UNIVERSITI TUN HUSSEIN ONN MALAYSIA

FINAL EXAMINATION
SEMESTER II
SESSION 2010/2011

COURSE NAME	:	DATA STRUCTURE & ALGORITHM
COURSE CODE	:	BIT 1073 / BIT 10703
PROGRAMME	:	BACHELOR OF INFORMATION TECHNOLOGY
EXAMINATION DATE	:	APRIL / MAY 2011
DURATION	:	2 HOURS 30 MINUTES
INSTRUCTION	:	ANSWER **ALL** QUESTIONS IN SECTION A AND **THREE (3)** QUESTIONS IN SECTION B

THIS QUESTION PAPER CONTAINS **TWELVE (12)** PAGES

SECTION A

Instruction: Answer **ALL** questions.

Q1 Which of the following statement is **FALSE**?

(a) Arrays are dense lists and static data structure.
(b) Data elements in a linked list need not be stored in adjacent space in memory.
(c) A linked list is a linear static data structure.
(d) Linked lists are collection of nodes that contain information part and next pointer.

Q2 When new data are to be inserted into a data structure, but there is no available space; this situation is usually called ________________.

(a) underflow
(b) overflow
(c) housefull
(d) saturated

Q3 Linked lists are best suited for ________________.

(a) relatively permanent collections of data
(b) the size of the structure and the data in the structure are constantly changing
(c) both of above situation
(d) none of above situation

Q4 Which of the following is not the required condition for binary search algorithm?

(a) The list must be sorted.
(b) There should be the direct access to the middle element in any sublist.
(c) There must be mechanism to delete and/or insert elements in a list.
(d) None of the above.

Q5 Which of the following sorting algorithm is of divide-and-conquer type?

(a) Bubble sort
(b) Insertion sort
(c) Quick sort
(d) All of the above

Q6 In a graph, `e = (u, v)` means ________________.

(a) u is adjacent to v but v is not adjacent to u
(b) e begins at u and ends at v
(c) u is processor and v is successor
(d) both b and c

Q7 If every node u in G is adjacent to every other node v in G, a graph is said to be ____________________.

(a) isolated
(b) complete
(c) finite
(d) strongly connected

Q8 The operation of processing each element in a list is known as _______________.

(a) sorting
(b) merging
(c) inserting
(d) traversal

Q9 The worst case occur in linear search algorithm when _________________.

(a) the item is somewhere in the middle of the array
(b) the item is not in the array at all
(c) the item is the last element in the array
(d) the item is the last element in the array or is not there at all

Q10 The post order traversal of a binary tree is DEBFCA. Find out the pre-order traversal for the same binary tree.

(a) ABFCDE
(b) ADBFEC
(c) ABDECF
(d) ABDCEF

SECTION B

Instruction: Answer **THREE (3)** questions only.

Q11 Below is a C code segment written to do a set of tasks. Go through the code segment carefully and answer the following questions.

```
#include <stdio.h>
#include <ctype.h>
#define MAXSIZE 200

int myList[MAXSIZE];
int posA, posB;

void main()
{
   void doSomething2(int);
   int doSomething1();
   int index=1, i, num;
   posA = 0;
   posB = 0;

   printf("Program for DEMO ");

   while(index != 3)
   {
      printf("\n MAIN MENU: ");
      printf("\n 1. DO Task A");
      printf("\n 2. DO Task B" );
      printf("\n 3. Exit");
      printf("\n Input1: ");
      scanf("%d",&index);

      switch(index)
      {
      case 1:
         printf(" Input2: ");
         scanf("%d",&num);
         doSomething2(num);
         break;
      case 2: i=doSomething1();
         printf("\n Value is %d",i);
         break;
      default: printf("Bye ... ");
         return;
   }
   }
}
```

```
void doSomething2(int a)
{
  if(posB > MAXSIZE){
           printf("\n Condition 1: ");
           return;
}
  else{
    myList[posB]=a;
    posB++;
    printf("\n Value pos A=%d, posB=%d", myList[%d],
myList[];
            posA, posB);
    }
}

int doSomething1()
{
  int a;
  if(posA == posB){
     printf("\n Condition 2: ");
     return(0);
  }
  else{
     a = myList[posA];
     posA++;
  }
     return(a);
}
```

(a) What are the appropriate names for posA dan posB?

(2 marks)

(b) Describe each of the followings:

(i) Condition 1
(ii) Condition 2
(iii) Task A
(iv) Task B

(6 marks)

(c) Consider the following input sequence:

```
1  <enter>
2  <enter>
1  <enter>
3  <enter>
1  <enter>
4  <enter>
2  <enter>
2  <enter>
2  <enter>
2  <enter>
3  <enter>
```

Write the output of the program when you enter the input sequence during the program execution. Remember that everytime you press `<enter>` after any input data, your program will respond accordingly and communicate interactively with you.

(Note: Do not miss any lines of output especially the prompts generated by the program).

(10 marks)

(d) What is the content of myList[ ] when the program ends?

(2 marks)

Q12 **(a)** Given the following integer list:

```
88     9      66     115    39     5      599
```

Show a trace (step by step) for each execution of :

(i) Insertion sort.

(4 marks)

(ii) Bubble sort.

(4 marks)

(b) Given the following C code segment.

```
void xSort(int a[5], int n)
{
      for (int k=1; k<n; k++) {

      for (int i=k; i>0 && a[i-1] > a[i]; i--) {
            int temp = a[i-1];
            a[i-1] = a[i];
            a[i] = temp; // line 7
                         // line 8
            }
      }
}
```

Assuming there is a `printf()` command to print all elements of array `a[]` on `line 8`, trace the operation of `xSort` by showing the contents of the array in **Table 1**. Assume the initial value for `a[ ]`is `{23, 10, 34, 2, 12}` and *n* is 5.

Table 1: Operation of xSort at *k* pass.

k	a[0]	a[1]	a[2]	a[3]	a[4]
initial	23	10	34	2	12
1					
2					
3					
4					
5					
6					

(4 marks)

(c) Given the C code segment below.

```
#include <stdio.h>
int F(int);
int main (){
    int k;
    k = F(4);
    printf("\n k = %d", k);  //line 6
    return 0;
}

int F( int x) {
    int z = 10;
    if(x > 1)
        z = z * F(x-1); //line 13
    return z;
}
```

(i) Write the output of the program, printed at line 6.

(4 marks)

(ii) What will be the output if line 13 in the C code segment is changed to `z = x * F(x-1);` ?

(4 marks)

Q13 **(a)** Consider a postfix notation of arithmetic expression as given:

```
ABC * + DE * F + G / -
```

(i) Write the infix notation of the given expression.

(4 marks)

(ii) Construct the expression by using a binary tree.

(4 marks)

(iii) What is the structure of the binary tree?

(2 marks)

(b) Consider the binary tree in **Figure Q13(b)** .

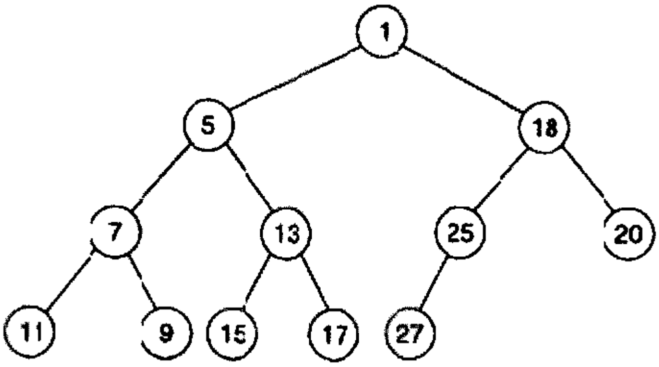

Figure Q13(b)

(i) Draw the tree to show at each step for deleting the node with key 5 by preserving the tree structure.

(4 marks)

(ii) Write a pseudo-code to delete an arbitrary node from such a binary tree with *n* nodes that preserves the structure.

(4 marks)

(iii) What is the complexity for best and worst case in searching for a value in a binary tree?

(2 marks)

Q14 **(a)** A directed graph is given in **Figure Q14(a):**

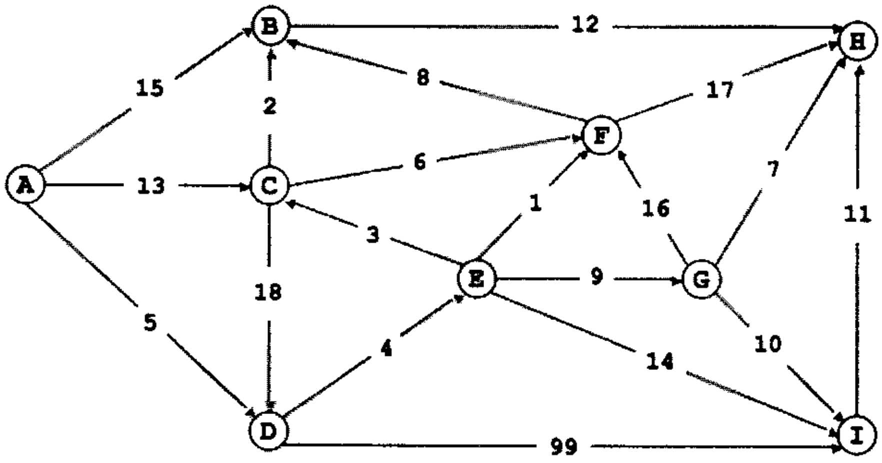

Figure Q14(a)

(i) Apply the Dijkstra's algorithm to find a Minimal Spanning Tree (MST) starting from vertex *C* to every other vertices. To show your work, you need to fill in **Table 2.**

Table 2: Graph traversal for Minimal Spanning Tree using Dijkstra's Algorithm

Vertex	A	B	D	E	F	G	H	I
C								

(5 marks)

(ii) Draw the graph in **Figure Q14(a)** and show:

- all the selected edges in your MST,
- the sequence of edges being added to the Minimum Spanning Tree and
- the weight at each vertex.

(9 marks)

(b) Given the graph in **Figure Q14(b)**:

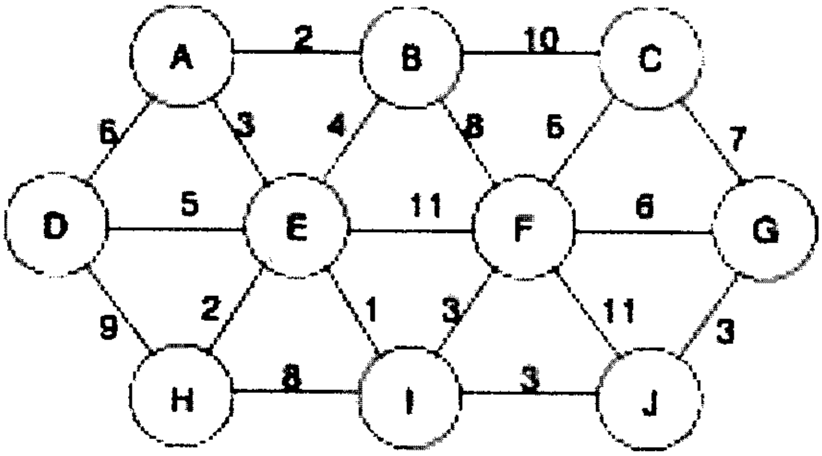

Figure Q14(b)

(i) Use Kruskal algorithm to find Minimum Spanning Tree (MST) by showing the sequence of your selected edges by completing the following **Table 3**.

Table 3: Graph traversal for Minimal Spanning Tree using Kruskal Algorithm.

Edge	Weight	Accept (Yes/No)

(4 marks)

(ii) Draw the graph to show the selected edges.

(2 marks)