

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 <code>x</code> позиции напред.
<code>next(iterator, x)</code>	Връща нов итератор, сочещ с <code>x</code> позиции напред.
<code>prev(iterator, x)</code>	Връща нов итератор, сочещ с <code>x</code> позиции назад.
<code>distance(iterator1, iterator2)</code>	Връща цяло число – броя позиции между <code>iterator1</code> и <code>iterator2</code>

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

Вид итератор	Структури, за които се използва	Поддържа следните операции:	
<u>Random-access</u> предоставят директен достъп до дадена позиция	std::vector	It + n it += n it++ ++it it - n it -=n it-- --it	It == it1 it != it1 it < it1 it <= it1 it > it1 it >= it1

<u>Bidirectional</u> могат да се предвижват и в двете посоки, имат предефиниран next() и prev()	stl::list stl::map stl::set	lt++ ++it it-- --it	lt == it1 it != it1
<u>Forward</u> могат да се предвижват само в една посока, имат предефиниран next()	stl::forward_list	lt++ ++it	lt == it1 it != it1

Допълнителна литература:

[Iterator categories](#)

[Iterators in C++ STL - GeeksforGeeks](#)