





```
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);
```

```
2. Шаблони на функции

Декларацията на параметризирана функция ще изглежда по следния начии:

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

```
2. Шаблони на функции

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;
  .......
}
```

```
2. Шаблони на функции

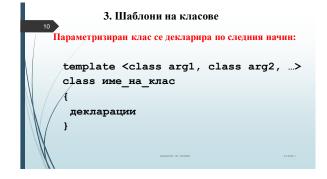
template <class T>
void mySwap( T &a, T &b)

{
    T tmp = a;
    a = b;
    b = tmp;
}
```

```
2. Шаблони на функции

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
```



```
3. Шаблони на класове

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

**MME_Ha_KJAC <**KOHKPETEH_TUIT> име_Ha_Oбект;
```



```
3. Шаблони на класове
template <class T>
class Pair
{
    T a,b;
};

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



```
3. Шаблони на класове

template <class T>

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