет за многомерните масиви се заделя чрез `new` като посочим за всяка размерност колко елемента трябва да има



Масиви



Въпроси?



Министерство на образованието и науката (МОН)

- Настоящият курс (презентации, примери, задачи, упражнения и др.) е разработен за нуждите на Национална програма "**Обучение за ИТ кариера**" на МОН за подготовка по професия "Приложен програмист"



Министерство
на образованието
и науката



Национална
програма
„Обучение за
ИТ кариера“

- Курсът е базиран на учебно съдържание и методика, предоставени от **фондация "Софтуерен университет"** и се разпространява под свободен лиценз **CC-BY-NC-SA**



SoftUni
Foundation

