

## Функции

КН, УП Практикум 2ра група 2023/2024, Богомил Стоянов, Виолета Кастрева

### 1. Какво е функция?

Функция в C++ е като рецепта. Това е именуван набор от инструкции, който изпълнява конкретна задача.

### 2. Как да създадете функция:

За да създадете функция, трябва да я декларирате и дефинирате.

Декларация: Казва на компилатора за името на функцията, какво връща и от какво се нуждае като параметри (ако има).

```
int add(int a, int b); // Това декларира функция с име 'add'.
```

Дефиниция: Предоставя действителния код за функцията.

```
int add(int a, int b) {  
    return a + b; // Това дефинира как работи 'add'.  
}
```

### 3. Части от функция:

Тип на връщане: какво функцията връща. В нашия пример `int` означава, че функцията ще върне цяло число. Когато функцията не връща нищо - `void`.

Име на функция: Името, което давате на функцията (в нашия случай "add").

Параметри: Това са като съставки за рецептата. В "add" `a` и `b` са параметри. Те са стойностите, които предавате на функцията.

Тяло на функцията: Във фигурни скоби поставяте кода, който казва на функцията какво да прави.

Връщане: Казва на функцията какво да върне. В 'add' `return a + b;` означава, че връща сумата от `a` и `b`.

### 4. Използване на функция:

За да използвате функция, вие я „извиквате“.

Пример:

```
int result = add (3, 4); // Извикване на 'add' с 3 и 4.
```

Резултатът ще бъде 7, защото `add(3, 4)` изчислява сумата и я връща.

### 5. Function overloading:

Можете да имате множество функции с едно и също име, но с различни параметри. Правилната функция се избира въз основа на какви параметри предоставяте.

### 6. Защо да ползваме функции?

Функциите в програмирането са от решаващо значение за:

- Модулност: Разделяне на кода на управляеми секции.
- Повторна употреба: Използване на един и същ код на различни места.
- Абстракция: Скриване на сложни детайли.
- Четливост: Прави кода по-разбираем.
- Тестване: Опростяване на отстраняването на грешки.
- Поддръжка: По-лесни актуализации и поправки.

- Споделяне: Повторно използване на код в различни проекти.

## 7. Още примери:

```
#include <iostream>
```

```
void greet() {  
    std::cout << "Hello, world!" << std::endl;  
}
```

```
int main() {  
    greet(); // Извикване на функцията  
    return 0;  
}
```

-----

```
#include <iostream>
```

```
int square(int num) {  
    return num * num;  
}
```

```
int cube(int num) {  
    return num * square(num);  
}
```

```
int main() {  
    int number = 4;  
    int result = cube(number);  
    std::cout << "Cube of " << number << " is " << result <<  
std::endl;  
    return 0;  
}
```