

# РАЗРАБОТКА НА C++

## Урок 5. Контейнер map

# План

1. Контейнер map. Его особенности, методы, подвиды.
2. Контейнеры set и multiset
3. Контейнер list

# Что такое словарь?

Это контейнер, в котором все элементы **пары “ключ - значение”**

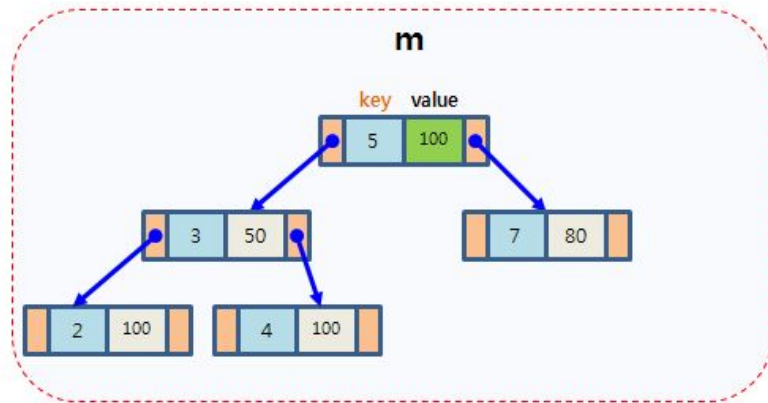
Ключи не могут повторяться как номера в телефонной книге.

Значения могут повторяться, как имена в телефонной книге.

В словарях все значения сортируются автоматически по ключу, так как `map` разработан на основе бинарного дерева.

```
#include <map>
```

```
map<int, string> m1;
```



# Работа со словарём

Для создания словаря нужно указать два типа данных в угловых скобках: один для ключа, второй - для значения:

```
map<int, string> m; // пустой словарь
map<char, int> m1 { // словарь из трёх пар ключ-значение
    {'a', 1}, {'b', 1}, {'c', 2}
};
```

# Работа со словарём

Вставка элементов:

```
m[1] = "a"; // добавление нового ключа и значения для него  
m.insert(make_pair(2, "b")); // добавление новой пары ключ-значение
```

Доступ к элементам осуществляется по ключу:

```
m[2] = "c"; // изменение значения по ключу  
cout << m[2]; // вывод значения по ключу
```

Удаление элементов:

```
m.erase[1]; // удаляем пару с ключом 1
```

# Работа со словарём

Размер словаря можно определить с помощью метода `size()`:

```
cout << m.size();
```

Метод `empty()` возвращает 1, если словарь пуст:

```
cout << m.empty();
```

Перебор элементов осуществляется в цикле:

```
for (auto elem : m)
    cout << "Key " << elem.first << ", value " << elem.second << endl;
```

Метод `count(n)` нужен, чтобы узнать, есть ли пара с ключом `n` в словаре:

```
cout << m.count(1);
```

# Разновидности словарей

В обычных словарях данные хранятся в отсортированном по ключу виде, а сам ключ должен быть уникальным.

Существуют также контейнеры **unordered\_map** (неупорядоченный словарь) и **multimap** (такой словарь, в котором отсутствует условие уникальности ключа, т.е. если вы произведете поиск по ключу, то получите не единственное значение, а набор элементов с одинаковым значением ключа).

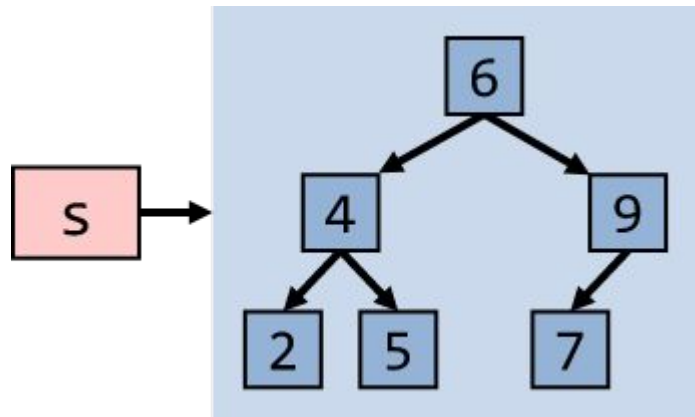
# Что такое множество?

**set** - это контейнер, в котором все элементы **уникальны** и отсортированы. Если элемент уже есть в множестве, он не будет добавлен.

**multiset** - это множество без условия уникальности ключа.

```
#include <set>
```

```
set<int> s1;
```





# Что такое список?

Это контейнер, в котором все элементы имеют **указатель на элемент спереди и сзади**.

У списков нет индексов, но можно в любой момент найти нужный элемент с помощью итераторов.

Можно вставлять и удалять **из любого места** в списке с максимальной скоростью среди всех контейнеров

```
#include <list>
```

```
list<int> l1;
```

