

Основные итераторы

Итераторы ввода (Input iterators)

Итераторы ввода возвращает итератор потока ввода:

```
istream_iterator
```

Алгоритм for_each()

```
template <class InputIterator, class Function>
```

```
Function for_each(InputIterator first, InputIterator last,  
                  Function f)
```

```
{  
    while (first != last) f(*first++);  
    return f;  
}
```

```
#include <iostream>
```

```
#include <algorithm>
```

```
using namespace std;
```

```
void print(int value)
```

```
{  
    cout << value << ' '  
}
```

```
int main()
```

```
{  
    int a[] = { 0, 1, 2, 3, 4 };  
  
    // Визуализация элементов массива  
    for_each(a, a + 5, print);  
    cout << endl;  
  
    return 0;  
}
```

Результат работы программы:

```
0 1 2 3 4
```

```
#include <iostream>
#include <algorithm>
#include <vector>

using namespace std;

void print(int value)
{
    cout << value << ' ';
}

int main()
{
    // Объявление стандартного контейнера - вектор элементов
    vector<int> v(5);

    // Визуализация элементов вектора
    for_each(v.begin(), v.end(), print);
    cout << endl;

    // Явная инициализация элементов вектора
    for (int i = 0; i < v.size(); ++i) v[i] = i;

    // Визуализация элементов вектора
    for_each(v.begin(), v.end(), print);
    cout << endl;

    return 0;
}
```

Результат работы программы:

```
0 0 0 0 0
0 1 2 3 4
```

```
#include <iostream>
#include <algorithm>
#include <vector>

using namespace std;

void print(int value)
{
    cout << value << ' ';
}

int main()
{
    // Объявление стандартного контейнера - пустой вектор элементов
    vector<int> v;

    // Инициализация вектора - добавление элементов в контейнер
    for (int i = 0; i < 5; ++i) v.push_back(i);

    // Визуализация элементов вектора
    for_each(v.begin(), v.end(), print);
    cout << endl;

    return 0;
}
```

Результат работы программы:

0 1 2 3 4