Kingdom of Saudi Arabia
Ministry of Higher Education
College of computer & information technology
Department of Computer Science

Qassim University جامعة القصيم

المملكة العربية السعودية
وزارة التعليم العالي
كلية الحاسب وتقنية المعلومات
قسم علوم الحاسب

Final Exam
First Semester: 1437-38 **(371)**
Time Allowed: 2 hours
Subject: Data Structures (CSC-214) (CS-214)
Dated:21/04/1438 | 19-01-2017

ID. :_____
Name:_____
Serial no. _____

For Instructor use only

Outcomes	A	B	C	D	E	F	G	H	I	J	K
Course covering Level		H	H							M	

Question	1	2	3	4	5
Covered Outcome	B,C	C,J	B,C	C,J	B,J
Allocated mark	8	9	9	9	5
Earned mark					

Total					
A	B	C	I	J	exam
					40

Q-1: Choose the correct answer: (each has ½ mark) ***[8 Marks]***

A: Assume **Curr** points to Last node and **Prev** points to second last node of a **Circular Linked List**. Which is used to delete the Last node from linked list? (only click one option)
a. Prev -> next = Head **b.** Curr->next =Head **c.** Head = Curr->next **d.** None of these

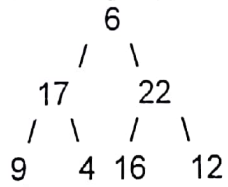
B: What can be the maximum number of nodes at a particular level in a complete binary tree. Assume $L = 4$, where L is level number.
a. 8 **b.** 31 **c.** 16 **d.** 4 **e.** 15 **f.** None of these

C: Arithmetic expressions always create?
a. Two-Tree /Strictly Binary Tree **b.** Complete Tree **c.** General Tree **d.** None of these

D: In specific, the complexity of Binary Search Tree using these values will be (5, 8, 12, 15, 20, 25, 28, 32)
a. O(log n) **b.** O(n) **c.** O(n/2) **d.** None of these

E: If Left-Subtree and Left-Child (LL) imbalance found in AVL tree, what type of rotation is needed.
a. Single Right Rotation **b.** Double (Right and then Left Rotation)
c. Single Left Rotation **d.** Double (Left and then Right Rotation) **e.** None of these

F: If the binary tree below is printed by a preorder traversal, what will the result be?



a. 9, 4, 17, 16, 12, 22, 6 **b.** 9, 17, 6, 4, 16, 22, 12 **c.** 6, 9, 17, 4, 16, 22, 12
d. 6, 17, 22, 9, 4, 16, 12 **e.** 6, 17, 9, 4, 22, 16, 12 **f.** None of these

G: Assume **Curr** points to Last node and **Prev** points to second last node of linked list. Which is not used to delete the Last node from linked list?
a. Prev -> next = NULL **b.** Head = Head ->next **c.** Prev->next = Curr->next **d.** both a & c

Scanned by CamScanner

H: In a linked list implementation of **queue**, keeping track of a **front** node and a **rear** node with two reference variables. Which of these reference variables will change during an insertion into an EMPTY queue?

a. Neither changes **b.** Only front changes. **c.** Only rear changes. **d.** Both front and rear.

I: Balance Factor (BF) of every node in AVL tree can be calculated as:

a. Number of Edges on Right sub-tree - Number of Edges on Left sub-tree
b. Number of Edges on Left sub-tree - Number of Edges on Right sub-tree
c. Left successor - Right Successor **d.** None of these

J: Construction of Binary Search Tree with duplicate option (secondary key), every node has:

a. All less and unique values on left-sub-tree and all bigger or similar/equal on right-sub-tree
b. Duplicate Values on left-sub-tree and unique values on right-sub-tree
c. All values must be unique **d.** None of these

K: If (Head->Forw = = Head && Head->Back = = Head) of a Circular Doubly Header linked List then:

a. List has many data nodes **b.** One data node is there **c.** List is Empty **d.** None of these

L: If you are going to construct AVL tree using these values: (8, 4, 5), What type of rotation is needed to make it height-balanced:

a. Single Right Rotation **b.** Double (Right and then Left Rotation)
c. Single Left Rotation **d.** Double (Left and then Right Rotation) **e.** None of these

M: In delete operation of BST, we need inorder successor of a node when the node to be deleted has 2 children. Which of the following is true about inorder successor needed in delete operation?

a. Inorder Successor is always a leaf node.
b. Inorder successor is always either a leaf node or a node with empty left sub-tree.
c. Inorder successor may be a father of its left child.
d. Inorder successor is always either a leaf node or a node with empty right sub-tree..

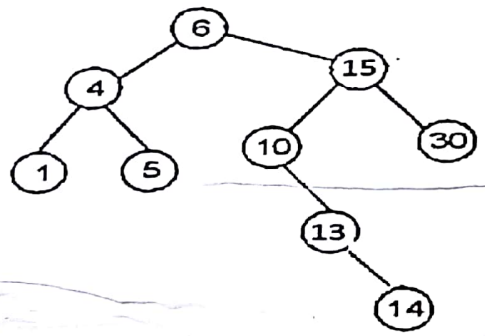
N: In a linked list implementation of queue, keeping track of a front node and a rear node with two reference variables. Which of these reference variables will change during an insertion into a NONEMPTY queue?

a. Neither changes **b.** Only front changes. **c.** Only rear changes. **d.** Both front and rear.

O: Enqueue('D'); Enqueue('C'); Dequeue( ); Enqueue('A'); Dequeue( ); Enqueue('A'); Enqueue('B'); Enqueue('C'); Dequeue( ); are done with an empty queue. What are sequence of values available on the Queue.

a. DCAABC **b.** DCABC **c.** DCAB **d.** ABC

P: Write Balance Factor on every node of the following BST:



Scanned by CamScanner

Q-2-A: Using the above tree given on (Q-1- P), what will be the output of following functions if called from main( ) function using original Root of the Tree. [4 Marks]

e.g. cout << fun1(ROOT); AND cout<<fun2(ROOT);

```
void func1(struct NODE *N)
{ if (N)
    { if(N->Left)  func1(N->Left);
      cout<<N->info<<", ";
      if(N->Right) func1(N->Right);
    }
}
```

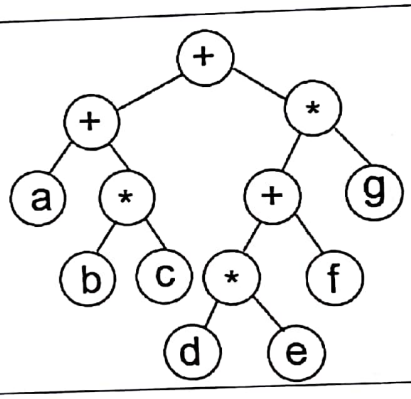
OUTPUT:

```
void func2(struct NODE *N)
{ if (N)
    { if(N->Right) func1(N->Right);
      cout<<N->info<<", ";
      if(N->Left)  func1(N->Left);
    }
}
```

OUTPUT:

Q-2-B: Using the following given tree structure and sample Tree, write the output of the given function. Assume you have called this function from main( ) { DO (Root); } using original Root: [5 Marks]

```
struct  treeNode
   {  char   symbol;
      struct  treeNode *Left;
      struct  treeNode  *Right;
   };
```

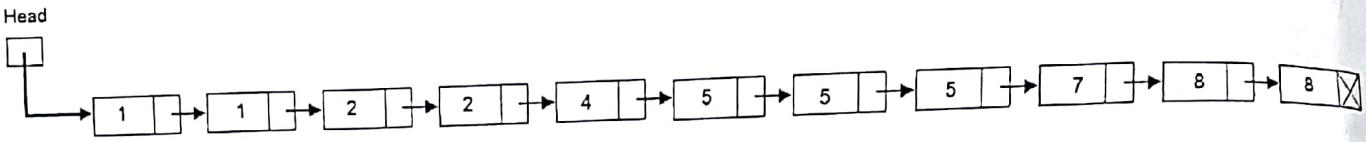



```
void DO (treeNode  *T)
{
   if( T != NULL )
   {
     if(T->Left != NULL && T->Right != NULL) cout<<"(";
     DO (T->Left);
     cout << T->symbol <<" ";
     DO (T->Right);
     if(T->Left != NULL && T->Right != NULL)  cout<<")";
   }
}
```

OUTPUT:

Scanned by CamScanner

Q-3-A: Suppose that a ***Sorted*** Liked List is defined in memory as given below: *[2 Marks]*

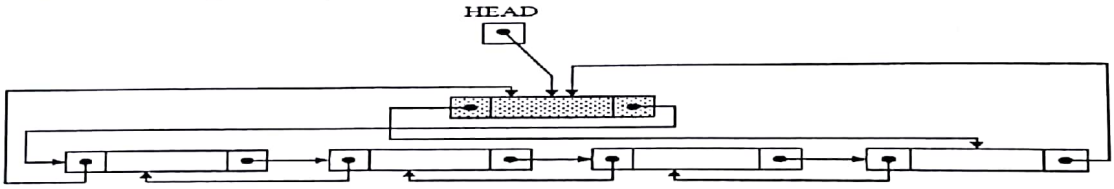


What does the following function do with this Linked List. Explain your answer by redrawing the Figure showing current state of Linked List.

```
void ABC( )
   {  struct LIST  *P = Head;
        while( P != NULL)
            {  while ((P->Next != NULL) && (P->Next->Info == P->Info))
                       {
                           P->Next = P->Next->Next;
                       }
                P = P->Next;
            }
   }
```

Answer:

Q-3-B: Given the following structure of the Doubly Linked list: *[5 Marks]*



The type of each element is the following structure:

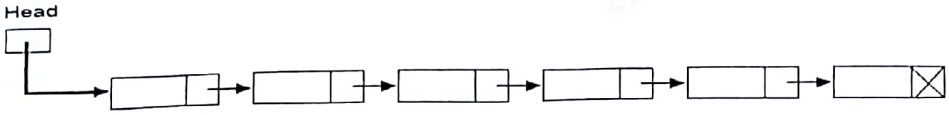
```
struct node
{  int info;
   node *back;
  node *forw; };
```

node * Head= new node ;//global variable

Write a function with the following prototype: int Count (int X);
that counts the number of times a given integer X occurs in the list.

Scanned by CamScanner

Q-3-C: Suppose we have following linked List in memory with some nodes. *[2 Marks]*



Show what would happen if we applied the following statements to this List.

```
Node *T = Head;
while ( T->Next != NULL)
                { T = T->Next; }
T->Next = Head;
```

Explain your answer by redrawing the Figure showing current state of Linked List.

Answer:

Q-4-A: Write a function to search for a specific value in the binary search tree and return true if the value is found, false otherwise. *[5 Marks]*

```
bool Search( NODE *pRoot, int Item )
  {



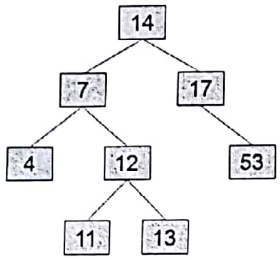

  }
```


Scanned by CamScanner

Q-4-B: Given a sequence of numbers as key values: 11, 6, 8, 19, 4, 10, 5, 17, 43, 49, 31, 18 *[4 Marks]*

A. Draw a binary search tree by inserting the above numbers into Binary Search Tree.	*B. Re-draw BST after deleting the* **Root** *Node.*

Q-5-- Following is an AVL Tree. Insert 8 in this AVL tree and then make it height balanced. Show every step. *[5 Marks]*



Scanned by CamScanner