

Асоциативен масив

Библиотека: map

Синтаксис: map <key_type, value_type> name;

Асоциативният масив (речник) е шаблонна¹ структура данни, елементите на която се състоят от наредена двойка² (ключ, стойност). Ключът и стойността могат да бъдат от произволни тип данни, независими един от друг.

Ключовете са уникални стойности, играещи ролята на индекси в масива. Един ключ може да бъде съпоставен на една единствена стойност и ключовете са подредени³. Също така, тук предефинираният оператор [] освен достъп до стойността на даден ключ извършва и добавяне на нов ключ в асоциативният масив със стойност равна на стойността по подразбиране на value_type.

Асоциативният масив, както повечето структури от STL, поддържа работа с итератори. Тук стойността на итератора също е наредена двойка (ключ, стойност).

Пример:

```
map<string, double> m;  
m["Pi"] = 3.14;  
for(map<string, double>::iterator it = m.begin(); it != m.end(); ++it)  
    cout<<"("<it->first<<", "<it->second<<")"<<endl;
```

Често използвани методи и оператори:

[key]	Оператор за достъп до стойността, отговаряща на ключ key.
[key] = value	Добавя наредената двойка (key, value) в масива. Ако ключ key съществува, заменя стойността му с value.
.find(key)	Проверява дали ключа key фигурира в масива. Връща итератор към съответния елемент, ако той съществува, или към края на масива, в противен случай.
.count(key)	Брои срещанията на ключа key в масива. Поради уникалността на ключовете, този брой винаги е 0.
.clear()	Изчиства всички елементи.
.erase(key)	Изтрива елемента с ключ key, ако той съществува.
.erase(it)	Изтрива елемента, сочен от итератор it.
.erase(it_begin, it_end)	Изтрива всички елементи, намиращи се между итераторите it_begin и it_end.

¹ От произволен тип (int, double, vector, ...)

² Тип std::pair<key_type, value_type>

³ т.е. за типа key_type трябва да е предефиниран оператор < и оператор ==

<code>.empty()</code>	Проверява дали масивът е празен.
<code>.size()</code>	Връща размера на масива.
<code>.begin()</code>	Връща итератор към началото на масива.
<code>.end()</code>	Връща итератор към края на масива.