

STL

Введение

Шаблоны C++. Базовые понятия

- ❑ Шаблоны классов
- ❑ Шаблоны функций

Шаблоны классов

```
class pair_int_char
{
private:
    Int first;
    char second;
public:
    pair_int_char(int x, char y)
        : first(x)
        , second(y)
    {}
    void Print()
    {
        std::cout << first << std::endl;
        std::cout << second << std::endl;
    }
};
```

```
class pair_bool_double
{
private:
    bool first;
    double second;
public:
    pair_bool_double(bool x, double y)
        : first(x)
        , second(y)
    {}
    void Print()
    {
        std::cout << first << std::endl;
        std::cout << second << std::endl;
    }
};
```

Шаблоны классов

```
template<typename T1, typename T2>
class pair
{
private:
    T1 first;
    T2 second;

public:
    pair(T1 x, T2 y)
        :first(x)
        ,second(y)
    {}

    void Print()
    {
        std::cout << first << std::endl;
        std::cout << second << std::endl;
    }
};
```

Шаблоны функций

```
template<typename T>
T max(T x, T y)
{
    if(x<y)
        return y;
    else
        return x;
}
```

Оценка сложности алгоритма

Функцией трудоемкости называется отношение, связывающие входные данные алгоритма с количеством элементарных операций. ($O(n)$, $O(n*n)$, $O(n*\log(n))$)

STL

- ❑ Библиотека обобщенных компонент
- ❑ Гарантии производительности

Основные компоненты STL

- ❑ Контейнеры
- ❑ Обобщенные алгоритмы
- ❑ Итераторы
- ❑ Функциональные объекты
- ❑ Адаптеры
- ❑ Аллокаторы

Обзор STL. Контейнеры

- Контейнеры последовательностей:
 - ▣ `vector<T>`
 - ▣ `deque<T>`
 - ▣ `list<T>`

- Ассоциативные контейнеры:
 - ▣ `set<Key>`
 - ▣ `multiset<Key>`
 - ▣ `map<Key, T>`
 - ▣ `multimap<Key, T>`

Обзор STL. Обобщенные алгоритмы

- Find
- Max
- Merge
- Replace
- Sort
-

Обзор STL. Итераторы



- Однонаправленные
- Двунаправленные
- Произвольного доступа