This Paper consists of three problems from which candidates are required to attempt any one problem.

Candidates are expected to do the following:

1. Write an algorithm for the selected problem. [3]
   (Algorithm should be expressed clearly using any standard scheme such as pseudo code or in steps which are simple enough to be obviously computable.)

2. Write a program in JAVA language. The program should follow the algorithm and should be logically and syntactically correct. [5]

3. Document the program using mnemonic names / comments, identifying and clearly describing the choice of data types and meaning of variables. [2]

4. Code / Type the program on the computer and get a printout (hard copy). Typically, this should be a program that compiles and runs correctly. [2]

5. Test run the program on the computer using the given sample data and get a printout of the output in the format specified in the problem. [5]

6. Viva-Voce on the Selected Problem. [3]

In addition to the above, the practical file of the candidate containing the practical work related to programming assignments done during the year is to be evaluated as follows:

- Programming assignments done throughout the year (by the teacher) [5]
- Programming assignments done throughout the year (by the Visiting Examiner) [5]
Solve any one of the following Problems.

Question 1
A Goldbach number is a positive even integer that can be expressed as the sum of two odd primes.

Note: All even integer numbers greater than 4 are Goldbach numbers.

Example:  
6 = 3 + 3  
10 = 3 + 7  
10 = 5 + 5

Hence, 6 has one odd prime pair 3 and 3. Similarly, 10 has two odd prime pairs, i.e. 3 and 7, 5 and 5.

Write a program to accept an even integer ‘N’ where N > 9 and N < 50. Find all the odd prime pairs whose sum is equal to the number ‘N’.

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

Example 1:  
INPUT: N = 14  
OUTPUT: PRIME PAIRS ARE: 3, 11  
7, 7

Example 2:  
INPUT: N = 30  
OUTPUT: PRIME PAIRS ARE: 7, 23  
11, 19  
13, 17

Example 3:  
INPUT: N = 17  
OUTPUT: INVALID INPUT. NUMBER IS ODD.

Example 4:  
INPUT: N = 126  
OUTPUT: INVALID INPUT. NUMBER OUT OF RANGE.
Question 2

Write a program to declare a matrix A[ ][ ] of order (M×N) where ‘M’ is the number of rows and ‘N’ is the number of columns such that the values of both ‘M’ and ‘N’ must be greater than 2 and less than 10. Allow the user to input integers into this matrix. Perform the following tasks on the matrix:

(a) Display the original matrix.
(b) Sort each row of the matrix in ascending order using any standard sorting technique.
(c) Display the changed matrix after sorting each row.

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

Example 1:

INPUT: 
M = 4
N = 3

ENTER ELEMENTS OF MATRIX
11 -2 3
5 16 7
9 0 4
3 1 8

OUTPUT:

ORIGINAL MATRIX
11 -2 3
5 16 7
9 0 4
3 1 8

MATRIX AFTER SORTING ROWS
-2 3 11
5 7 16
0 4 9
1 3 8

Example 2:

INPUT: 
M = 3
N = 3

ENTER ELEMENTS OF MATRIX
22 5 19
7 36 12
9 13 6
OUTPUT:

ORIGINAL MATRIX

22  5  19
7  36 12
9  13  6

MATRIX AFTER SORTING ROWS

5   19  22
7   12  36
6   9  13

Example 3:

INPUT:  M = 11
        N = 5

OUTPUT:

MATRIX SIZE OUT OF RANGE

Question 3

The names of the teams participating in a competition should be displayed on a
banner vertically, to accommodate as many teams as possible in a single banner.
Design a program to accept the names of N teams, where 2 < N < 9 and display
them in vertical order, side by side with a horizontal tab (i.e. eight spaces).
Test your program for the following data and some random data:

Example 1:

INPUT:  N = 3
        Team 1: Emus
        Team 2: Road Rols
        Team 3: Coyote

OUTPUT:

E        R        C
m          o        o
u          a        y
s          d        o
t
R        e
o        l
s
Example 2:

INPUT:  
N = 4  
Team 1: Royal  
Team 2: Mars  
Team 3: De Rose  
Team 4: Kings  

OUTPUT:  
R M D K  
o a e i  
y r n  
a s R g  
l o s  
s e  

Example 3:

INPUT:  
N = 10  

OUTPUT:  INVALID INPUT