***Computer Lab Report***

******

**Name : Indraneel Pal**

**Class : B.e.e(1st year)**

**Department of electrical Engineering,**

**Jadavpur University**

**Roll no : 001610801142**

***Introduction***

Today we cannot live without computers. Computers are widely used in different scientific areas all over the world. A computer can perform arithmetic operations and take logical decisions at very high speed. But to implement this brain behind computers requires a lot of programming. A program can be written in many different languages. In modern computer systems the programs are written in more convenient languages which are broadly classified into two groups: (i) assembly language and (ii) machine language.

In assembly language a program is written in symbolic codes called mnemonics, rather than binary codes which are used in machine language. In most of the cases there exists a one to one correspondence between assembly language and machine language, so these two languages are similar. Like the machine language assembly language is also machine dependent.

Most of the computer programs are written in a language whose instruction set is more compatible with human languages and human thought processes. This language is known as HIGH LEVEL LANGUAGE. High level language has three significant advantages over the machine language viz. simplicity, uniformity and portability , i.e., machine independence.

A program written in a high level or assembly language must, however be translated into machine language before its execution. A program written in a high level language or assembly language is called source program. The source program when translated to a machine language is known as object program. The assembly language program is translated by assembler and high level language is translated by a compiler or interpreter. Every high level language is translated by a compiler or interpreter.

***History Of Fortran***

The first high level programming language developed was FORTRAN (Formula Translation). It was developed at IBM by John Backus and his team (**1954-1957**) and was called FORTRAN I. FORTRAN was originally developed for manipulation of numerical data. Since then, FORTRAN has attracted wide attention and has undergone a number of changes and each new version is made with a few changes in basic statements and includes some additional features. The two major versions are FORTRAN-II and FORTRAN-IV. The FORTRAN was standardized in **1966** by the American National Standards Institute (ANSI). After standardization a FORTRAN program written for one computer could be run on another computer without any modifications and then its applications evolved from strictly numerically oriented to move general applications involving character and file manipulations. A new FORTRAN standard (ANSIX**3.9-1978)** known as FORTRAN **77**, was approved in 1977. Several new features were included in this version to make it more structured. The Latest version of FORTRAN is known as FORTRAN 90/95. It includes graphics , structured and pointer variables and other features.

This Report explains why we can do great things in FORTRAN

***Contents***

**This Report contains**

**FORTRAN PROGRAMS on**

**Ten topics:**

|  |
| --- |
| 1)To find the average of N numbers |
| **2)To find the maximum of N numbers** |
| **3)To find the minimum of N numbers** |
| **4)To calculate SIN(X) upto first three terms and comparing with the actual result** |
| **5)To calculate LOG(X) upto first three terms and comparing with the actual result** |
| **6)To sort an array of N numbers in ascending and descending order** |
| **7)To find the roots of a quadratic equation** |
| **8)To find the Factorial of a number** |
| **9) To find NCR and NPR of a Number** |
| **10)To find the value in a B-H curve** |

**Program-1**

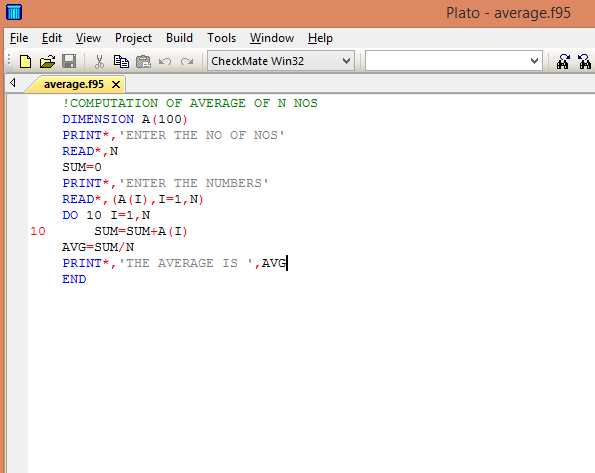
Q: Write a Program in FORTRAN to calculate the average of N numbers

Assumptions: No such assumptions were taken

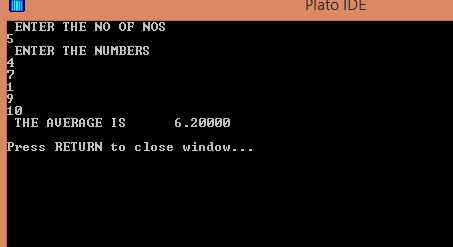
Constraints: N<**100**

Algorithm:

**Program:**

****

**Output:**

****

**Result:** The Final Output **6.2** matches with the expected result.

**Program-2**

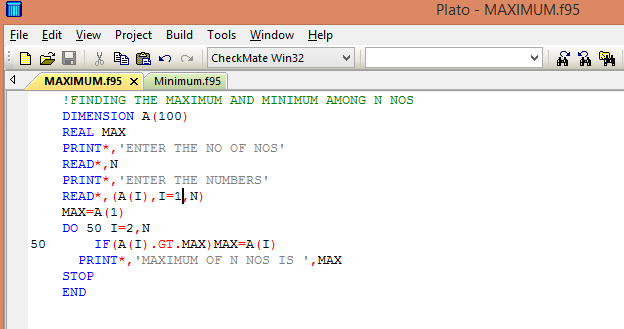
Q: Write a Program in FORTRAN to find the maximum of N numbers

Assumptions: No such assumptions were taken.

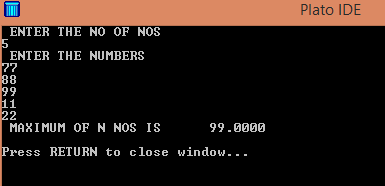
Constraints: N<**100**

Algorithm:

Program:



Output:



Result:

Out of the above array of five numbers **99** is given as the maximum number which is true.

**Program-3**

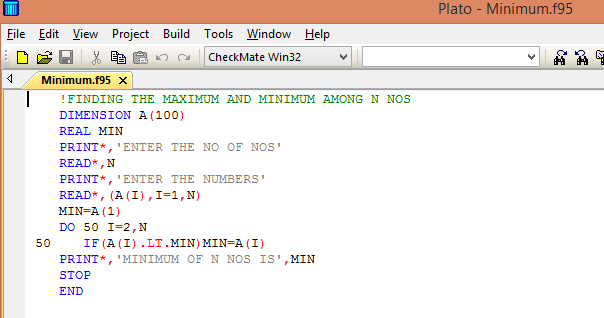
Q: Write a Program in FORTRAN to find the minimum of N numbers

Assumptions: No such assumptions were taken.

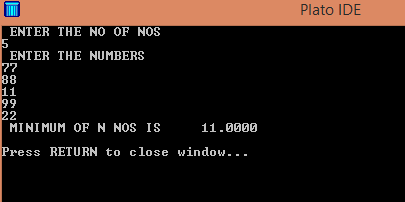
Constraints: N<**100**

Algorithm:

Program:



Output



Result:

Out of the above array of five numbers **11** is given as the minimum number which is true.

**Program-4**

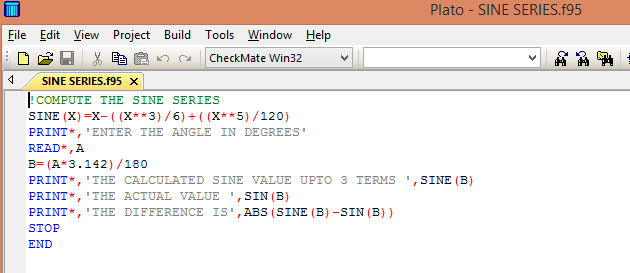
Q: Write a program in FORTRAN to calculate the value of SIN(X) upto the first three terms and compare it with the actual result

Assumptions: Answer might be small, expressed in exponential order.

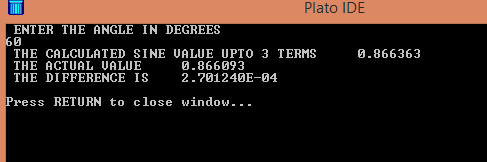
.Constraints: No such constraints are required.

Algorithm:

Program:



Output:



Result: The final output for angle 60 degrees i.e. the difference between the calculated value and actual value is very small, so the accuracy of the program is very high.

**Program-5**

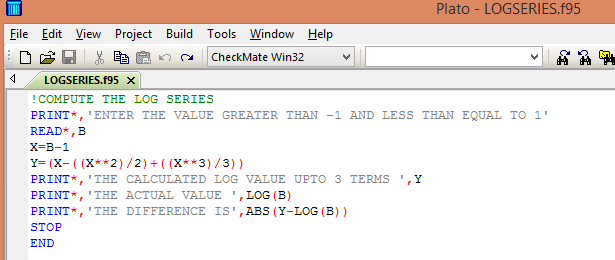
Q: Write a program in FORTRAN to calculate the value of LOG(X) upto the first three terms and compare it with the actual result

Assumptions: Answer might be small, expressed in exponential order.

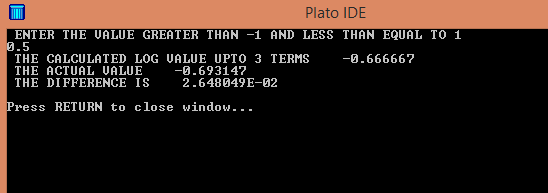
Constraints: X>0 and <=1

Algorithm:

Program:



Output:



Result: The final output for X=0.5 i.e. the difference between the calculated value and actual value is very small, so the accuracy of the program is very high.

**Program-6**

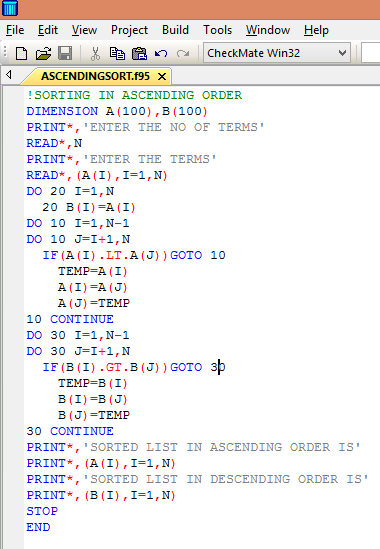
Q: Write a program in FORTRAN to sort an array of N elements in ascending and descending order

Assumptions: No such assumptions are made.

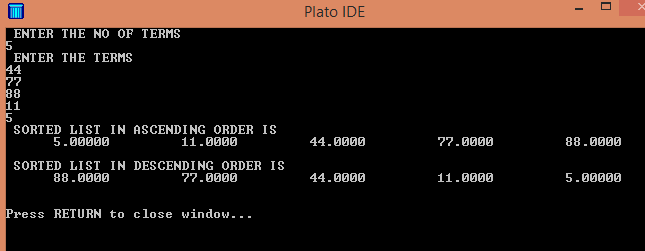
Constraints : N<100

Algorithm:

Program:



Output



Result: The output is as expected.

**Program-7**

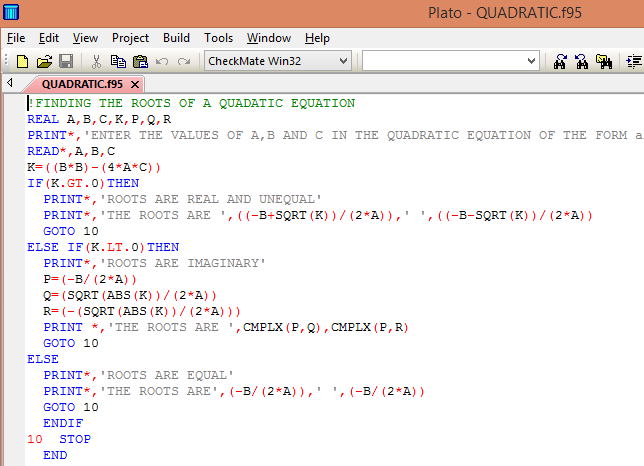
Q: Write a program in FORTRAN to calculate the roots of a quadratic equation of the form ax2+bx+c.

Assumptions: No such assumptions are required.

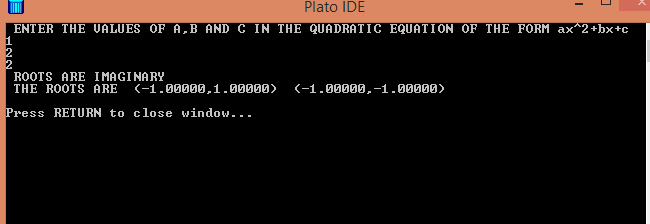
Constraints: No such constraints are required.

Algorithm

Program:



Output



Result : The roots of the quadratic equation x2+2x+2 are imaginary which is satisfied by the program

**Program-8**

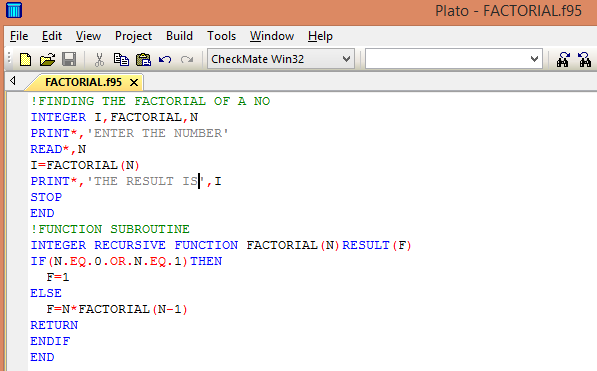
Q : Write a program in FORTRAN to find out the Factorial of a number N.

Features: This program is done with the concept of Recursion and Function Defining.

Constraints: N must be an Integer.

Algorithm:

Program:



Output:



Result: The factorial of 5 is given by the program as 120 which is true, so the program must be correct.

**Program-9**

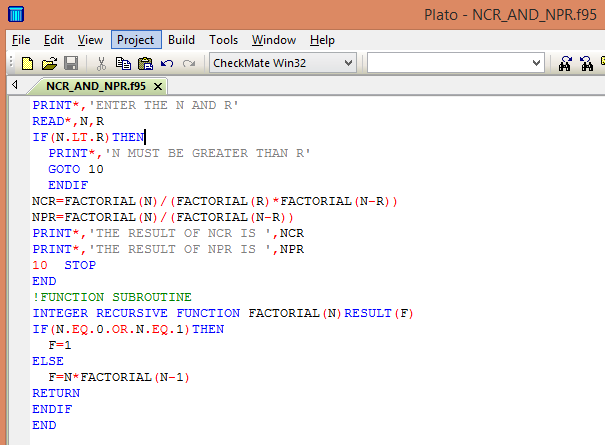
Q : Write a program in FORTRAN to calculate NCR and NPR of numbers N and R

Features : This program is done with the help of the previous program.

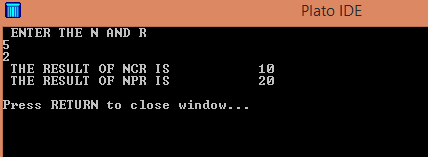
Constraints: N and R must be Integers.

Algorithm:

Program:



Output :



Result:

The output is as expected, so the program must be correct.

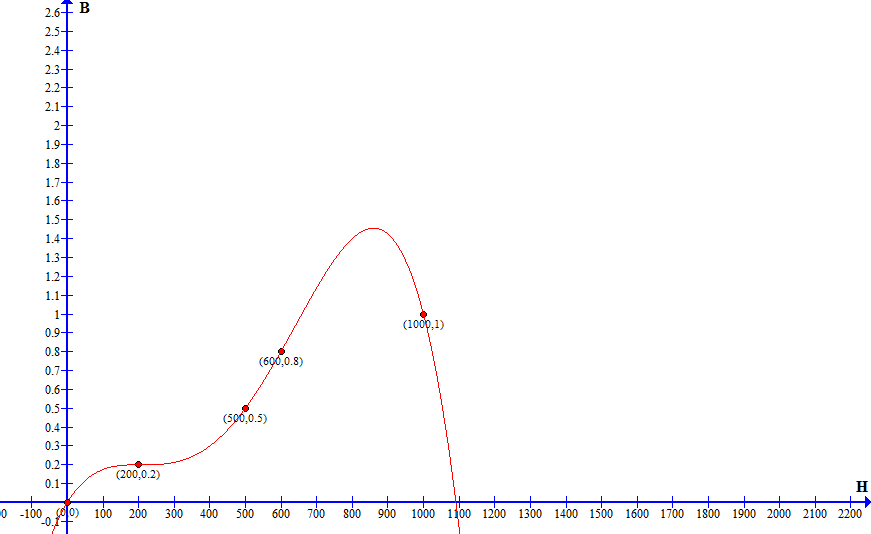
**Program-10**

Q: Write a program in FORTRAN to calculate the B-value corresponding to a given H in a given B-H curve.

Assumptions: The graph between a given pair of H is considered to be linear.

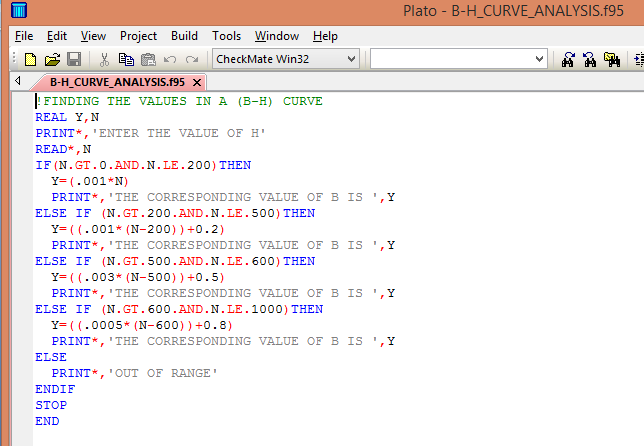
Constraints: H value must not exceed the maximum given limit

Graph:

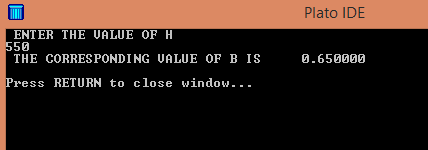


Algorithm:

Program:



Output:



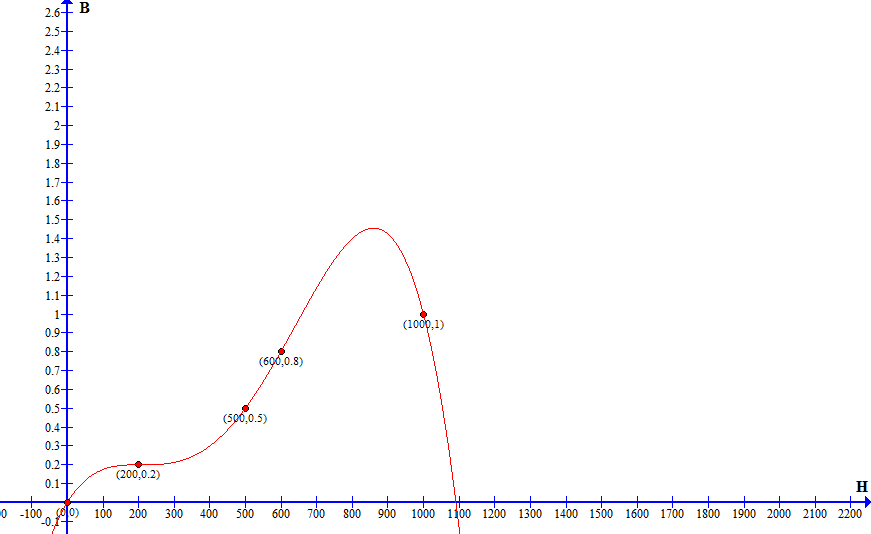
Result:

The result is as expected considering the graph to be linear between two given pair of H.

NOTIFICATION

The above curve is taken as linear between two given pair of points but actually the curve is not linear between the points, so the actual value differs from the calculated value by great margin. So the program is made with some modifications to give more or less correct values with greater reduction in the difference between the actual and calculated values.

Graph:

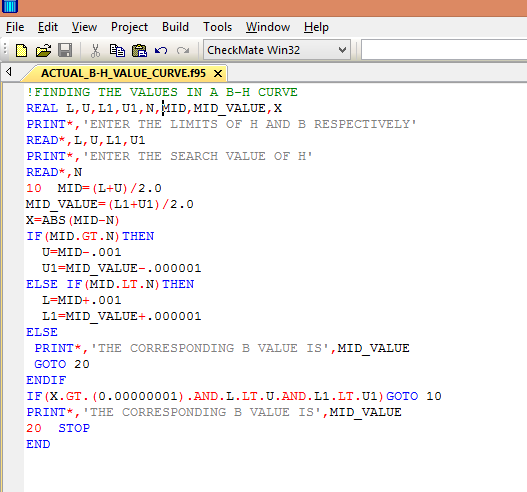


Assumptions : No such assumptions were made.

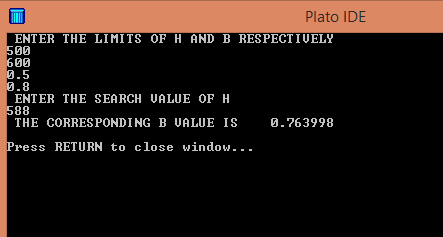
Constraints: The H value must not exceed 1000.

Algorithm:

Program:



Output:



Result: The final output maintains a high degree of accuracy.

**Conclusion**

So far we have solved ten programs in FORTRAN. However this journey of programs has made us learn quite a lot. From being a novice we have become masters of numerical computations in FORTRAN. We have solved varieties of problems like finding solutions of sine series, log series, roots of a quadratic equation, sorting, searching and even finding values by interpolation method in a B-H curve. This journey has also made us realize how powerful a language like FORTRAN can be and the numerous mysteries we can solve with it. In a word this sessional class under the guidance of DC Sir has given us a in depth knowledge of FORTRAN.

**Acknowledgement**

Whenever it comes to such Reports I am always indebited to my mentor and teacher DC Sir who has given me an ocean of knowledge about FORTRAN which has immense importance in my studies and my future prospects in Engineering as well.