MARTHOMA RESIDENTIAL SCHOOL
First Model Examination - Dec 2018

Class-XII

Computer Science

Time: 3 hrs
Marks: 70

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 Distributive law and prove it with the help of a truth table. [1]

(b) Draw the truth table to prove the propositional logic expression. [1]

$(X \Rightarrow Y) \wedge (Y \Rightarrow X) = X \Leftrightarrow Y$

(c) Find the dual for the Boolean equation: $A + A' B = A + B$ [1]

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

$F(X, Y, Z) = (X+Y').(Y'+Z)$

(e) Minimize: $F = XY + (XZ)' + XY'Z$ using Boolean laws. [1]

Question 2

(a) What do you mean by an Exception. What is the use of finally in try- catch block? [2]

(b) What is a stack? What are the 2 operations performed on a stack? [2]

(c) Each element of an array arr[15][20] requires 'W' bytes of storage. If the address of arr[6][8] is 4440 and the Base Address at arr[1][1] is 4000 , find the width 'W' of each cell in the array arr[ ][ ] when the array is stored as Column Major Wise. [2]

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

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

(e) Define Dominant term. What is the Big O complexity of binary search method. [2]

Question 3

(a) The given method Demo( ) is a part of some class. Answer the following questions with dr working. [5]

```
void Demo( int Y)
{
if(Y>0)
{    Demo( Y/2);
     System.out.print(Y%2);
}
}
```

i) What will the above function prints when invoked as Demo(13)?

ii) What will the above function prints when invoked as Demo(7)?

iii) State in one line that what the function Demo(...) do apart from recursion?

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)** = ∑(0, 2, 3, 6, 8, 10, 11, 14, 15)

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

(ii) Draw the logic gate diagram for the reduced expression. Assume that the variables and t complements are available as inputs.

(b) Given the Boolean function **F(P, Q, R, S)** = **Π**(5, 7, 8, 10, 12, 14, 15)

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

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

Question 5

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

- 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
B	-	Belongs to a backward class
P	-	Is physically impaired

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

Output: S[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 S(A, F, B

(i) Simplify the following Boolean expression and draw the gate for the reduced expression :
F= A'B + AB'C + A [3]

(ii) Draw the logic diagram of 4:1 Multiplexer. [2]

Question 6

(a) Define Universal gates. Give one example and show how it works as an AND gate. [2]

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

(c) Draw a truth table with a 3 input combination which outputs 1 if there are odd number of 0's. Also derive an **SOP** expression for the output. Reduce the expression using Karnaugh Map. [5]

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 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

[A composite number is a number with more than 2 factors]

A class Composite contains a two dimensional array of order [m x n]. The maximum value possible for both 'm' and 'n' is 20. Design a class Composite to fill the array with the first (m x n) composite numbers in column wise. The details of the members of the class are given below: [10]

Class name : Composite

Data members/instance variables:

arr[ ] [ ] : stores the composite numbers column wise
m : integer to store the number of rows
n : integer to store the number of columns

Member functions/methods:

Composite(int mm, int nn) : to initialize the size of the matrix m=mm and n=nn
int isComposite(int p) : returns 1 if number is composite otherwise returns 0
void fill ( ) : to fill the elements of the array with the first (m × n) composite numbers in column wise

void display( ) : displays the array in a matrix form

Specify the class Composite giving details of the constructor (int,int), int isComposite (int), void fill( ) and void display( ). Define a main( ) function to create an object and call the functions accordingly to enable the task.

Question 8

A class **Admission** contains the admission numbers of 50 students. Some of the data members / member functions are given below: [10]

Class name : Admission

Data member/instance variable:

Adno[ ] : integer array to store admission numbers

Member functions/methods:

Admission( ) : constructor to initialize the array elements

void fillArray( ) : to accept the elements of the array in ascending order

int binSearch(int l, int u, int v) : to search for a particular admission number (v) using **binary search and recursive technique** and returns **1** if found otherwise returns **-1[** u is upper bound and l is lower bound of the array]

Specify the class Admission giving details of the constructor, void fillArray( ) and int binSearch(int, int, int) . Define the main( ) function to create an object and call the functions accordingly to enable the task.

Question 9

A class **Merger** has been defined to merge two sorted integer arrays in ascending order. Some of the members of the class are given below: [10]

Class name : Merger

Data members/instance variables:

int a[ ] : to store the elements of an array

int n : to store the size of the array

Member functions:

Merger (int nn) : constructor to assign n=nn

void accept( ) : to accept the elements of the array in ascending order(assume no duplicates are entered)

Merger mix(Merger M) : to merge the current object array elements with the parameterized array elements and return the resultant object.

void display( ) : to display the elements of the array

Specify the class **Merger**, giving details of the constructor(int), void accept( ), Merger mix(Merger M) and void display( ). Define the main( ) function to create an object and call the function accordingly to enable the task.

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 also be achieved by using comments in the program and mnemonic names or pseudocodes for algorithms. The program 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 Stock has been defined to store the details of the stock of a retail store. Define a subclass Purchase to store the details of the items purchased with the new rate and updates the stock. Some of the members of the classes are given below: [5]

Class name : Stock

Data members/instance variables:

item : to store the name of the item

qty : to store the quantity of an item in stock

rate : to store the unit price of an item

amt : to store the net value of the item in stock

Member functions:

Stock (...) : parameterized constructor to assign values to the data members

void display( ) : to display the stock details

Class name : Purchase

Data members/instance variables:

pqty : to store the purchased quantity

prate : to store the unit price of the purchased item

Member functions / methods

Purchase(...) : parameterized constructor to assign values to the data members of both classes

void update ( ) : to update stock by adding the previous quantity by the purchased quantity and replace the rate of the item if there is a difference in the purchase rate.Also update the current stock value as: (quantity * unit price)

void display( ): to display the stock details before and after updating.

Assume that the super class Stock has been defined. Using the **concept of Inheritance,** specify the class **Purchase** giving details of the **constructor(...), void update( )** and **void display( ).The super class and the main function need not be written.**

Question 11

A bookshelf is designed to store the books in a stack with LIFO(Last In First Out) operation. Define a class Book with the following specifications: [5]

Class name : Book

Data members/instance variables:

name[ ] : stores the names of the books

point : stores the index of the topmost book

max : stores the maximum capacity of the bookshelf

Methods/Member functions:

Book(int cap) : constructor to initialise the data members max = cap and point = −1

void tell( ) : displays the name of the book which was last entered in the shelf. If there is no book left in the shelf, displays the message "SHELF EMPTY"

void add(String v) : adds the name of the book to the shelf if possible, otherwise displays the message "SHELF FULL"

void display( ) : displays all the names of the books available in the shelf

Specify the class Book giving the details of **ONLY** the functions void tell( ) and void add(String). Assume that the other functions have been defined.

The main function need not be written.

Question 12

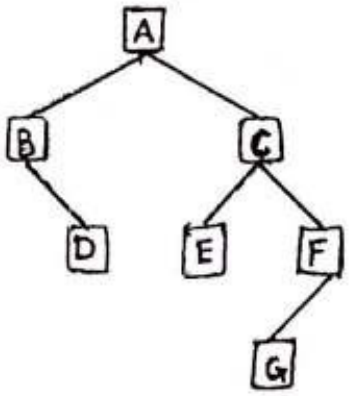
(a) A linked list is formed from the objects of the class **Node**. The class structure of the Node is given below:

```
class Node
{
 int n;
 Node next;
}
```

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 Start, int b) [2]

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



(a) Write the post order traversal of the tree.

(b) State the root and the height of the tree

(c) Separate the Internal nodes and the External nodes of the tree. [3]