



УВОД В ПРОГРАМИРАНЕТО – УПРАЖНЕНИЕ №8

22.11.2022

СОФТУЕРНО ИНЖЕНЕРСТВО, ГРУПА 6

АСИСТЕНТ: ЕЛЕНА ТУПАРОВА



ЗА КАКВО ЩЕ СИ ГОВОРИМ ДНЕС?

- От миналата седмица – двоично представяне на отрицателни числа
- Указатели и псевдоними
- Указателна аритметика
- Указатели и масиви

ДВОИЧНО ПРЕДСТАВЯНЕ НА ОТРИЦАТЕЛНИ ЧИСЛА

- Три основни стратегии:
 - Само фиксиран бит (най-старшият) за знака – *прав код*;
 - One's complement – *обратен код*;
 - Two's complement – *допълнителен код*.

УКАЗАТЕЛИ И ПСЕВДОНИМИ (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 псевдоним)
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 😊

НАЧИНИ ЗА ПРЕДАВАНЕ НА ПАРАМЕТРИ НА ФУНКЦИЯ

- По стойност
- Чрез указател
- Чрез псевдоним

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);
```