1.3 Deque `std::deque`

Use for

- Similar purpose of `std::vector`
- Basically `std::vector` with efficient `push_front` and `pop_front`

Do not use for

- C-style contiguous storage (not guaranteed)

Notes

- Pronounced 'deck'
- Stands for Double Ended Queue

Example Code

```
std::deque<int> d;

//---------------------------------
// General Operations
//---------------------------------

// Insert head, index, tail
d.push_front(value);                    // head
d.insert(d.begin() + index, value);     // index
d.push_back(value);                     // tail

// Access head, index, tail
int head = d.front();       // head
int value = d.at(index);    // index
int tail = d.back();        // tail

// Size
unsigned int size = d.size();

// Iterate
for(std::vector<int>::iterator it = d.begin(); it != d.end(); it++) {
    std::cout << *it << std::endl;
}

// Remove head, index, tail
d.pop_front();                  // head
d.erase(d.begin() + index);     // index
d.pop_back();                   // tail

// Clear
d.clear();
```