CS180 - Lists

Note Title 10/21/2011

Announcements

- Program 4 will be up (Vector.h isn't updated)
  due next Thursday
- No class Monday
- Reading assignment: read 6.2 (section on lists)

operator [ ] : O(1) ✓ ☺

Lists:

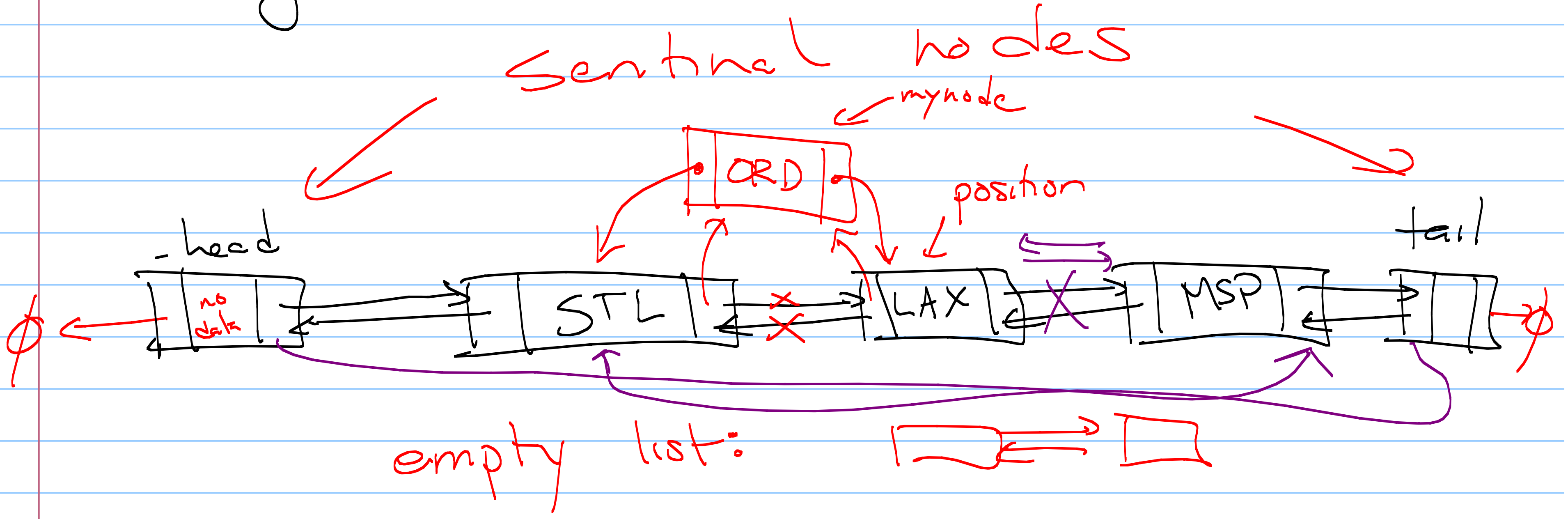
Motivation: insert in vectors is slow!

(Running time?) O(n)

Idea: If I know where the element should go, inserting should be easy.

Node* mynode = new Node · -

Doubly Linked Lists



insert (ORD) - where?

indicate a node & insert before it

O(1) time

Problem: Pointers!

What do we need in order to know where we should insert?

position (→ Node pointer)

Nodes are private!

Solution: "Wrap" Nodes inside a class.

↳ Iterators

Write functions which allow restricted pointer operations.

Iterators

An iterator will give the user a "pointer", but with a heavily controlled structure.
(So they can't touch nodes directly.)

⇒ no seg faults

Compromise: Functionality versus info. encapsulation

STL functions

Mix of iterators & list functions.

Usage:

```
list <int>   mylist;
list <int> :: iterator   it;
```

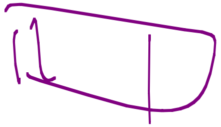
Code:

2 internal classes

```
private:
    Node struct
        data:
      Node*   _next
      Node*   _prev
      Object  _data
```

```
Iterator class (public)

    private:
        Node*   _current;
```



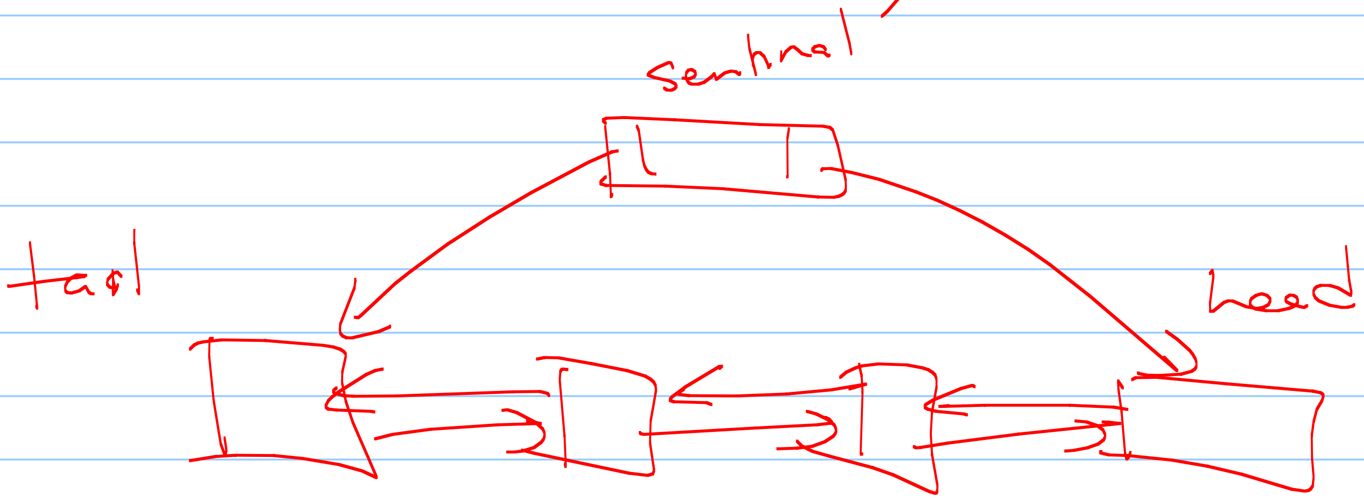
List data:

```
int _size;

Node _head;
Node _tail;
```

Other ways

circularly linked lists


sentinel
tail
head