



AUTUMN MID SEMESTER EXAMINATION-2017

Data Structure & Algorithms

[CS-2001]

Full Marks: 25

Time: 1.5 Hours

Answer any five questions including question No.1 which is compulsory.

The figures in the margin indicate full marks.

Candidates are required to give their answers in their own words as far as practicable and all parts of a question should be answered at one place only.

Q1 Answer the following questions:

(1 x 5)

- a) Find the worst case time complexity of the following C function.

```
int func(int n)
{
    int i, j, k, p, q = 0;
    for (i = 1; i < n; ++i)
    {
        p = 0;
        for (j = n; j > 1; j = j/2)
            ++p;
        for (k = 1; k < p; k = k*2)
            ++q;
    }
    return q;
}
```

- b) An array $X[10 \dots 25, 15 \dots 40]$ requires four bytes of storage for each element. Base address of the array X is 1500. Calculate the size of array X and determine the address of $X[15][20]$ considering column-major representation.
- c) Consider the sparse matrix shown below. Diagrammatically represent it in triplet and single header linked list.

		0	1	2	3	4	← Column Index
Row index →	0	1	4	0	2	0	
	1	0	0	0	0	2	
	2	0	0	7	0	0	
	3	0	0	4	5	0	
	4	0	0	4	0	0	
	5	0	6	0	0	0	

- d) In a single linked list, a node pointer p points to a specific node. The data in p and its previous and next node are same. Write a code segment to delete the duplicate nodes. E.g. if input: 1->3->3->3->5->6 then output: 1->3->5->6
- e) What is the output of the following function for a single linked list with the values 1 -> 2 -> 3 -> 4 -> 5-> 6? start is pointing to the first node in the list.

```

void fun(struct node *start)
{
    if (start == NULL) return;
    if (start->next != NULL) fun (start->next->next);
    printf("%d", start->data);
}

```

- Q2 a) Define and explain data structure and abstract data type (ADT) with suitable examples. (2.5)
- b) Design a data structure to represent a polynomial and write an algorithm or pseudocode or C code snippet to multiply two polynomials. (2.5)
- Q3 a) Write a function rearrange() that exchanges data of every node with its next node, starting from the first node in a single linked list. (2.5)
 Example: Input: 1->2->3->4->5->6->7 Output: 2->1->4->3->6->5->7
 Input: 1->2 Output: 2->1
 Input: 1 Output: 1
- b) Write an algorithm or pseudocode or C code snippet to reverse a double linked list by changing node values. (2.5)
 Example: Input: 1->2->3->4->5->6->7 Output: 7->6->5->4->3->2->1
- Q4 a) Write an algorithm or pseudocode or C code snippet to delete a node with a given data/key and its previous node (if available) in a single linked list. (2.5)
- b) What are advantages and disadvantages of circular linked list over single linked list? Write an algorithm or pseudocode or C code snippet to convert a double linked list into circular double linked list. (2.5)
- Q5 Write a function which will traverse the single linked list only once and split the original linked list into two sub-lists, where first and second sub-list contains the even position nodes and odd position nodes of the original list respectively, and then join the second sub-list at the end of the first sub-list. Example: If the list contains 3->4->2->1->7->9->8, then the function need to produce 3->2->7->8->4->1->9. (5)
- Q6 Write a function that will traverse a 1-D array from the beginning to the end and reverse all the increasing and decreasing sequence of components present in the array. Component is the collection of continuous elements either in increasing or decreasing order. Example: Let an array contains 1, 2, 3, 7, 4, 2, 9, 7, 8 elements. Here the components are "1, 2, 3, 7", "4, 2", "9, 7", and "8". The function should produce the array with elements 7, 3, 2, 1, 2, 4, 7, 9, 8. (5)
- Q7 Write Stack ADT. Write Push and Pop operations in the stack. Convert the infix expression $(a + b * c / (d * e + f) / g ^ h)$ to postfix expression using stack. (5)