1. I will announce TAs by tomorrow afternoon.
2. Assign #5 - print/ I don't care if you take length or remove/ supposed not. not remove nodes that have a specific value.
3. not doing we grade on our own.
In-class exercise...

• Get into groups of 4-5.
• Why is having a tail in a linked list advantageous?
• How are you sorting for assignment #5?
• Draw the pic for swapping nodes.
  – What is base case?
  – What are other cases?
Complexity – Big O

• Based on what?
  
  *Data + Algorithm*

• Why is this important?

  *Scalability*
Constant time – O(1)

```c
struct node * push(struct node *head, int n) {
    struct node *temp=malloc(sizeof(struct node));
    temp->val=n;
    temp->next=head;
    head=temp;
    return head;
}
```
Linear time – $O(n)$

```c
int length(struct node *head) {
    int n=0;
    while(head!=NULL) {
        n++;
        head=head->next;
    }
    return n;
}
```
Quadratic time – $O(n^2)$

```c
void bubble_sort(struct node *head, int size) {
    ...
    for(int iteration=1; iteration<size; iteration++) {
        for(int i=0; i<size-iteration; i++) {
            if(current->val > current->next->val) {
                // swap values
            }
            // move current to next node
        }
        current=head;
    }
}
```
Logarithm (base 2) time – $O(\log_2 n)$

```cpp
int binarySearch(const int list[], int length, int item) {
    int first = 0, last = length - 1, mid;
    bool found = false;
    while (first <= last && !found)
    {
        mid = (first + last) / 2;
        if (list[mid] == item)
            found = true;
        else if (list[mid] > item)
            last = mid - 1;
        else
            first = mid + 1;
    }
    if (found) { return mid; }
    else { return -1; }
} //end binarySearch
```

$n = 1024 = 2^{10}$
Final Project...

• Understanding Merge Sort...
• It will be posted today...
• Absolutely no late finals accepted!!!
Sorting Algorithms...

• [https://www.youtube.com/watch?v=kPRAOW1kECg](https://www.youtube.com/watch?v=kPRAOW1kECg)

• Search for Sorting Dancers