CS 180 - Linked queues

Note Title 9/22/2010

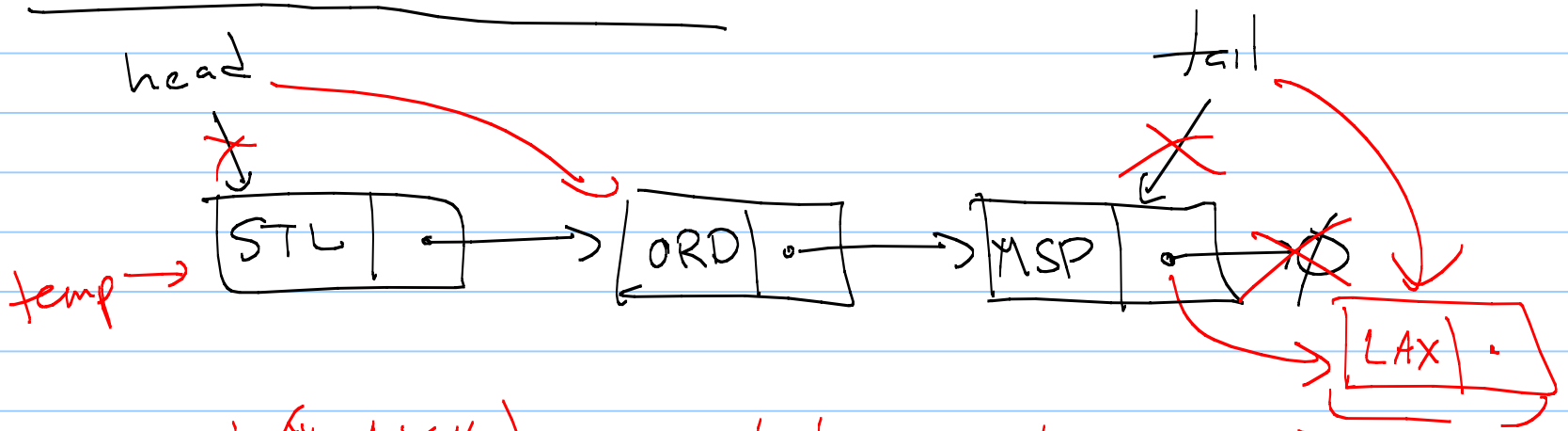
Announcements

- Test Friday!
  Review is tomorrow in class.

- Program 2 is up - due Oct. 2
  Checkpoint on next Tuesday

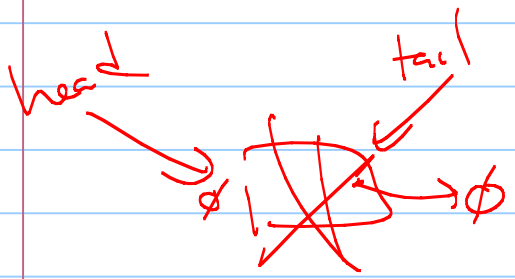

Functions
Push
Pop
Front
empty
size
return size
head
tail

Linked Queues – FIFO

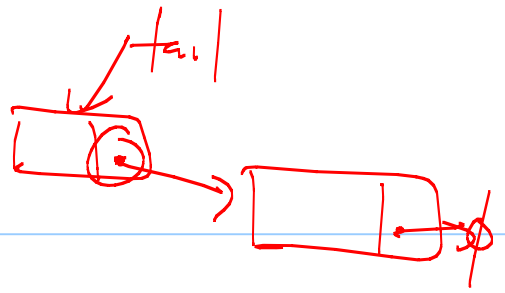


push("LAX") : tail->next =

pop() : temp = head
head = head->next;
delete temp




```
private:
  int size;
  Node* head;
  Node* tail;
```



Push

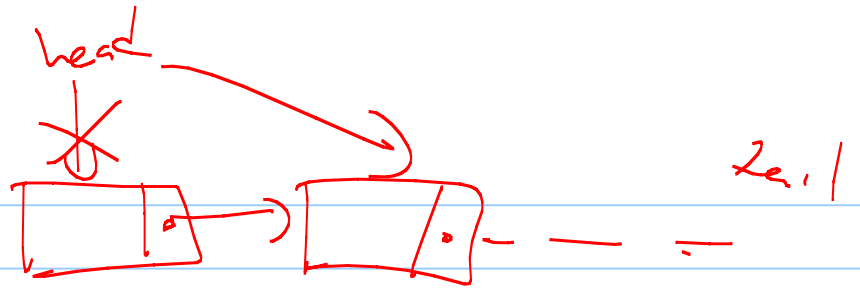
```
void push(const Object& elem) {
    Node* v = new Node(elem);
    if (empty()) {
        head = v;
        tail = v;
    }
    else {
        tail -> next = v;
        tail = tail -> next;
    }
    size++;
}
```

tail = v;

Front

```
const Object& front() const {
    if (empty())
        raise error;
    return head->element;
}
```

Pop




```
void pop() {
    if (empty())
        raise error;
    Node* temp = head;
    if (size == 1) {
        head = NULL;
        tail = NULL;
    }
    else
        head = head->next;
    delete temp;
    size--;
}
```

Not covering isempty & size.

Helper + Housekeeping Functions

What will copy constructor, operator=, & destructor need?

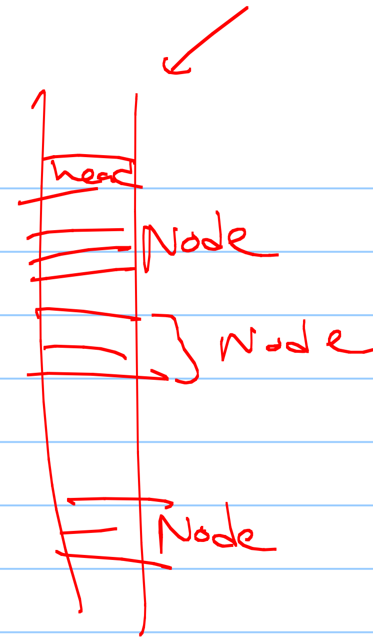
helper functions:

- copyFrom
- removeAll

} private, user will never see

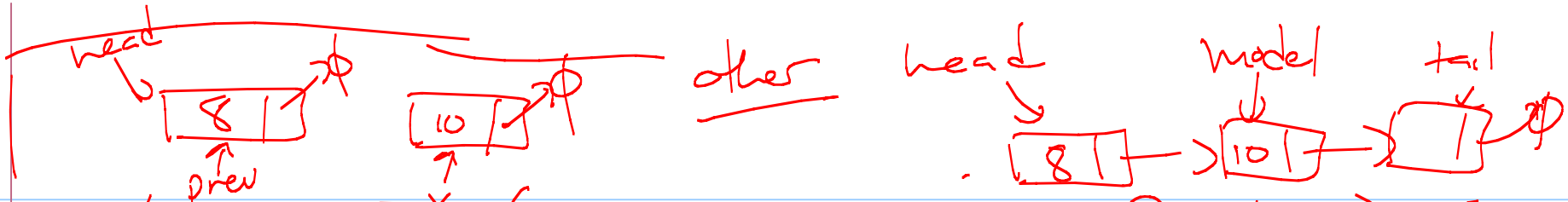
pop already deletes

```
void removeAll ( ) {
    while (!empty())
        pop();
}
```



array

```
delete [] Q;
```




```
void copyFrom(const LinkedQueue& other) {
    head = NULL;
    Node* model = other.head;
    Node* prev = NULL;
    while (model != NULL) {
        Node* v = new Node(model->element, NULL);
        if (head == NULL)
            head = v;
        else
            prev->next = v;
        model = model->next;
        prev = v;
    }
    size = other.size;    tail = prev; }
```