**CSCI 520.001**

**Programming Assignment #6**

Turn in your work in the drop box for Assignment 6 (for Week 6) in eCollege 520.001 course by 11:59pm on Thursday, Oct. 16

1. Complete the following **linked list** **implementation** of STACK in C++ to include another member function **void peekAll().** Function void peekAll()displays the entire stack content without changing the stack cointent. **Do not call empty() and pop() functions. Function peekAll() displays an empty line if the stack is empty.**

**class STACK**

**{**

**private:**

**struct node**

**{ int item; node\* next;**

**node(int x, node\* t)**

**{ item = x; next = t; }**

**};**

**typedef node \*link;**

**link head;**

**public:**

**STACK(int)**

**{ head = 0; }**

**int empty() const**

**{ return head == 0; }**

**void push(int x)**

**{ head = new node(x, head); }**

**int pop()**

**{ int v = head->item; link t = head->next;**

**delete head; head = t; return v; }**

**void peekAll()**

**{**

**/\* implement this part \*/**

**}**

**};**

1. Write a C++ program that creates a stack of integer numbers (for up to 100 numbers), and then it does the following on user’s input (**your program must check for conditions: e.g. no pop when the stack is empty**):

P x : pushes integer x onto the stack

R : pops off the stack top and display

E : outputs “empty” if stack is empty; otherwise “not empty”

K : outputs the entire content of the stack in last-in first-out order without changing the stack content

See below for a sample dialogue.

In this assignment, both correctness and efficiency of your programs are important.

Example Dialogue (input can be read from a specific file, or can be directed to a file):

E <ENTER>

empty

P 1

P 10

E <ENTER>

not empty

K

10 1

P 19

R

19

R

10

E <ENTER>

not empty

P 5

P 20

K

20 5 1

E <ENTER>

not empty

R

20

R

5

R

1

R

nothing to pop off; stack is empty

K

<nothing to display>

P 2

E <ENTER>

not empty

K

2