Notes chapter 16

Summary and array based implementation of the stack and queue

A linked list based implementation of the stack and queue

**Stacks**

Push adds one

Pop takes one out

Top read the top

TopAndPop took it off and have it

isEmpty check to see if empty

makeEmpty check to see if its empty

All basic operations so be constant time

First in last out

**Queues**

Enqueue get in line

getFront look at the first thing

Dequeue get out of line

isEmpty check if empty

makeEmpty make it empty

Called a line in America

All basic operations so be constant time

First in first out

With linked list

Stacks is about the same but the memory is worse

Queues is memory that is worse. Better then add because we don’t have to worry about wrap around.

Double ended queue

You can add to the front

You can add to the rear

You can remove front

And remove from the rear