MARTHOMA RESIDENTIAL SCHOOL
Second Model Examination - Jan 2020

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

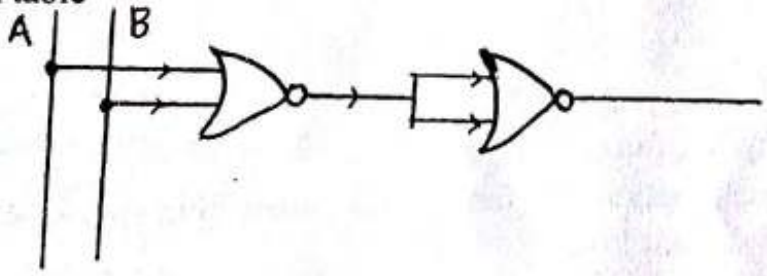
b) Verify if the following proposition is true or false using truth table

A => B = B' => A' [1]

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

d) Find **F'** if F= b.a+a'c +c' [1]

e) Name the logic gate obtained from the following circuit diagram and write its truth table [1]



Question 2

a) What is the use of finally keyword? What is the difference between error and exception? [2]

b) Convert the following infix notation to postfix notation [2]

(m + n / f) + q – r * k + s

c) Define the terms 1) multiple inheritance 2) this keyword [2]

d) Each element of an array **A[-5, ...15 , 10,.....35]** requires 4 bytes of storage. If the Base Address is 2500, determine address of A[15 , 30] if the array is stored in RowMajor Order.

[2]

e) Evaluate the Big O Complexity for the following code:.

```
int z=1;
while(z<=N)
{
z++;
}
for(int x=1 ; x<=N ; x++)
 {
 for  (int y=1; y<=N y++)
  {     System.out.println("Helloworld");
     }
      }
```

[2]

Question 3

The following is a function of some class. Answer the given questions with dry run / working.

```
void try_fun (int num,int d)
{
if (num>1)
{
  if(num%d==0)
  {
  System.out.print(d+",");
  try_fun(num/d,d);
  }
else
try_fun(num,d+1);
}
}
```

i) What will be the output of try_fun (20,2)?

ii) What will be returned by try_fun (13,2)?

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

[5]

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(p, q, r, s)** = $\sum$(0,2,3,4,5,11,12,13)

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 using **NAND** gates only. Assume that the variables and their complements are available as inputs. [1]

b) Given the Boolean function **F(P, Q, R, S)= Π(1,6,7,8,9,10,11,14,15)**

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 using **NOR** gates only. Assume that the variables and their complements are available as inputs. [1]

Question 5

a) The National College of Journalism is offering courses in three different categories of journalism, which are the print, the web and the broadcasting media. A student is eligible to apply, if he/she satisfies any one of the following criteria

Student is a graduate in any discipline with an aggregate of 75% or above and with a record of Literary skills.

OR

Student is a graduate in Mass Communication with an aggregate of 75% or above

INPUTS

S: Graduate in any discipline

P:.Graduate in Mass Communication

C: Aggregate of 75% or above

T: Record of literary skills

Output **A** denotes eligible to apply [1 indicates YES ad 0 indicates NO in all cases]

Draw the truth table for the inputs and outputs given above and write the **SOP** expression for **A(S,P,C,T)** [5]

b)

(i) Convert the expression: F=P.Q+Q.R' into canonical SOP form [3]

(ii) Write the contra positive and inverse form of the conditional proposition **p => q** and also draw its truth table. [2]

Question 6

a) Define multiplexer. Draw the logic diagram and truth table of of 4 x 1 multiplexer [4]

b) Simplify the following Boolean expression using Boolean laws. Mention the laws at each step. **A. [(B +C).(AB+AC)']** [2]

c) Draw the logic circuit of a Full Adder and write the Boolean expression for SUM and CARRY for half adder . [4]

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

A class **Exchange** has been defined to accept a sentence in which each word is separated by single blank space terminated by full stop(.) and forms a word containing all the consonants followed by all the vowels and full stop. For eg. Input: It is a pen.

Output: tspnIiae.

Some of the members of the class are given below:

Class name: **Exchange**

Data member/instance variable:

sent: to store a sentence

newStr: to store the new word

size : integer to store the length of the sentence

Member functions/methods:

Exchange ( ): default constructor to initialise values to all the instance variables

void readsentence( ): to accept the sentence terminated by full stop.

void NewWord( ): forms a word containing all the consonants followed by all the vowels present in the sentence 'sent' and stores in 'newStr'. The word must be terminated by full stop.

void display ( ) : displays the original sentence and the word with suitable message.

Specify the class **Exchange** giving the details of the **constructor( ), void readsentence( ), void NewWord( )** and **void display( ).** Define the **main( )** function to create an object and call the functions accordingly to enable the task.

[10]

Question 8

A class **Finder** defines a recursive function to find total uppercase and lowercase letters from a string.

Some of the members of the class are given below:

Class name: **Finder**

Data member/instance variable:

st: to store a sentence

up, sm: integers to store total uppercase and total lowercase letters respectively.

len : integer to store the length of the sentence

Member functions/methods:

Finder ( ): default constructor to initialise values to all the instance variables

void readStr( ): to accept the string.

void RecLetters(int L): find and store total number of capital and total number of small letters from st and store in up, sm respectively using **recursive technique.**

void Print( ) : displays the original sentence and by invoking the recursive function, display, the total number of capital and total number of small letters stored in up, sm respectively with suitable message.

Specify the class **Finder** giving the details of the **constructor( ), void readStr( ), void RecLetters(int L),** and **void Print( )** . Define the **main( )** function to create an object and call the functions accordingly to enable the task.

[10]

Question 9

A class **Ascending** contains integers that are already arranged in **ascending order** Some of the members of the class are given below:

Class name: **Ascending**

Data members/instance variables:

arr[ ] : an array of integers sorted in ascending order

n: number of integers in array

Member functions/methods:

Ascending (int nn): constructor to assign n=nn and create a ascending list of size n

void inputList( ) : input integers in the array arr[] that are already sorted in the ascending order.

Ascending Merge(**Ascending** mat): to merge the **Ascending** object **mat** with the current **Ascending** object and return a third **Ascending** list object which is also sorted in **ascending** order.

void displayList( ) : displays the list of sorted integers

Define the class **Ascending** giving details of the constructor( ), **void inputList( ), Ascending Merge(Ascending mat):** and **void displayList( ).** Define the **main( )** function to create objects and call the functions accordingly to enable the task.

[10]

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 **Company** has been defined to store the details of a company and a sub class **Sales** enables the computation of purchase of products..The details of both the classes are given below:

Class name : Company

Data members/instance variables:

name: stores the name of the company

code: stores the code number of the product

total: stores the total number of units of the product in the stock.

Member functions/methods:

Company(.....) : parameterized constructor to assign values to the instance variables

void show( ): displays all the data members

Class name : Sales

Data members/instance variables:

price: stores the price of the product in decimals

units: stores the total number of units to be purchased.

amt: stores the amount to be paid in decimals

net_amt: stores the final amount to be paid

Member functions / methods

Sales(....): parameterized constructor to assign data members of both the classes and amt=0.0

void compute( ) : calculates the total amount(amt=units * price) , if the total units of the product to be purchased are available in the stock. Also a discount of **15%** will be given if total amount of purchase is more than 1 lakh. Calculate the net amount to be paid by adding **4.5%** on the amount after the discount. Display a message **"NOT AVAILABLE"**, if units to be purchased are more than the stock. Also update the stock after removing the units sold from the total.

void show( ): displays the details of the company net amount to be paid and current stock available.

Assume that the super class Company has been defined. Using the concept of inheritance, specify the class **Sales** giving details of the **constructor**, functions **void compute( )** and **void show( ).**

The super class, main function and algorithm need NOT be written. [5]

Question 11

Strange is an entity which can hold at the most 20 integers. The strange restriction is that an integer can only be added from the top or removed from the top. This is like a pile of

china plates where we can add a plate to the pile from the top and remove a plate from the top only. Define the class with following details:

Class name: **Strange**

Data member/instance variable:

arr[ ] : integer array to hold the integer elements

cap : integer to store the maximum capacity of the array

top : integer to point the index of the topmost element.

Member functions/methods:

Strange (int max) : constructor to initialize the data member **cap = max, top= -1** and create the integer array

void pushItem(int value) : to add integer value to the top of the strange , if possible, else display the message "**strang is full cannot push item**"

int popItem( ) : to remove integer value from the top of the strange and returns if strange is not empty, else, outputs a message **"strange is empty"** and returns **-1111**

Specify the class **Strange** giving the details of **void pushItem(int value) and int popItem( ) . The main function need not be written.**

[5]

Question 12

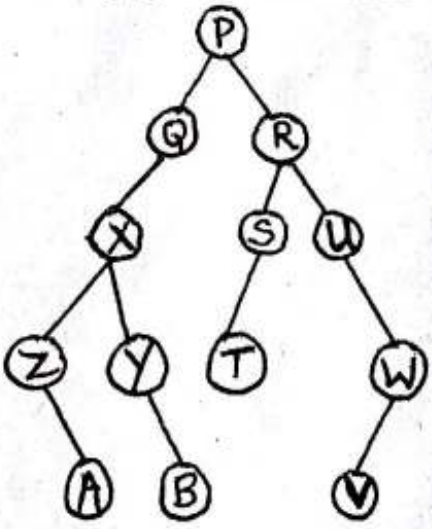
(a) A linked list is formed from the objects of the class **Node**. The class structure of the Node is given below: [2]

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

Write an ***Algorithm*** **OR** a ***Method*** to find and print the product of composite integers only from the linked list. The method declaration is as follows: void CompProd(Node S)

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

[3]



(a) Write the preorder traversal of the tree.

(b) State the internal nodes of the tree and the height of the tree.

(c)List the pair of siblings and number of pair of siblings.