SLIIT ACADEMY

Higher Diploma
in
Information Technology

Final Examination
Year 2, Semester 1 (2022)
January Intake

Design and Analysis of Algorithms (IT1205)

Duration: 3 Hours

Instructions to Candidates:

- This is a closed book examination.
- This paper contains 5 questions on 5 pages without the cover page.
- Answer all questions in the WORKBOOK provided.
- Read all questions before answering.
- The total marks obtainable for this examination is 100.

QUESTION ONE **(20 marks)**

a) Briefly explain Divide & Conquer algorithm design method? (2 marks)

b) MERGESORT Algorithm is given below in figure 1.

```
Procedure MERGESORT (A,p,r)
if p < r
    then    q ← ⌊(p + r )/2⌋
            MERGESORT (A,p,q)
            MERGESORT (A,q+1, r)
            MERGE (A,p,q,r)

Procedure MERGE (A, p, q, r)
n1 ← q - p + 1
n2 ← r - q
create arrays L[1.. n1 + 1] and R[1.. n2 + 1]
for i ← 1 to n1
    do L[i] ← A[p + i - 1]
for j ← 1 to n2
    do R[j] ← A[q + j]
L[n1 + 1] ← ∞
R[n2 + 1] ← ∞
i ← 1
j ← 1
for k ← p to r
    do if L[i] ≤ R[j]
          then A[k] ← L[i]
               i ← i + 1
       else
               A[k] ← R[j]
               j ← j + 1
```

Figure 1

i. Analyze the MergeSort algorithm given in figure 1 to find the running time equation. Justify your answer. (4 marks)

ii. Solve the running time equation in part **i** using recursion tree method. (5 marks)

iii. Solve the running time equation in part **i** using repeated substitution method. (5 marks)

iv. Solve the running time equation in part **i** using master method. (4 marks)

QUESTION TWO **(20 marks)**

a) Briefly describe the following terms using tree terminology. (4 x 1 mark)

i. binary tree
ii. depth of a node
iii. number of elements in height ***h*** binary tree
iv. height of a complete binary tree

b) Analyze the Heapify algorithm given in Figure 2 to find the running time. Justify your answer. (4 marks)

```
Procedure HEAPIFY (A,i)
l ← LEFT_CHILD (i);
r ← RIGHT_CHILD (i);
if l ≤ heap_size[A] and A[ l ] > A[ i ]
    then largest ← l;
    else largest ← i;
if r ≤ heap_size[A] and A[r] > A[largest]
     then largest ← r;
if largest ≠ i
      then exchange A[i] ↔ A[largest]
      HEAPIFY (A,largest)
```

Figure 2

c)

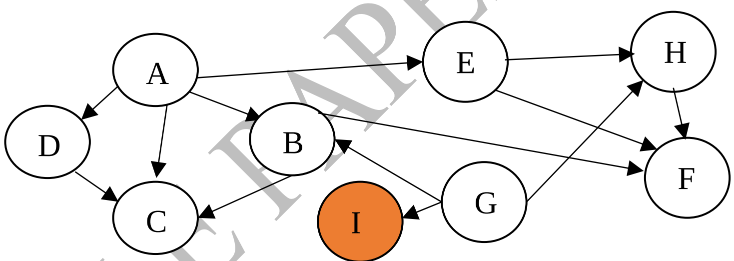

Figure 3

i. Represent the graph given in Fighre3 using Adjacency Matrix (2 marks)
.

ii. Illustrate the Breadth First Search for the graph given in Figure 3. (5 marks)
Source Vertex: A
Draw the graph & the queue for each step with time stamps, finally draw the breadth first tree.

iii. Illustrate the Depth First Search for the graph given in Figure 3. (5 marks)
Source Vertex: A
Draw the graph & the stack for each step with time stamps, finally draw the depth first tree/s.

QUESTION THREE **(20 marks)**

a) Briefly explain Greedy Method using an example. (2 marks)

b) What is a Minimum Cost Spanning Tree? Name its properties. (2 marks)

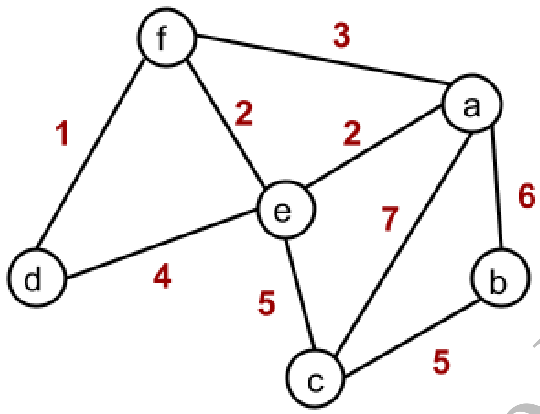

Figure 4

c) Using Kruskal's algorithm, find the Minimum Cost Spanning Tree (MCST) of the graph given in Figure 4, Calculate the cost of its MCST.
[Hint: use the alphabetical order to break ties.] (5 marks)

d) Using Prim's algorithm, find the Minimum Cost Spanning Tree (MCST) of the graph given in Figure 4, Calculate the cost of its MCST. Source vertex: a
[Hint: use the alphabetical order to break ties.] (5 marks)

e) Apply the Dijkstra's algorithm to find the shortest path from the vertex N to all the other vertices of the graph given in Figure 5. (6 marks)

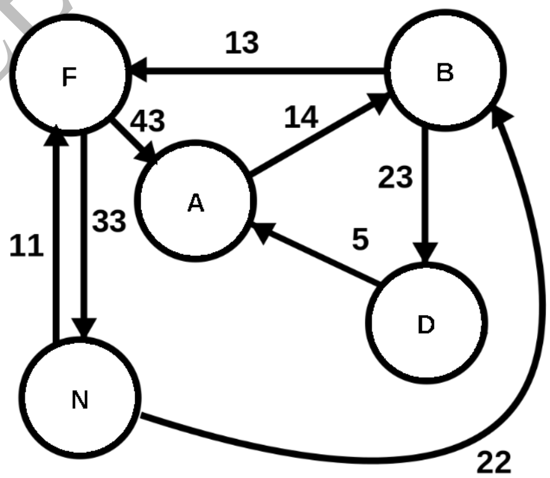

Figure 5

QUESTION FOUR **(20 marks)**

a) Briefly explain Dynamic Programming. (2 marks)

b) Briefly explain the elements of dynamic programming. (4 marks)

c) Given a chain $(A_1, A_2, ..., A_{n-1}, A_n)$ of n matrices, where for $i = 1,2, \ldots, n$ matrix A_i has dimension $p_{i-1} \times p_i$. Assume that $m[i,j]$ is the minimum number of scalar multiplications needed to compute the matrix $A_{i \ldots j} = A_i \times A_{i+1} \times ..... \times A_{j-1} \times A_j$ and it is defined below.

$$m[i,j] = \begin{cases} 0 & \text{if } i = j \\ \min\limits_{i \le k < j} \{m[i,k] + m[k+1,j]\} + p_{i-1} p_k p_j & \text{otherwise} \end{cases}$$

Consider the following set of metrics A_1, A_2, A_3 and A_4 with their dimensions of ***2* x *5*, *5* x *3*, *3* x *4*** and ***4* x *1*** respectively.

i. Draw and fill the *m* and *s* tables. (6 marks)

ii. Find the optimal parenthesizing of the matrices for the above sequence of matrices using algorithm given in Figure 6. (6 marks)

```
Print-Optimal-Parens (s, i, j)
if i=j
        then print "A"i
else
        print "("
        Print-Optimal-Parens (s, i, s[i,j])
        Print-Optimal-Parens (s, s[i,j]+1, j)
        print ")"
```

Figure 6

iii. Hence find the optimal number of scalar multiplications of the above sequence of matrices. (2 marks)

QUESTION FIVE **(20 marks)**

a) Explain the following, with examples. (2 x 1 mark)

i. Fixed length coding

ii. Variable length coding

b) i. What is the problem with the code words given in Table 1? (2 x 1 mark)

ii. How to overcome the problem identified in part **i**?

Symbol	Frequency	Code word
A	5	0
B	2	01
C	1	001

Table 1

c) Develop a code word for the text: **GOHOMEGONA**

i. Develop a code word for the above text using Shanon Fano algorithm. (4 marks)

ii. Develop a code word for the above text using Huffman algorithm. (4 marks)

iii. Develop a code word for the above text using Hu Tucker algorithm. (4 marks)

iv. Generate the symbol stream of above text using Ziv Lempel model. (4 marks)

Hint: Use the following ASCII values.

Symbol	ASCII Code
A	65
E	69
G	71
H	72
M	77
N	78
O	79

End of the question paper