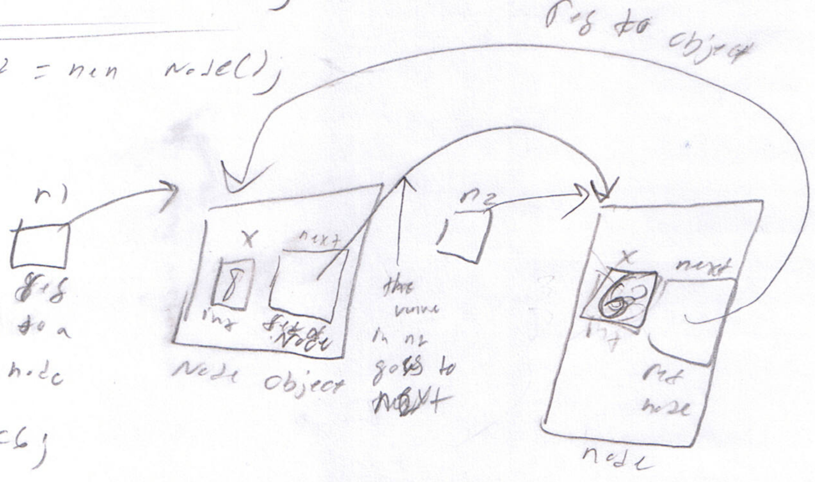
cmp 338

9.17.07

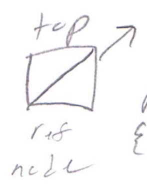
```
public class Node
{
    int x
    node next;
}
```

```
Node n1 = new node();
Node n2 = new node();
n1.x = 8;
n1.next = n2;
n1.next.x = 6;
n1.next.next = n1;
```



```
public class StackLLInd
{
    Node top = null;

    public boolean isEmpty()
    {
        return top == null;
    }
    public void push(int x)
    {
    Node temp new Node();
    temp.x = 4;
    temp.next = top;
    top = temp;
```



Node temp = new node(4, top); } Top = new node(6, top);

```
public class node
{
    int x
    node next
    public node (int x, node next)
    {
        this.x = x;
        this.next = next;
```

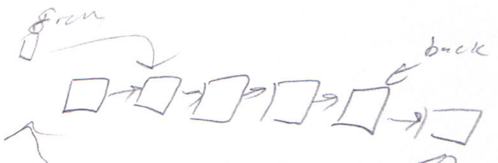
(private) (same thing)

Cmp 338

2

9.17.09

Queue

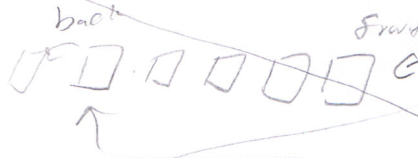


back.next = new Node()

back = back.next

temp = front.next;

return temp



cant get this off right away ~~w/o~~out going all the way through

back = new node(x, back)

Queue ADT
(abstract Data type)

enQueue - adds an Item to the Queue at the back

dequeue - Removes Item from the front of the Queues

front - reports the value of the Item at the front of the queue

back - reports the value of the Item at the back of the queue

isEmpty - returns whether the queue is empty or not

```
public class StackList
{
private Node top = null;

  public boolean isEmpty()
  {
     return top == null;
  }
 public void push(int x)
 {
   top = new Node(x, top);
 }
 public int pop()
 {
  int temp = top.x
  top = top.next
  return temp
 }
 public int top()
 {
   return top.x;
 }
```

Cmp 388

9.17.09

Queue

```
public class QueueLink
{
    private Node front = null;
    private Node back = null;
  public boolean isEmpty()
  {
     return front == null;
  }
  public int front()
  {
     return front.x;
  }
 public int back()
 {
    return back.x;
 }
 public void enqueue(int x)
    back.next = new Node(x, null);
    back = back.next;
public int dequeue()
{
  int  temp = front.x
  front = front.next;
  return temp;
}
```