

# Масиви и указатели

гл. ас. д-р. Нора Ангелова

---

# Указатели и масиви

---

- Едномерни масиви
- Двумерни масиви

# Указатели и масиви

```
int arr[40] = {1, 2, 3, 4};
```

arr – указател към първи елемент на масив.

arr – съдържа адрес на arr[0].

```
arr == &arr[0]
```



Какъв е типът на arr?

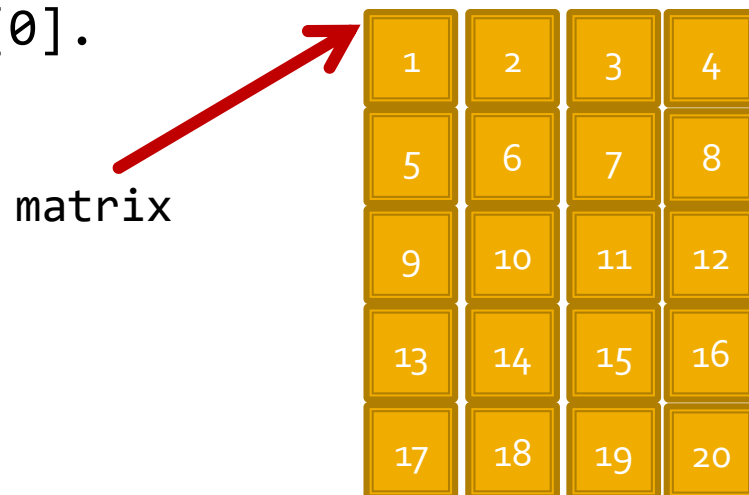
```
int *
```

# Указатели и масиви

```
int matrix[5][4] = {  
    { 1,  2,  3,  4},  
    { 5,  6,  7,  8},  
    { 9, 10, 11, 12},  
    {13, 14, 15, 16},  
    {17, 18, 19, 20}  
};
```

`matrix` – указател към първи елемент на масива, който също е масив.

`matrix` – съдържа адреса на `matrix[0]`.



# Указатели и масиви

```
int matrix[5][4] = {  
    { 1,  2,  3,  4},  
    { 5,  6,  7,  8},  
    { 9, 10, 11, 12},  
    {13, 14, 15, 16},  
    {17, 18, 19, 20}  
};
```

`matrix[i]` – масив от цели числа.

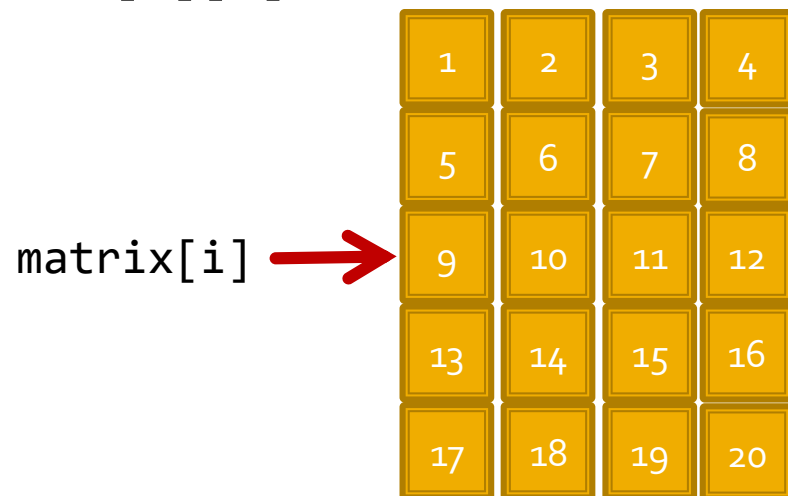
`matrix[i]` – съдържа адреса на `matrix[i][0]`.

Какъв е типът на `matrix`?

`<първи_елемент_на_масива>*`

`<масив_от_цели_числа>*`

`int**`



Задача:

Да се изведат стойностите на двумерен масив.

Да се използват указателна аритметика.

```
int matrix[5][4] = {  
    { 1,  2,  3,  4},  
    { 5,  6,  7,  8},  
    { 9, 10, 11, 12},  
    {13, 14, 15, 16},  
    {23, 24, 25, 26}  
};
```

```
for(int i=0; i < 5; i++) {  
    for(int j=0; j < 4; j++)  
        cout << (*(matrix + i)+j) << " ";  
    cout << endl;  
}
```

# Указатели и масиви

```
int arr[40] = {1, 2, 3, 4};
int matrix[5][4] = {
    { 1,  2,  3,  4},
    { 5,  6,  7,  8},
    { 9, 10, 11, 12},
    {13, 14, 15, 16},
    {23, 24, 25, 26}
};
```

Да се изведе на стандартния изход:

1. Адресът на първия елемент на `arr`;
2. Адресът на последния елемент на `arr`;
3. Петият елемент на `arr`;
4. Адресът на `matrix`;
5. Адресът на третия ред на матрицата `matrix`;
6. Адресът на елемента на четвърти ред и втори стълб на `matrix`;
7. Последният елемент на `matrix`;

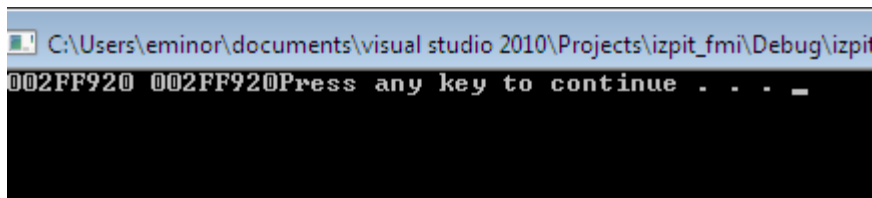
```
cout << arr;
cout << arr + 39;
cout << *(arr + 4) << arr[4];
cout << matrix;
cout << matrix + 2;
cout << *(matrix+3) + 1;
cout << (*(matrix+4) + 3)
<< matrix[4][3];
```

# Указатели и масиви

```
int arr[40] = {1, 2, 3, 4};  
int matrix[5][4] = {  
    { 1,  2,  3,  4},  
    { 5,  6,  7,  8},  
    { 9, 10, 11, 12},  
    {13, 14, 15, 16},  
    {23, 24, 25, 26}  
};
```

1. Адресът на третия ред на матрицата `matrix`;

```
cout << matrix + 2;           cout << *(matrix + 2) + 0;
```





# Указатели и масиви

```
int arr[40] = {1, 2, 3, 4};  
int matrix[5][4] = {  
    { 1,  2,  3,  4},  
    { 5,  6,  7,  8},  
    { 9, 10, 11, 12},  
    {13, 14, 15, 16},  
    {23, 24, 25, 26}  
};
```

// Можем ли да отместим само с броя на елементите

1. Адресът на третия ред на матрицата matrix;

```
cout << matrix + 8;          ERROR
```

```
matrix + 8*sizeof(int*) (not int)
```

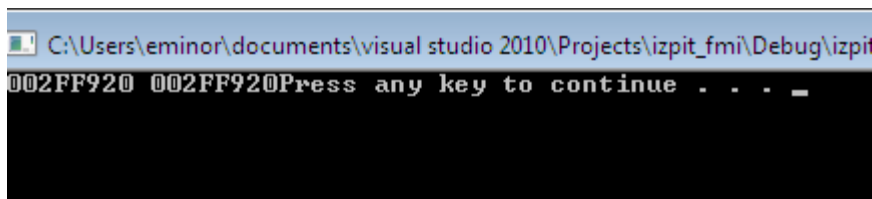
# Указатели и масиви

```
int arr[40] = {1, 2, 3, 4};  
int matrix[5][4] = {  
    { 1,  2,  3,  4},  
    { 5,  6,  7,  8},  
    { 9, 10, 11, 12},  
    {13, 14, 15, 16},  
    {23, 24, 25, 26}  
};
```

1. Адресът на третия ред на матрицата `matrix`;

```
cout << *matrix + 8;
```

```
cout << matrix + 2;
```



---

Следва продължение . . .