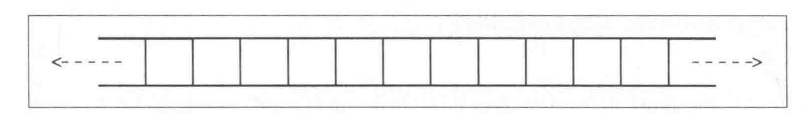
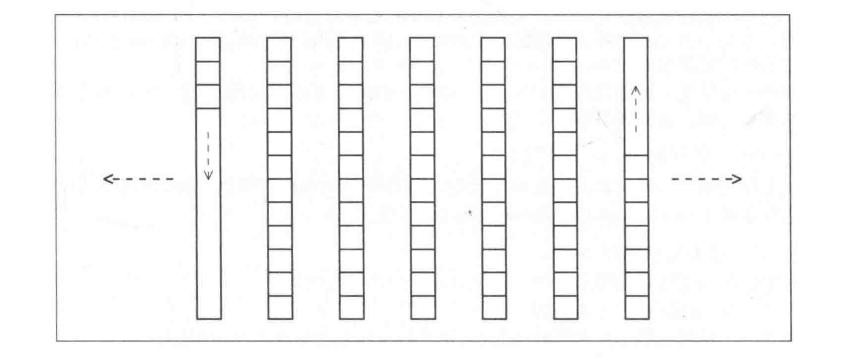
deque：采用dynamic array管理数据，提供随机访问。deque类似vector，但其头尾都能插入和删除元素。

deque逻辑结构



deque内部结构：



deque不提供容量操作（没有类似vector中的capactiy()和reserve()）

例：程序stl\_test45

// 解决vs上<<无法直接输出string类型的问题

namespace *std*

{

// <<运算符重载

*ostream*& operator <<(*ostream*& os, const *string*& src)

{

return os << src.*c\_str*() << *endl*;

}

}

int *main*(int argc, char\* argv[])

{

*deque*<*string*> coll;

coll.*assign*(3, "string");

coll.*push\_back*("last string");

coll.*push\_front*("first string");

*cout* << "original: " << *endl*;

*copy*(coll.*cbegin*(), coll.*cend*(), *ostream\_iterator*<*string*>(*cout*, "\n"));

*cout* << *endl*;

*cout* << "remove first and last: " << *endl*;

coll.*pop\_front*();

coll.*pop\_back*();

for (*deque*<*string*>::*size\_type* i = 1; i < coll.*size*(); ++i)

{

coll[i] = "another " + coll[i];

}

*cout* << "resize: " << *endl*;

// 在末端插入"resize string"或删除末端元素

// 使size() = 4

coll.*resize*(4, "resize string");

*copy*(coll.*cbegin*(), coll.*cend*(), *ostream\_iterator*<*string*>(*cout*, "\n"));

return 0;

}

输出为：

original:

first string

string

string

string

last string

remove first and last:

string

another string

another string

resize:

string

another string

another string

resize string