Name:_____   Roll #: _____

National University of Computer and Emerging Sciences, Lahore Campus

	Course: Program: Duration: Paper Date: Section: Exam:	Data Structures BS(Computer Science) 90 Minutes 19-Oct-2020 ALL Midterm Exam 1	Course Code: Semester: Total Marks: Page(s): Section: Roll No:	CS 218 Fall 2020 50 5

Instruction/Notes: Answer in the space provided

You can ask for rough sheets but **they will not be graded or marked**

In case of confusion or ambiguity make a reasonable assumption. Questions are not allowed

Good luck!

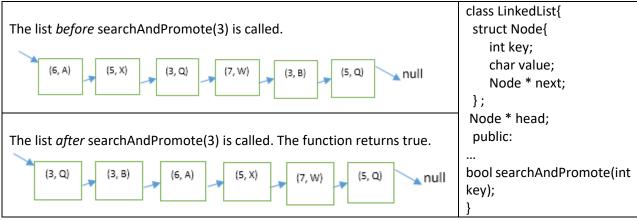**Question 1:**   **(Marks: 12+3)**

We wish to add a method called searchAndPromote to our singly linked list class. Each node in our list contains an integer key, a character value, and the next pointer. The function searchAndPromote searches the list for a given key, and if found, promotes the node containing that key to the start of the list. The idea is that the keys being searched for more recently are more likely to be processed again in the future, so bringing them to the start of the list will speed up the subsequent updates and deletes that need to process keys based on specific (key, value).

The keys in our list are not unique. Therefore, searchAndPromote must *promote all nodes with the key being searched for*. Moreover, while promoting these nodes, we need to make sure that the nodes appear at the start of the list in the same order in which they were found during search. The following example should make this clear. The table below also contains the definitions of node and list:

The list *before* searchAndPromote(3) is called.
 	
	class LinkedList{
	  struct Node{
	    int key;
	    char value;
	    Node * next;
	  } ;
	 Node * head;
	  public:
	…
	bool searchAndPromote(int key);
	}

The list *after* searchAndPromote(3) is called. The function returns true. 	

As you can see in this example, since (3, Q) is found before (3, B) during the search, the order of these nodes at the start of the list is (3, Q), (3, B).

Note: you must not delete or create new nodes, but simply reconnect the existing nodes in the desired fashion.

Write down the C++ definition for the function: bool searchAndPromote(int key) as explained above

What is the time complexity of this function given that there are n nodes in the list?

Question 2:   **(Marks: 12+3)**

Write down a C++ implementation using stack to determine if an input character string is of the form

$$s_1 \ \$s_1^R \$s_2 \ \$s_2^R \ ... s_k \ \$s_k^R$$

where s_i is a string consisting of different combinations of letters A and B and s_i^R is the reverse of s_i (that is. if s_i ="ABABBA", s_i^R must equal "ABBABA"). Thus a string is in proper form if it consists of any number of such strings separated by character '$'. At each point you may read only the next character of the string. Sample input and output is shown below

Input: ABAB**$**BABA**$**ABBBAB**$**BABBBA
Output: The string is in proper format
Input: ABAB**$**BABA**$**ABBBA**$**ABBAB
Output: The string is not in proper format

What is the time complexity of your function if input string has n characters?

Question 3:   **(Marks: 10+5+5)**
Part(a)
Perform a step count analysis on the following program and derive an equation for T(n) for the worst case.

`void mystery(int A[], int n){`
`    int i = n-1;`
`    for(int i=n-1;i>0 && A[i]<A[i/2]){`
`            swap(A[i],A[i/2]);`
`            i = i/2;`
`    }`
`}`
	`res = 0;`
	`for( i = 1; i < n; ++i )`
	`        for( j = 1; j < i ; ++j )`
	`                for( k = 0; k <n; k*2 )`
	`                        ++res;`

Solution:	

Part(b)
We want to implement a class queue using an object of circular singly linked list as its data member to store the elements of the queue. The class of circular linked list has only one data member tail of type node* that points to the last element of the list if list is not empty and null otherwise . How we can implement the insert (enqueue) and remove(dequeuer) functions of class queue using insertion and removal functions of circular linked list. What will be time complexity of these functions? No need to write code just explain your answer in 3-4 lines.

Part(c)
We wish to implement "UNDO" functionality in a Text Editor. We need to remember only **the last ten operations.** Which data structure would be best(in terms of space and time) to implement such a stack: Single Linked list, Double linked list, Circular linked list, Array, or Circular Array. Explain your answer.