

1

ШАБЛОНИ НА ТИПОВЕ

1. *Обща характеристика*
2. *Шаблони на функции*
3. *Шаблони на класове*

ШАБЛОНИ НА ТИПОВЕ

3.5.2020 г.

2

1. Обща характеристика

Макро-механизъм, който позволява на програмистите да дефинират обобщени функции или класове, на базата на хипотетичен аргумент.

- Кодът на обобщените функции или класове се конкретизира когато техните дефиниции се използват с реални обекти.
- Обобщените дефиниции на функциите или на класовете се наричат **шаблони (templates)**.
- Шаблоните са техника на C++ за генериране на код за типове и функции.

ШАБЛОНИ НА ТИПОВЕ

3.5.2020 г.

3

1. Обща характеристика

Общата декларация на параметризиран тип започва със следната конструкция:

```
template <class име_тип>
декларация_на_програмна_единица
```

След като бъдат дефинирани, шаблоните се използват, като се определя име, следвано от аргумент:

```
име_на_шаблон <аргументи>
```

ШАБЛОНИ НА ТИПОВЕ

3.5.2020 г.

4

2. Шаблони на функции

Дефиниране шаблон на функция

```
template <class тип>
дефиниция_на_функция
```

ШАБЛОНИ НА ТИПОВЕ

3.5.2020 г.

5

2. Шаблони на функции

```
template <class T>
T max( T a, T b)
{ return a>b ? a : b; }
```

```
int n, m;
float x,y;
int i = max(n,m);
float z = max(x,y);
```

ШАБЛОНИ НА ТИПОВЕ

3.5.2020 г.

6

2. Шаблони на функции

Декларацията на параметризирана функция ще изглежда по следния начин:

```
template <class T>
T max( T, T );
```

ШАБЛОНИ НА ТИПОВЕ

3.5.2020 г.

2. Шаблиони на функции

7

Възможно е списъкът на параметризираните типове да съдържа повече от един елемент:

```
template <class T1, class T2>
T1 fun(T1 a, T2 b)
{
    int k;
    T2 x;
    .....
}
```

UNIVERSITY OF TULSA

3.5.2020

2. Шаблиони на функции

8

```
template <class T>
void mySwap( T &a, T &b)
{
    T tmp = a;
    a = b;
    b = tmp;
}
```

UNIVERSITY OF TULSA

3.5.2020

2. Шаблиони на функции

9

```
class Person
{
public:
    Person( void );
    Person( char *, unsigned int );
    Person( Person & );
    ~Person( void );
    void operator=(Person &);
    char *getname( void );
    void display( void );
private:
    char *name;
    int age;
}; //_end.Person
```

UNIVERSITY OF TULSA

3. Шаблиони на класове

Параметризиран клас се декларира по следния начин:

```
template <class arg1, class arg2, ...>
class име_на_клас
{
    декларации
}
```

UNIVERSITY OF TULSA

3.5.2020

3. Шаблиони на класове

11

При деклариране на обекти на параметризирания клас трябва да се посочат конкретните значения на параметризираните типове.

име_на_клас <конкретен_тип> име_на_обект;


UNIVERSITY OF TULSA

3.5.2020

3. Шаблиони на класове

12

```
template <class arg>
class Name
{
    declarations
};
```



UNIVERSITY OF TULSA

3.5.2020

13

3. Шаблини на класове

```
template <class T>
class Pair
{
    T a,b;
};
```

```
Pair <int> jeans;
Pair <double> gloves;
Pair <CStr> glasses;
```

3.5.2020 <

14

3. Шаблини на класове

```
template <class T>
class Pair
{
    T a, b;
};
```

Pair <int> jeans;

3.5.2020 <

3.5.2020 <

15

3. Шаблини на класове

```
template <class T>
class Pair
{
public:
    Pair(T x1, T y1) : x(x1), y(y1)
    {
    }
    void showQ();
private:
    T x,y;
};
```

3.5.2020 <

3.5.2020 <