CZ1106 Problem Solving and Computation II

Tutorial 4: Data Structure

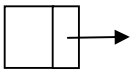
22 March 2007, 6:00pm
Venue: Temasek Hall

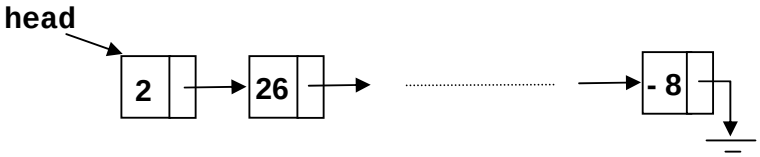
Students are to present their answers to questions 4 and 5.

1. Assume that **`head`** points to the first node of a linked list which contains integers with these declarations:

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

struct node *head;
```





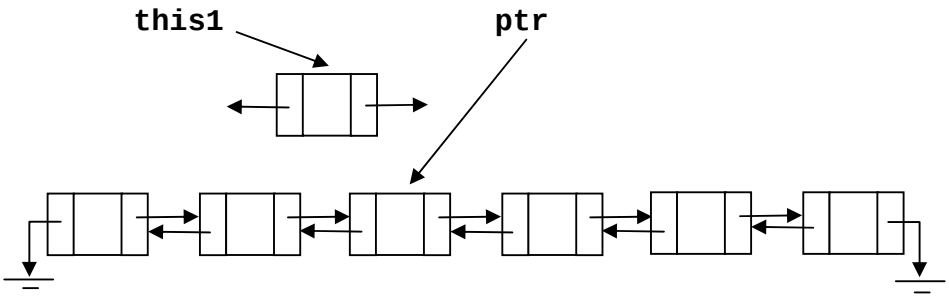
Write two functions which return respectively
(i) the maximum of the values in the linked list
(ii) the minimum of the values in the linked list

2. Write the code segment that will split two linked lists. The function should have two pointers as arguments: **ptr1** will point to the beginning of the list, and **ptr2** to the node at which it should be split, so that all nodes before the node pointed to by **ptr2** are in the first list and all nodes after it are in the second list.

3. Write a function named **`insert_left()`** to insert a node pointed to by **`this1`** to the left of a node pointed to by **`ptr`** in a doubly linked list:

```
struct node
{
  int i;
  struct node *left;
  struct node *right;
};

struct node *ptr, *this1;
```

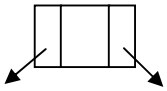


4. One of the nodes in a circular linked list is pointed to by **`ptr`**. Write the code segment that will count the number of nodes in the circular linked list.

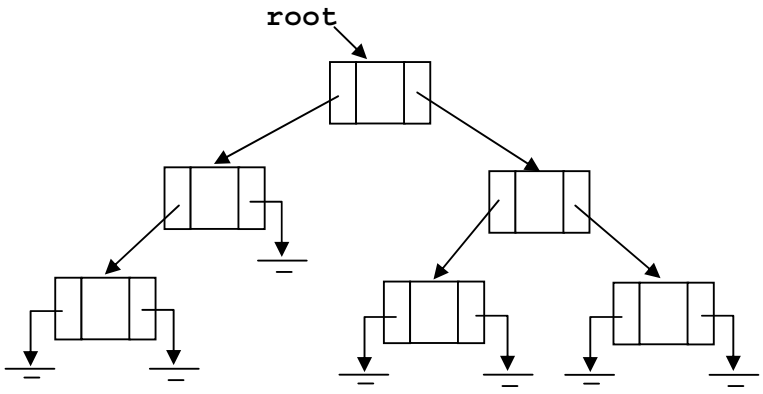
5. Given the following declarations used for a tree node:

```
struct node
{
  int i;
  struct node *left;
  struct node *right;
};

struct node *root;
```



A data structure called **Tree** is pointed to by the **`root`** pointer, and has been constructed as shown in the following diagram:



What is the objective of the following code segment?

```
int objective (node *root)
{
   if (root == NULL)
      return 0;

   return (1 + objective (root->left)  +
           objective (root->right));
}
```