Egypt-Japan University of Science and Technology.
Electronics, Communications and Computer Engineering (ECCE) School.
CSE Dept.
Dr. Ahmed Arafa.
Data Structure.

Exercise 6: Linked Lists

Q1: Write and call a function that add a node to the start of a linked list of students. The node contains id, name and degree of each student. Use the program you created in the last exercise and add an entry to the function in the switch case and call it for execution.

```
void addatStart(ListNode *& listHead,int id, string name,float degree)
{
    ListNode *newNode;
    newNode = new ListNode;
    newNode->id=id;
    newNode->name=name;
    newNode->degree=degree;
    newNode->next=NULL;
    if (!listHead)
        listHead = newNode;
    else
    {
        newNode->next = listHead;
        listHead = newNode;
    }
}
```

****

Q2: Write and call a function that deletes the first node in a linked list. Use the program you created in the last exercise and add an entry to the function in the switch case and call it for execution.

```
void deleteStartNode(ListNode * &listHead)
{
    ListNode *nodePtr;
    nodePtr = listHead;
    if (!listHead)
    {
        cout << "Empty List!" << endl;
        return;
    }
    else
    {
        listHead = NULL;
        delete nodePtr;
    }
}
```

****

Q3: Write and call a function that deletes a node at the end of a linked list using single pointers. Use the program you created in the last exercise and add an entry to the function in the switch case and call it for execution.

```
void deleteLastNode1(ListNode * &listHead)
{
ListNode *blastNode= listHead;
    if (!listHead)
    {
        cout << "Empty List!" << endl;
        return;
    }
```

```
    else if(listHead->next==NULL)
    {
        listHead = NULL;
        delete blastNode;
    }
    else
    {
        while (blastNode->next->next != NULL)
        {
            blastNode = blastNode->next;
        }
        delete blastNode->next;
        blastNode->next=NULL;
    }
}
```

****

Q4: Write and call a function that reverses the linked list. Use the program you created in the last exercise and add an entry to the function in the switch case and call it for execution.

```
void reverse_list(ListNode *& listHead)
{
    if(listHead == NULL)
        cout <<"\n List is Empty!" << endl;
    else
    {
        ListNode * current_node = listHead, * next_node= NULL, * previous_node= NULL;
        while ( current_node !=NULL)
        {
            next_node= current_node->next;
            current_node ->next = previous_node;
            previous_node = current_node;
            current_node = next_node;
        }
        listHead = previous_node;
    }
}
```

****

Q5: Write a function to search about a specific student in the list using his id. Use the program you created in the last exercise and add an entry to the function in the switch case and call it for execution.

```
bool searchID(ListNode*  head, int id)
 {
    ListNode* current = head;
    while (current != NULL)
    {
      if (current->id == id)
          return true;
       current = current->next;
    }
    return false;
 }
```

****

Q5: These Questions are assigned as an assignment and delivered to the lab instructor and should be finished in the lab. Each student should answer the two questions.

1-Write and call a function that deletes the whole linked list.

2-Write and call a function that deletes a node by its position in the list.