//deque插入和删除

#include<iostream>

using namespace std;

#include<string>

#include<deque>

void print(const deque<int>& d) {

for (deque<int>::const\_iterator it = d.begin(); it != d.end(); it++) {

cout << \*it << " ";

}

cout << endl;

}

void test01() {

deque<int>d1;

//尾插

d1.push\_back(10);

d1.push\_back(20);

//头插

d1.push\_front(100);

d1.push\_front(200);

print(d1);

//尾删

d1.pop\_back();

print(d1);

//头删

d1.pop\_front();

print(d1);

}

void test02() {

deque<int>d1;

d1.push\_back(10);

d1.push\_back(20);

d1.push\_front(100);

d1.push\_front(200);

print(d1);

//insert插入

d1.insert(d1.begin(), 1000);

print(d1);

d1.insert(d1.begin(),2, 11);

print(d1);

//按照区间插入

deque<int>d2;

d2.push\_back(1);

d2.push\_back(2);

d2.push\_back(3);

d1.insert(d1.begin(), d2.begin(), d2.end());

print(d1);//1 2 3 11 11 1000 200 100 10 20

}

void test03() {

deque<int>d1;

d1.push\_back(10);

d1.push\_back(20);

d1.push\_front(100);

d1.push\_front(200);

//删除

deque<int>::iterator it = d1.begin();

it++;

//d1.erase(d1.begin());

d1.erase(it);

//按照区间的方式删除

d1.erase(d1.begin(), d1.end());

d1.clear();//这两行是一样的

}

int main() {

//test01();

//test02();

test03();

system("pause");

return 0;

}