Roll Number:________________ No. of Pages: **02**

Thapar University, Patiala

Department of Computer Science and Engineering

MID SEMESTER EXAMINATION

B. E. (Second/Third Year): Semester-IV/VI (2016)	Course Code: UCS406 Course Name: Data Structures and Algorithms
March25, 2017	Saturday, 10:30 –12:30Hrs
Time: 2 Hours, M. Marks: 25	Name Of Faculty: MAK,VIK,SP, TPB, SUK,RAJ, ASG, AKM

Note: Attempt all questions. Assume missing data, if any, suitably

Q1. Write an algorithm for Quicksort.
Given an array of ten integers: 5 3 8 9 1 7 0 2 6 4
Suppose we partition this array using quicksort's partition function. Show all the steps to get the resulting array after the first complete execution of the partition procedure (**first selected pivot gets right position**) **(3+1=4)**

Q2. Solve the following recurrence relation. After getting the final result, express it in Big-O notation

$$T(n) = \begin{bmatrix} 1 & if & n = 1 \\ T(n/2) + n & if & n > 1 \end{bmatrix}$$ **(3)**

Q3. Convert the following infix expressions into its equivalent postfix expression using stack;

A) A * (B*D)/E-F*(G+H/K)

B) Evaluate the result obtained in part a with given values: A=9,B=4,D=E=F=K=2, G=5,H=6 **(2+1)**

Q4. Modify the Bubble Sort algorithm such that **if after i^{th} pass we get sorted array**, it will not repeat the process **for $(i+1)^{th}$** pass. **(2)**

Q5. Consider the following operations on a Queue data structure that stores int values.

```
enqueue(3);
enqueue(5);
enqueue(9);
printf("%d",dequeue( ));          // d1
enqueue(2);
enqueue(4);
printf("%d",dequeue( ));          // d2
printf("%d",dequeue( ));          // d3
enqueue(1);
enqueue(8);
```

I. After the code above executes, how many elements would remain in the queue?

II. What value is returned by the last dequeue operation (denoted above with a d3 in comments)?

III. If we replace the *printf* statements (denoted in comments as d1, d2 and d3) with the statement enqueue(dequeue( )); the Queue would contain which order of int values after all instructions have executed? **(0.5+0.5+1)**

PTO

Q6. Write an algorithm for singly linked list to:

A) Insert a value at a particular position

B) Delete a particular value **(2+2)**

Q7. A) Consider the following function that takes reference to head of a Doubly Linked List as parameter. Assume that a node of doubly linked list has previous pointer as *prev* and next pointer as *next*. The function is called with the list containing the integers 1, 2, 3, 4, 5, 6 in the given order. What does the following function do? What will be the contents of the list after the function completes execution? **(1+1)**

```
void fun(struct node **head_ref)
{     struct node *temp = NULL; struct node *current = *head_ref;
while (current !=  NULL)
    {      temp = current->prev;
current->prev = current->next;
current->next = temp;
current = current->prev;
    }
if(temp != NULL )
        *head_ref = temp->prev;
}
```

B) Consider following function that takes a single-linked list of integers as a parameter. Assume that a node of singly linked list has data field as *value* and next pointer as *next*. The function is called with the list containing the integers 1, 2, 3, 4, 5, 6, 7 in the given order. What does the following function do? What will be the contents of the list after the function completes execution? (1+1)

```
void fun(struct node *list)
{
  struct node *p, * q;
  int temp;
  if ((!list) || !list->next)
      return;
  p = list;
  q = list->next;
  while(q)
  {
     temp = p->value;
     p->value = q->value;
     q->value = temp;
     p = q->next;
     q = p?p->next:0;
  }
}
```

Q8. Suppose the numbers 0, 1, 2, ..., 9 were pushed onto a stack **in that order**, but that pops occurred at random points between the various pushes. Each pop will also print the deleted element.

A) List out the **valid sequence** of push and pop operation to achieve the output :
3, 2, 6, 5, 7, 4, 1, 0, 9, 8.

B) Explain why it is not possible that 3, 2, 6, 4, 7, 5, 1, 0, 9, 8 is a valid sequence in which the values could have been popped off the stack (2+1=3)