

#include <iostream>

using namespace std;

#include <string>

#include <vector>

#include <algorithm>

#include<functional>

void myPrint(int val)

{

cout << val << " ";

}

void test01()

{

vector<int> v;

v.push\_back(10);

v.push\_back(50);

v.push\_back(30);

v.push\_back(40);

v.push\_back(50);

sort(v.begin(), v.end());

for\_each(v.begin(), v.end(), myPrint);

cout << endl;

sort(v.begin(), v.end(), greater<int>());

for\_each(v.begin(), v.end(), myPrint);

cout << endl;

}

int main()

{

test01();

system("pause");

return 0;

}



#include <iostream>

using namespace std;

#include <string>

#include <vector>

#include <algorithm>

#include<ctime>

void myPrint(int val)

{

cout << val << " ";

}

void test01()

{

srand((unsigned int)time(NULL));

vector<int> v;

for (int i = 0; i < 10; i++)

{

v.push\_back(i);

}

random\_shuffle(v.begin(), v.end());

for\_each(v.begin(), v.end(), myPrint);

cout << endl;

}

int main()

{

test01();

system("pause");

return 0;

}





#include <iostream>

using namespace std;

#include <string>

#include <vector>

#include <algorithm>

void myPrint(int val)

{

cout << val << " ";

}

void test01()

{

vector<int> v1;

vector<int> v2;

for (int i = 0; i < 10; i++)

{

v1.push\_back(i);

v2.push\_back(i+2);

}

vector<int> vTarget;

vTarget.resize(v1.size() + v2.size());

merge(v1.begin(), v1.end(), v2.begin(), v2.end(), vTarget.begin());

for\_each(vTarget.begin(), vTarget.end(), myPrint);

cout << endl;

}

int main()

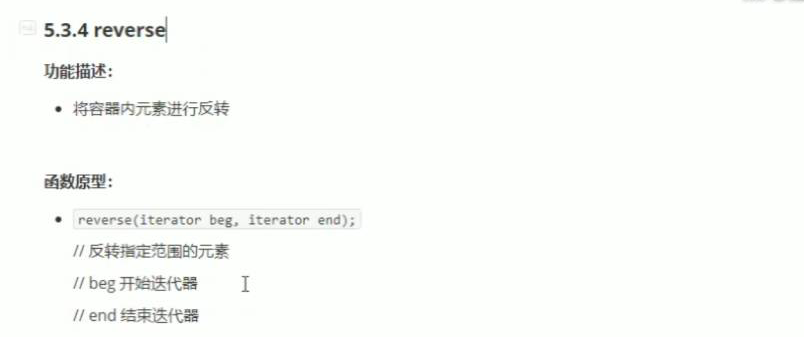
{

test01();

system("pause");

return 0;

}



#include <iostream>

using namespace std;

#include <string>

#include <vector>

#include <algorithm>

#include<functional>

void myPrint(int val)

{

cout << val << " ";

}

void test01()

{

vector<int> v;

v.push\_back(10);

v.push\_back(50);

v.push\_back(30);

v.push\_back(40);

v.push\_back(50);

cout << "反转前： " << endl;

for\_each(v.begin(), v.end(), myPrint);

cout << endl;

cout << "反转后 ： " << endl;

reverse(v.begin(), v.end());

for\_each(v.begin(), v.end(), myPrint);

cout << endl;

}

int main()

{

test01();

system("pause");

return 0;

}