Bach Khoa University
Faculty of Computer Science and Engineering

Data structure & Algorithm
Fall 2017

Lab 3 – Linked List

- Linked list

Problem 1

Write the following utility functions:

a) Create a list of length N with random data: `node* CreateListRandom(int n)`
b) Insert a node at the end of the linked list: `void InsertTail(node*& head, int newData)`
c) Insert a node before the head of the linked list: `void InsertHead(node*& head, int newData)`
d) Delete a node having delData: `void DeleteData(node*& pHead, int delData)`

Problem 2

Given a linked list, write a function to:

a) Sum every element of a linked list.
b) Evaluate the polynomial defined by the linked list, with x inputted by user:

```
node* head1 = CreateListRandom(4); // 1 (head) -> 7 -> 14 -> 0 -> NULL
float x;
cin >> x;
float result = PolyEval(head1, x); // Return 1x^3 + 7x^2 + 14x + 0
```

Problem 3*

Write a recursive function to reverse a linked list:

```
PrintLinkedList(head); // 1 (head) -> 5 -> 7 -> 9 -> 2
node* newHead = ReverseList(head);
PrintLinkedList(newHead); // 2 (head) -> 9 -> 7 -> 5 -> 1
```

You **must** use the prototype: `node* ReverseList(node* root, int& n),` n is a variable containing the length of the linked list.

Problem 4

Propose a linked list called StudentNode that stores:

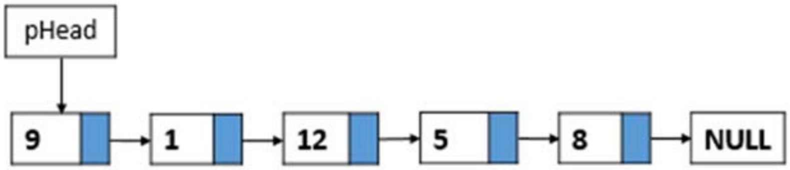
- His/her Math and English scores.
- His/her name.

- His/her student ID.

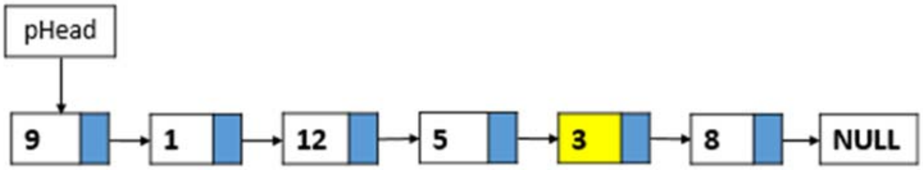
Create a linked list using the newly made StudentNode struct. The linked list created should have at least 3 nodes. Print the linked list.

Problem 5*

Write a function to insert a new node into the linked list after the first N nodes of the linked list. For example, given the linked list:



Calling `insertList(pHead, 4, 3)` would add a new node with data = 3 after the 4th node of the linked list:



Note:

- If N ≤ 0 then insert the new node as head of the linked list.
- If N is greater than the number of nodes in the linked list, insert the new node at the end of it.