

#include <iostream>

using namespace std;

#include <string>

#include <deque>

void printDeque(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;

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

{

d1.push\_back(i);

}

printDeque(d1);

deque<int>d2(d1.begin(), d1.end());

printDeque(d2);

deque<int>d3(10, 100);

printDeque(d3);

deque<int>d4(d3);

printDeque(d4);

}

int main()

{

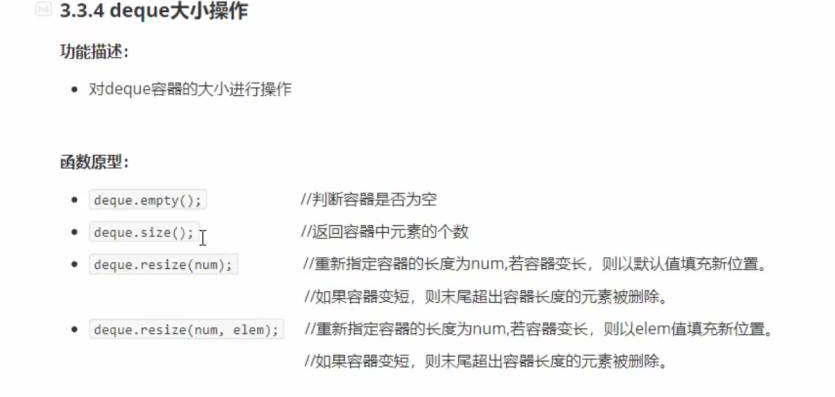
test01();

system("pause");

return 0;

}







#include <iostream>

using namespace std;

#include <string>

#include <deque>

void printDeque(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;

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

{

d1.push\_back(i);

}

printDeque(d1);

if (d1.empty())

{

cout << "d1为空。" << endl;

}

else

{

cout << "d1不为空。" << endl;

cout << "d1的大小为： " <<d1.size()<< endl;

}

d1.resize(15,1);

printDeque(d1);

}

int main()

{

test01();

system("pause");

return 0;

}



#include <iostream>

using namespace std;

#include <string>

#include <deque>

void printDeque(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(55);

d1.push\_front(40);

printDeque(d1);

//尾删除

d1.pop\_back();

printDeque(d1);

//头删除

d1.pop\_front();

printDeque(d1);

}

void test02()

{

deque<int> d1;

d1.push\_back(10);

d1.push\_back(20);

d1.push\_front(55);

d1.push\_front(40);

printDeque(d1);

d1.insert(d1.begin(),2, 200);//2个200

printDeque(d1);

}

int main()

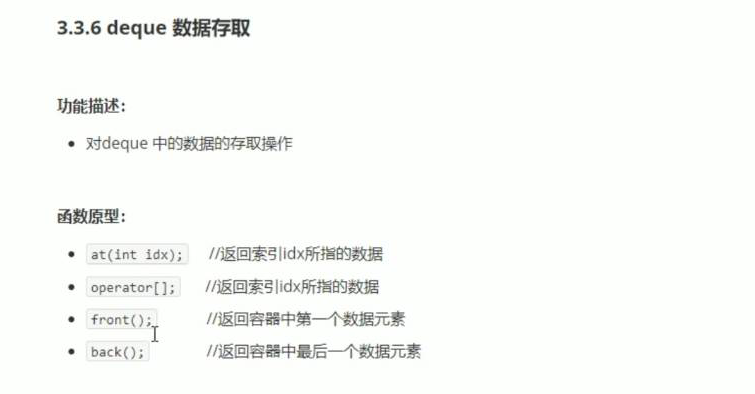
{

test02();

system("pause");

return 0;

}



#include <iostream>

using namespace std;

#include <string>

#include <deque>

void test01()

{

deque<int> d1;

//尾插

d1.push\_back(10);

d1.push\_back(20);

d1.push\_back(20);

d1.push\_back(20);

d1.push\_back(20);

//头插

d1.push\_front(55);

d1.push\_front(40);

d1.push\_front(40);

d1.push\_front(40);

d1.push\_front(40);

for (int i = 0; i < d1.size(); i++)

{

cout << d1[i] << " " ;

}

cout<<endl;

for (int i = 0; i < d1.size(); i++)

{

cout << d1.at(i) << " ";

}

cout << endl;

cout << "访问第一个元素： " << d1.front() << endl;

cout << "访问最后一个元素： " << d1.back() << endl;

}

int main()

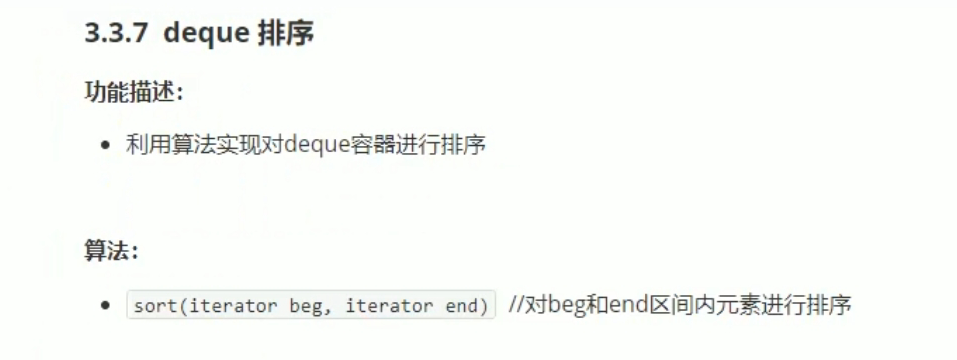
{

test01();

system("pause");

return 0;

}



#include <iostream>

using namespace std;

#include <string>

#include <deque>

#include<algorithm>

void printDeque(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;

d1.push\_back(10);

d1.push\_back(5);

d1.push\_back(55);

d1.push\_back(25);

d1.push\_front(500);

d1.push\_front(200);

d1.push\_front(100);

d1.push\_front(300);

printDeque(d1);

cout << "排序后 ： " << endl;

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

printDeque(d1);

}

int main()

{

test01();

system("pause");

return 0;

}