

Deque = Stack + Queue + push_front

- object: a container that holds some elements
- action: [constant-time] push_back (like push and enqueue), pop_back (like pop), pop_front (like dequeue), push_front
- application: job scheduling

Reading Assignment

be sure to go ask the TAs or me if you are still confused

Some Useful Implementations in C++

Standard Template Library (STL)

- container `vector`: dynamically growing dense array
- container `list`: doubly-linked list
- container `deque`: “chunked” linked-list implementation of deque
- container adapter `stack`: turning some container to a stack

```
1  template <typename T, typename Container = deque<T> >  
2  class stack;
```

- container adapter `queue`: turning some container to a queue

```
1  template <typename T, typename Container = deque<T> >  
2  class queue;
```

Some Useful Implementations in C++

```
1  #include <vector>
2  #include <stack>
3  #include <queue>
4  using namespace std;
5  vector<int> intarray;
6  stack<char, vector<char> > charstackonvector;
7  queue<double> doublequeue;
8  intarray.resize(20); intarray[3] = 5;
9  charstack.push_back(' ');
10 char c = charstack.pop_back();
11 doublequeue.push_back(3.14);
12 double d = doublequeue.pop_front();
```