Protip 1: deque

Deque is a data structure in the c++ library. It is a more powerful version of arrays and vectors, and it is like stack and queue combined. Its name is a shorthand of double-ended queue, so you can guess what it does.

How to declare:

#include <deque>

using namespace std;

deque<int> d; (when declared it is empty)

Some operations:

d.push\_back(x);

d.push\_front(x);

d.pop\_back(x);

d.pop\_front(x);

int s = d.size();

bool b = d.empty();

int ith\_element = d[i];

int front = d.front();

int back = d.back();

d.clear();

above all cost O(1)

d.insert(d.begin() + pos, x);

d.erase(d.begin() + pos);

above is O(n) (or O(n/2), so a little bit faster than vector)

If you do not know, erase in vector or deque is slow because the computer basically shifts all elements after the deleted element 1 position back.

you can even use the assignment operator (=)!

deque<int> d1, d2;

//do something with d1

d2 = d1;

I recommend using deque when you have situation where you need dynamic arrays or vectors because it is faster.