

Итератор

Синтаксис: `data_structure <type>:: iterator name;`

Итераторът е помощна структура данни, използвана за обхождане на линейни структури от STL (vector, map, list). Дефинира се като вида структура, нейния тип и ключовата дума iterator и се използва директно към обекти, съответстващи на дефиницията.

Пример:

```
vector<int> v;  
vector<int>:: iterator it = v.begin();  
while(it!=v.end())it++;
```

Основни функции, методи и оператори:

<code>*iterator</code>	Връща стойността на полето, сочено от итератора.
<code>++</code> <code>--</code>	Оператори за преместване с една позиция напред/назад.
<code>==</code> <code>!=</code>	Оператори за сравнение
<code>=</code>	Оператор за присвояване
<code>advance(iterator, x)</code>	Измества итератора с x позиции напред.
<code>next(iterator, x)</code>	Връща нов итератор, сочещ с x позиции напред.
<code>prev(iterator, x)</code>	Връща нов итератор, сочещ с x позиции назад.

В зависимост от структурата, за която са пригодени, итераторите могат да бъдат няколко вида:

Random-access – предоставят директен достъп до дадена позиция	<code>stl::vector</code>
Bidirectional – могат да се предвижват и в двете посоки, имат дефинирани <code>next()</code> и <code>prev()</code>	<code>stl::list</code> , <code>stl::map</code> , <code>stl::set</code>
Forward – могат да се предвижват само в една посока, имат дефиниран <code>next()</code> , но не и <code>prev()</code>	<code>stl::forward_list</code>