(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
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

(i) A + B = B + A;   [1]

AB = BA

represent which laws?
(a) Commutative
(b) Associative
(c) Distributive
(d) Idempotence

(ii) (A + B) + C = A + (B + C);   [1]

(AB)C = A(BC)

represent which laws?
(a) Commutative
(b) Associative
(c) Distributive
(d) Idempotence

(iii) A(B + C) =AB + AC;   [1]

A + BC = (A + B)(A + C)

represent which laws?
(a) Commutative
(b) Associative
(c) Distributive
(d) Idempotence

(iv) A + AB = A;   [1]

A(A + B) = A

represent which laws?

(a) Commutative

(b) Associative

(c) Absorption

(d) Idempotence

(v) Predict the output of the following [1]

boolean p;

p = ("BLUEJ".length() > "bluej".length()) ? true: false;

(a) true

(b) false

(c) 0

(d) 1

(vi) What is the difference between String and StringBuffer? [1]

(vii) What is the difference between charAt( ) and indexOf( ) ? [1]

(viii) Write disadvantages of an array. [1]

(ix) Construct the truth table for the expression (A . (A + B)). [1]

(x) What are the individual's right to privacy? [1]

Question 2

(i) Write the output and step value(dry run) for the following code: [3]

```
class dk3
{
public static void main(String args[])
{
  int m=2,n=15;
  for(int i=1;i<=5;i++)
  {
    m++;--n;
   System.out.println("m="+m);
   System.out.println("n="+n);
  }
} }
```

(ii) What is augmented reality? [1]

(iii) How is Augmented Reality different from Virtual Reality? [1]

(iv) Explain append( ) and insert( ) with example. [2]

(v) The following function Check( ) is a part of some class. What will the function Check( ) return when the values of both 'm' and 'n' are equal to 5? Show the dry run / working. [3]

```
int Check (int m, int n)
{
if (n = = 1)
return – m – –;
else
return + + m + Check (m, ––n);
}
```

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 3

(i) Convert the given number to their binary equivalents: $(165.35)_{10}$ **[1]**

(ii) Convert binary to decimal number: $(1000011)_2$ to $(\ )_{10}$ **[1]**

(iii) Convert octal to binary number: $(345.33)_8$ to $(\ )_2$ **[1]**

(iv) Convert to Hexa-decimal: $(110011101111)_2$ **[1]**

(v) Show that $X \vee \sim (Y \wedge X)$ is a tautology. **[2]**

(vi) Verify the following proposition with help of the truth table: **[2]**
$P \vee (\sim P \wedge Q) = P \vee Q$

(vii) Design a circuit to realize the following: $F(a,b,c) = AB + AC' + B'A'C$ **[2]**

Question 4

(i) Prove the following relation: $(P \wedge Q) \vee (P \wedge \sim Q) = P$ **[2]**

(ii) If $(\sim P \Rightarrow Q)$ then write its : Inverse & Converse. **[2]**

(iii) Construct the truth tables for the following: **[2]**
$(p \Rightarrow q) \wedge (q \Rightarrow p)$

(iv) Write advantages of using functions in programs. **[2]**

(v) Write a short note on significance of constructors in OOP classes. **[2]**

Question 5

(i) Compare iteration and recursion. **[2]**

(ii) What is Internet of Things? **[1]**

(iii) Choose the correct output of Java program with arrays **[2]**
```
public class Polo
{
public static void main(String args[])
```

```
{
String[] computer = {"RAM","HDD","MOUSE"};
String[] parts = {computer[0],computer[2]};
System.out.print(parts[1]);
} }
```

(iv) Derive the expression for a half Adder with Truth table and Logic diagram. **[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 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 6

Write a program to declare a square matrix A[][] of order M*M where 'M' is the **[10]**
number of rows and the number of columns , such that M must be greater than 2
and less than 10.Accept the value of M as user input. Display an appropriate
message for an invalid input. Allow the user to input integers into this matrix.
Perform the following task.

a)Display the original matrix.

b)Rotate the matrix 90 degree clockwise as shown below:

Original matrix   Rotated matrix

1   2   3   7   4   1

4   5   6   8   5   2

7   8   9   9   6   3

c)Find the sum of the elements of the four corners of the matrix.

Test your program for the following data and some random data:

Sample data:-

Original matrix

3   4   9

2   5   8

1   6   7

Matrix after rotation

1   2   3

6   5   4

7   8   9

Sum of the corner elements =20

Question 7

Caesar Cipher is an encryption technique which is implemented as ROT-13(Rotated by 13 places). It is a simple letter substitution Cipher that replaces a letter with the letter 13 places after it in the alphabets, with other characters remaining unchanged.

[10]

Write a program to accept a plain text of length l, where l must be >3&&<100. Encrypt the text if valid as per Caesar Cipher.

Sample Input: Hello! How are you?

Sample Output: Uryyb! Ujbnerlbh?

Question 8

(i) Write a program to sort an array using Bubble sort method. **[5]**

(ii) Write a program to search an element in an array using Linear Search. **[5]**

SECTION – C
Answer any **two** questions.

Question 9

Write a Java program to input a digit and print it in words using a function. **[5]**

Question 10

Write a java program to find whether a given number is odd or even or prime using a function. **[5]**

Question 11

Write a program to sort an array using Selection sort method. **[5]**

✶✶✶✶✶