Data Structure & Algorithms/CS-301/CS/IT/3rd Semester/2014

Mid Semester Examination

School of Computer Engineering

KIIT UNIVERSITY, BHUBANESWAR

Time: 2hrs **Full Mark: 50**

[ANSWER FIVE QUESTIONS INCLUDING QUESTION NUMBER 1]

1. Answer all the questions [2 X 5]

a. Find the equivalent postfix expression of the following infix expression using STACK.
a * ((b – c) * (d / e ^ f) - g) / h

Ans: **abc-def^/*g-*h/**

b. Find time complexity of the following code segment.

```
for (i=1 ; i<=n ; i=i*2){
     for (j=n ; j>=1 ; j=j/2){
               Statement_1;
     }
}
```

Ans: **O((log n)^2)**

Hint:

```
for (i=1 ; i<=n ; i=i*2){ // 1, 2, 4, 8, …, n
     for (j=n ; j>=1 ; j=j/2){ //n, n/2, n/4, n/8, …, 1
               Statement_1;
     }
}
```

c. Let a pointer called ***head*** is pointing to the first element of a doubly circular linked list. Write a function to reverse the content of the list by traversing each node only once.

Ans:

```
ptr1=head;
ptr2=head->prv;
while(ptr1->next != ptr2 && ptr1 != ptr2){
     tmp=ptr1->data;
     ptr1->data=ptr2->data;
     ptr2->data=tmp
}
```

d. Write down the overflow and underflow condition of a circular queue implemented as an array.

Ans:

Overflow

front=rear+1 || (front=0 && rear=MAX-1)

or

(rear+1)%MAX=front

Underflow

front == -1 || rear == -1

e. Let a two dimensional array having row range (40:70) and column range (50:100). The whole array is stored in row major order. If the address of location (41, 60) is *x* and address of location (70, 95) is **y**, and then find the address of location (65, 80).

Ans:

The number of nodes before location (41, 60)=(41-40)(100-50+1)+(60-50)=61
The number of nodes before location (70, 95)=(70-40)(100-50+1)+(95-50)=1575
Size of each element=(y-x)/(1575-61)
The number of nodes before location (65, 80)=(65-40)(100-50+1)+(80-50)=1305
the address of location (65, 80)=x + (1305-61) * (size of each element)

2. (a) How to represent a polynomial expression using linked list. Write pseudo code/ function code to add two polynomials. **[7]**

Ans:

```
struct polynomial {
        int cof;
        int exp;
        struct polynomial *next;
};
```

(b) Compare and contrast single and double linked list. **[3]**

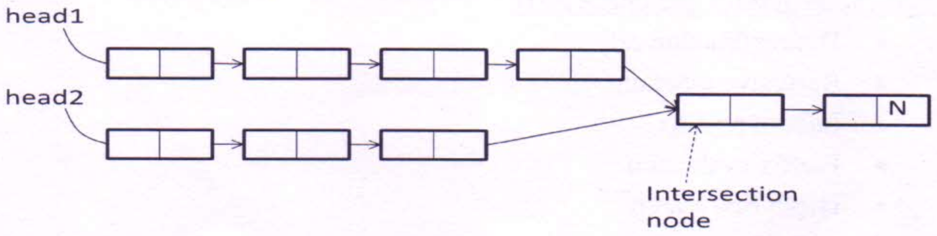
3. (a) What is sparse matrix? How to effectively represent sparse matrix? Write pseudo code/ function code to transpose a sparse matrix. **[7]**

(b) Write down the pseudo code/ function code to implement insertion and deletion operation in a queue using two stacks. **[3]**

Ans:

```
Void Insertion(int data){
        If(is_full(&new_STACK) && !is_empty(&old_STACK))
                Print("Cannot insert");
        Else if(is_full(&new_STACK) && is_empty(&old_STACK)){
                While(1){
                        If(is_empty(&new_STACK))
                                Break;
                        Else
                Push(pop(&new_STACK), &old_STACK);
        }
        Push(data, &new_STACK);
    }else
        Push(data, &new_STACK);
}
Int Deletion(int data){
    If(is_empty(&new_STACK) && is_empty(&old_STACK))
        Printf("Queue is empty");
        Return -1;
    Else if(!is_empty(&old_STACK))
        Return  pop((&old_STACK);
    Else if(is_empty(&old_STACK)){
        While(1){
            If(is_empty(&new_STACK))
                Break;
            Else
                Push(pop(&new_STACK), &old_STACK);
        }
        Return  pop((&old_STACK);
    }
}
```

4. (a) Write pseudo code/ function code to find intersection node's data present in the two linked list, where intersection node is represented as follows. **[7]**



Ans:

```
Void intersection_node(struct node *h1, struct node *h2){
    For(;h1!=NULL;h1=h1->next){
        For(ptr=h2;ptr !=NULL;ptr = ptr->next){
            If(h1==ptr){
                Printf("FOUND=%d",ptr->data);
                Return;
            }
        }
    }
    Printf("NOT FOUND");
    Return;
}
```

(NOTE: This is not the only sol^n. we can do this in o(n)).

(b) What is abstract data type (ADT)? Write down the representation of stack ADT. **[3]**

Ans:

An ADT is a set of elements with some well defined operation.

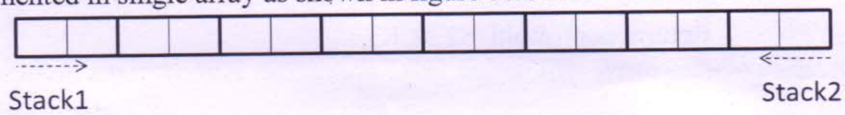
<u>The representation of stack ADT using array</u>

```
typedef struct{
        int data[MAX];
        int top;
}STACK_ADT;
```

<u>The representation of stack ADT using linked list</u>

```
struct node{
        int data;
        struct node *next;
};
typedef struct{
        struct node *top;
}STACK_ADT;
```

5. (a) Write pseudo code/ function code to perform PUSH and POP operation of two numbers of stacks implemented in single array as shown in figure below. **[7]**



Ans:

(b) Write in detail the application of stack and queue ADT. **[3]**

<u>application of stack and queue ADT</u>

- During function call
- Recursive function
- Infix to post fix
- Postfix evaluation
- Depth first search
- ….

<u>application of stack and queue ADT</u>

- Process scheduling
- Job que
- Breadth first search
- ….

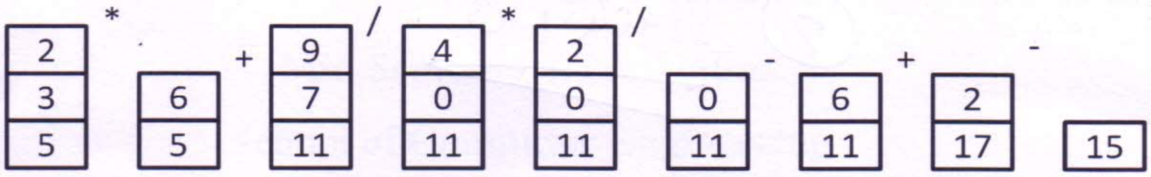
6. (a) Write pseudo code/ function code to implement the functionalities of output restricted double ended circular queue using an array. **[7]**

Ans:

(b) Evaluation of the following postfix expression. **[3]**

5 3 2 * + 7 9 / 4 * 2 / - 6 + 2 –

Ans:



*** BEST OF LUCK ***