COMPUTER SCIENCE
PAPER 1
(THEORY)

(Maximum Marks: 70)

(Time allowed: Three hours)

*(Candidates are allowed additional 15 minutes for **only** reading the paper.*
They must NOT start writing during this time.)

----

*Answer **all** questions in Part I (compulsory) and **six** questions from Part-II, choosing **two**
questions from Section-A, **two** from Section-B and **two** from Section-C .*

*All working, including rough work, should be done on the same sheet as the
rest of the answer.*

The intended marks for questions or parts of questions are given in brackets [ ].

PART I (20 Marks)

*Answer **all** questions.*

***While answering questions in this Part, indicate briefly your working and reasoning,
wherever required.***

Question 1

(a) State the *Commutative law* and prove it with the help of a truth table. **[1]**

(b) Convert the following expression into its canonical POS form: **[1]**

$$F(X, Y, Z) = (X+Y') \cdot (Y'+Z)$$

(c) Find the dual of: **[1]**

$$(A'+B) \cdot (1+B') = A'+B$$

(d) Verify the following proposition with the help of a truth table: **[1]**

$$(P \wedge Q) \vee (P \wedge \sim Q) = P$$

(e) If $F(A, B, C) = A' (BC' + B'C)$, then find F' **[1]**

Question 2

(a) What are *Wrapper classes?* Give *any two* examples. **[2]**

(b) A matrix A[m][m] is stored in the memory with each element requiring 4 bytes of
storage. If the base address at A[1][1] is 1500 and the address of A[4][5] is 1608,
determine the order of the matrix when it is stored in **Column Major Wise**. **[2]**

This Paper consists of 9 printed pages and 1 blank page.

(c) Convert the following *infix notation* to *postfix* form: **[2]**

$$A + (B - C * (D / E) * F)$$

(d) Define *Big 'O' notation*. State the *two* factors which determine the complexity of an algorithm. **[2]**

(e) What is *exceptional handling?* Also, state the purpose of *finally* block in a try catch statement. **[2]**

Question 3

The following is a function of some class which checks if a positive integer is a Palindrome number by returning true or false. (*A number is said to be palindrome if the reverse of the number is equal to the original number*.) The function does not use modulus (%) operator to extract digit. There are some places in the code marked by **?1?, ?2?, ?3?, ?4?, ?5?** which may be replaced by a statement / expression so that the function works properly.

```
boolean  PalindromeNum( int N )
{
        int rev= ?1?;
        int num=N;
        while( num>0)
        {
           int f= num/10;
           int s =  ?2?;
           int digit = num−?3?;
           rev=  ?4?  + digit;
           num /= ?5?;
        }
        if(  rev= =N )
           return true;
        else
           return false;
}
```

(i) What is the statement or expression at **?1?** **[1]**

(ii) What is the statement or expression at **?2?** **[1]**

(iii) What is the statement or expression at **?3?** **[1]**

(iv) What is the statement or expression at **?4?** **[1]**

(v) What is the statement or expression at **?5?** **[1]**

2

1218-868A

PART – **II (50 Marks)**

*Answer **six** questions in this part, choosing **two** questions from
Section A, **two** from Section B and **two** from Section C.*

SECTION - A

*Answer **any two** questions.*

Question 4

(a)   Given the Boolean function **F(A, B, C, D) = Σ (0,2,4,8,9,10,12,13).**

    (i)   Reduce the above expression by using 4-variable Karnaugh map, showing the various groups (i.e. octal, quads and pairs).   **[4]**

    (ii)   Draw the logic gate diagram for the reduced expression. Assume that the variables and their complements are available as inputs.   **[1]**

(b)   Given the Boolean function: **F(A, B, C, D) = π(3,4,5,6,7,10,11,14,15).**

    (i)   Reduce the above expression by using 4-variable Karnaugh map, showing the various groups (i.e. octal, quads and pairs).   **[4]**

    (ii)   Draw the logic gate diagram for the reduced expression. Assume that the variables and their complements are available as inputs.   **[1]**

Question 5

(a)   A training institute intends to give scholarships to its students as per the criteria given below :   **[5]**

    •   The student has excellent academic record but is financially weak.

             OR

    •   The student does not have an excellent academic record and belongs to a backward class.

             OR

    •   The student does not have an excellent academic record and is physically impaired.

The inputs are:

INPUTS	
A	Has excellent academic record
F	Financially sound
C	Belongs to a backward class
I	Is physically impaired

    (In all the above cases 1 indicates yes and 0 indicates no).

Output :   **X** [1 indicates yes, 0 indicates no for all cases]
Draw the truth table for the inputs and outputs given above and write the **SOP** expression for **X(A,F,C,I)**.

1218-868A   **Turn over**

(b) Using the truth table, state whether the following proposition is a *tautology*, *contingency* or a *contradiction*: **[3]**

$$\sim (A \wedge B) \vee (\sim A \Rightarrow B)$$

(c) Simplify the following expression, using Boolean laws: **[2]**

$$A \cdot (A' + B) \cdot C \cdot (A + B)$$

Question 6

(a) What is an *Encoder?* Draw the Encoder circuit to convert A-F hexadecimal numbers to binary. State an application of a Multiplexer. **[5]**

(b) Differentiate between *Half Adder* and *Full Adder*. Draw the logic circuit diagram for a Full Adder. **[3]**

(c) Using only NAND gates, draw the logic circuit diagram for $A' + B$. **[2]**

SECTION – B
*Answer **any two** questions.*
Each program should be written in such a way that it clearly depicts the logic of the problem.
This can be achieved by using mnemonic names and comments in the program.
(Flowcharts and Algorithms are **not** required.)
The programs must be written in Java.

Question 7

Design a class **Perfect** to check if a given number is a perfect number or not. [A number is said to be perfect if sum of the factors of the number excluding itself is equal to the original number] **[10]**

Example : $6 = 1 + 2 + 3$ (where 1, 2 and 3 are factors of 6, excluding itself)

Some of the members of the class are given below:

Class name : **Perfect**

Data members/instance variables:

    num : to store the number

Methods/Member functions:

Perfect (int nn)	:	parameterized constructor to initialize the data member num=nn
int sum_of_factors(int i)	:	returns the sum of the factors of the number(num), excluding itself, using **recursive technique**
void check( )	:	checks whether the given number is perfect by invoking the function *sum_of_factors( )* and displays the result with an appropriate message

Specify the class **Perfect** giving details of the **constructor( )**, **int sum_of_factors(int)** and **void check( ).** Define a **main( )** function to create an object and call the functions accordingly to enable the task.

Question 8

Two matrices are said to be equal if they have the same dimension and their corresponding **[10]** elements are equal.

For example the two matrices A and B given below are equal:

Matrix A

1	2	3
2	4	5
3	5	6

Matrix B

1	2	3
2	4	5
3	5	6

Design a class **EqMat** to check if two matrices are equal or not. Assume that the two matrices have the same dimension.

Some of the members of the class are given below:

Class name	:	**EqMat**

Data members/instance variables:

a[ ][ ]	:	to store integer elements
m	:	to store the number of rows
n	:	to store the number of columns

Member functions/methods:

EqMat(int mm, int nn)	:	parameterised constructor to initialise the data members m = mm and n = nn
void readarray( )	:	to enter elements in the array
int check(EqMat P, EqMat Q)	:	checks if the parameterized objects P and Q are equal and returns 1 if true, otherwise returns 0
void print( )	:	displays the array elements

Define the class **EqMat** giving details of the **constructor( ), void readarray( ), int check(EqMat, EqMat)** and **void print( )**. Define the **main( )** function to create objects and call the functions accordingly to enable the task.

Question 9

A class **Capital** has been defined to check whether a sentence has words beginning with a capital letter or not. **[10]**

Some of the members of the class are given below:

Class name   :   **Capital**

Data member/instance variable:

    sent   :   to store a sentence

    freq   :   stores the frequency of words beginning with a
                                                capital letter

Member functions/methods:

    Capital( )   :   default constructor

    void input( )   :   to accept the sentence

    boolean isCap(String w)   :   checks and returns true if word begins with a
                                                capital letter, otherwise returns false

    void display( )   :   displays the sentence along with the frequency
                                                of the words beginning with a capital letter

Specify the class **Capital**, giving the details of the **constructor( )**, **void input( )**, **boolean isCap(String)** and **void display( )**. Define the **main( )** function to create an object and call the functions accordingly to enable the task.

6

1218-868A

SECTION – C

*Answer **any two** questions.*

Each program should be written in such a way that it clearly depicts the logic of the problem stepwise.

This can be achieved by using comments in the program and mnemonic names or pseudo codes for algorithms. The programs must be written in Java and the algorithms must be written in general / standard form, wherever required / specified.

(Flowcharts are **not** required.)

Question 10

A super class **Number** is defined to calculate the factorial of a number. Define a sub class **[5]**
Series to find the sum of the series S = 1! + 2! + 3! + 4!+………......+n!

The details of the members of both the classes are given below:

Class name	:	**Number**
Data member/instance variable:		
n	:	to store an integer number
Member functions/methods:		
Number(int nn)	:	parameterized constructor to initialize the data member n=nn
int factorial(int a)	:	returns the factorial of a number (factorial of n = 1×2×3×………×n)
void display( )	:	displays the data members
Class name:		**Series**
Data member/instance variable:		
sum	:	to store the sum of the series
Member functions/methods:		
Series(…)	:	parameterized constructor to initialize the data members of both the classes
void calsum( )	:	calculates the sum of the given series
void display( )	:	displays the data members of both the classes

*Assume that the super class **Number** has been defined*. Using the **concept of inheritance**, specify the class **Series** giving the details of the **constructor(…)**,**void calsum( )** and **void display( )**.

The super class, main function and algorithm need NOT be written.

1218-868A **Turn over**

Question 11

Register is an entity which can hold a maximum of 100 names. The register enables the user to add and remove names from the top most end only.

Define a class **Register** with the following details:

Class name	:	**Register**

Data members / instance variables:

stud[ ]	:	array to store the names of the students
cap	:	stores the maximum capacity of the array
top	:	to point the index of the top end

Member functions:

Register (int max)	:	constructor to initialize the data member cap = max, top = −1 and create the string array
void push(String n)	:	to add names in the register at the top location if possible, otherwise display the message "OVERFLOW"
String pop( )	:	removes and returns the names from the top most location of the register if any, else returns "$$"
void display( )	:	displays all the names in the register

(a) Specify the class **Register** giving details of the functions **void push(String)** and **String pop( ).** Assume that the other functions have been defined. **[4]**

The main function and algorithm need NOT be written.

(b) Name the entity used in the above data structure arrangement. **[1]**

Question 12

(a) A linked list is formed from the objects of the class **Node**. The class structure of the Node is given below: **[2]**
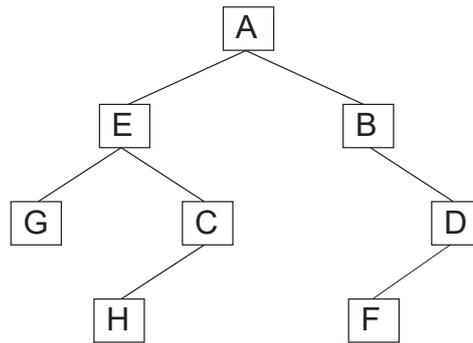
```
class   Node
{
        int n;
        Node   link;
}
```

Write an *Algorithm* **OR** a *Method* to search for a number from an existing linked list.

The method declaration is as follows:

void FindNode(Node str, int b)

(b)   Answer the following questions from the diagram of a Binary Tree given below:

(i)   Write the inorder traversal of the above tree structure.   **[1]**

(ii)   State the height of the tree, if the root is at level **0** (zero).   **[1]**

(iii) List the leaf nodes of the tree.   **[1]**

1218-868A