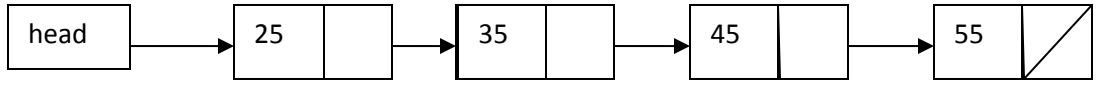
Pre-lab Worksheet

CS163 Review – Linear Linked Lists

Each group member should collaborate on this worksheet. All online students should participate!
The goal of these worksheets is to help prepare you for the next step in programming!
*It is **optional** if you receive an **E** (Excellent) or **P** (Proficient) on the midterm proficiency demos*

Accessing Data in a Linear Linked List *Assume the following list has already been created*



current

_____1. List one reason why we need more than a head pointer when working with a linear linked list:

Because we cannot traverse efficiently without using a temporary pointer because we could lose our data

_____2. Create the variable definition for current: node * current;

_____3. Set current to point to the first node (25): current = head;

_____4. Write the code for each of the following operations:

a. display the contents of the first node (25): cout << head->data

b. display the contents of the second node(35), using head: cout << head->next->data << endl;

c. traverse to the second node and display its contents (35) using current:

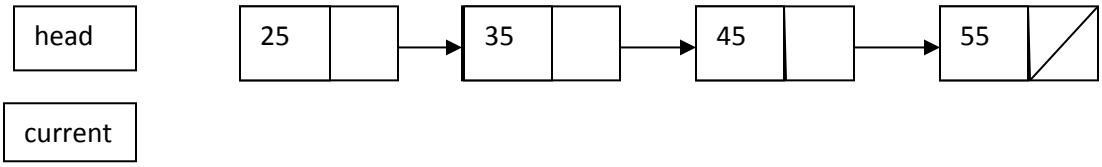
```
current = current->next;
cout << current->data << endl;
```

_____5. Assume that we have done some traversal through the linked list, how do we know if we are done traversing through all of the nodes?

```
if(!current)
  //we are done
```

_____6. Or, how do we know if we are AT the last node? if(!current->next)

Setting up a LLL:



_____1. Where should head pointer in the above list?

It should point to the node that contains 25

_____2. Why doesn't head->next; traverse to the next node on its own as a statement?

because it does not change anything, it just accesses the variable

_____3. Why don't I have to say head = new node; or current = new node; each time I need a pointer?

Because you do not have to create a node when you declare a pointer.

_____4. Write an if statement to find out if the list is empty: if(!head)

_____5. First, create a node structure that holds an integer and a next pointer

```
struct node
{
  int num;
  node * next;
};
```

_____6. In main(), create a pointer (temp) and a pointer (head). Set both to NULL.

```
int main()
{
  node * temp = NULL;
  node * head = NULL;

  return 0;
}
```