

Типове II: Шаблони

Калин Георгиев

14 март 2018 г.

Шаблони (на функции)

Еднообразни функции за различни типове

```
int findIndexMax
(int arr[], int arrSize)
{
    int indexMax = 0;
    for (int i = 1; i < arrSize; i++)
        if (arr[indexMax] < arr[i])
            indexMax = i;

    return indexMax;
}

int findIndexMax
(char arr[], int arrSize)
{
    int indexMax = 0;
    for (int i = 1; i < arrSize; i++)
        if (arr[indexMax] < arr[i])
            indexMax = i;

    return indexMax;
}
```

```
int findIndexMax
(double arr[], int arrSize)
{
    int indexMax = 0;
    for (int i = 1; i < arrSize; i++)
        if (arr[indexMax] < arr[i])
            indexMax = i;

    return indexMax;
}

...
```

Създаване на “Шаблон на функция”

```
template <typename T>
int findIndexMax (T arr[], int arrSize)
{
    int indexMax = 0;
    for (int i = 1; i < arrSize; i++)
        if (arr[indexMax] < arr[i])
            indexMax = i;

    return indexMax;
}
```

Използване на шаблона на функция

```

int main ()
{
    int arri[] = {1,5,6,7};
    cout << findIndexMax<int> (arri,4);

    double arrd[] = {2.1,17.5,6.0};
    cout << findIndexMax<double> (arrd,3);

    char arrc[] = "Hello";
    cout << findIndexMax<char> (arrc,5);

    char* arrstr[] = {"Hello", "World", "!"};
    cout << findIndexMax<char*> (arrstr,3); //!!!
}

```

```

template <typename T>
int findIndexMax
    (T arr[], int arrSize)
{
    int indexMax = 0;
    for (int i = 1; i < arrSize; i++)
        if (arr[indexMax] < arr[i])
            indexMax = i;

    return indexMax;
}

```

- Конкретният тип трябва да е съвместим с всички операции с него, които се прилагат в шаблона (в горния пример - <)!

Още един пример: печатане на “всякакви” масиви

```
template <typename T>
void printArray (T arr[], int arrSize)
{
    cout << "{"
    for (int i = 0; i < arrSize-1; i++)
        cout << arr[i] << ",";

    if (arrSize > 0) //no comma
        cout << arr[arrSize-1];

    cout << "}";
}
```

Шаблони на структури и класове

Създаване на “Шаблон на структура”

```
template <typename A, typename B>  
struct Pair  
{  
    A first;  
    B second;  
};
```

```
//...
```

```
Pair<int, char> p;
```

```
p.first = 0;  
p.second = 'a';
```


Създаване на “Шаблон на клас”

```
template <typename A, typename B>
class Pair
{
public:
    A first;
    B second;

    void print ()
    {
        std::cout << "first="
                    << first
                    << ", second="
                    << second
                    << std::endl;
    }
};
//...
Pair<int, char> p;
p.first = 0;
p.second = 'a';

p.print();
```

Благодаря за вниманието!