PHN – 103

ASSIGNMENT 3

STUDENT INFORMATION

NAME: **JINAL SHAH**

ENROLLMENT NUMBER: **20311011**

PROGRAM 1A

Problem:

Problem-1(a)

The distance between two points (x_1, y_1) and (x_2, y_2) on a Cartesian coordinate plane is given by the equation

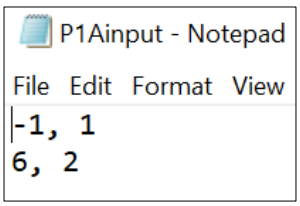
$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Write a Fortran program to calculate the distance between any two points (x1, y1) and (x2, y2) specified by the user. Use the program to calculate the distance between the points (–1,1) and (6,2).

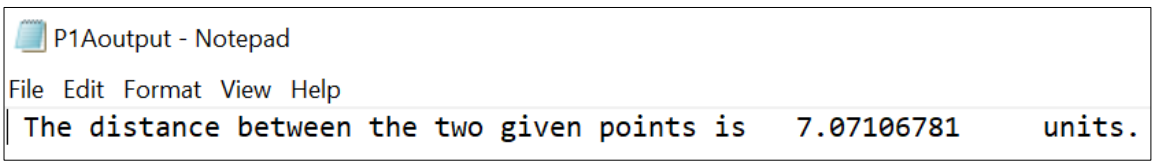
Source Code:

```
program P1A
implicit none

real :: x1, y1, x2, y2, d

!Reading values from the input file
    open (unit=1, file="P1Ainput.dat", status="old", action="read")
        read (1,*) x1, y1
        read (1,*) x2, y2
    close (1)

!Applying distance formula
    d = sqrt((x1-x2)**2 + (y1-y2)**2)

!Displaying output
    open (unit=2, file="P1Aoutput.txt", status="replace", action="write")
        write (2,*) "The distance between the two given points is", d, "units."
    close (2)

end program P1A
```

Input File:



Output File:



PROGRAM 1B

Problem:

Problem-1(b)

Given the two points S (0, 0, 0) and T (6.25, 5.75, 3.10), write a computer program to compute and print the following:

The distance between the two points as well as the direction cosines of the vector ST along X-, Y- and Z-axis

Source Code:

```
program P1B
implicit none

real :: x1, y1, x2, y2, z1, z2, d, l, m, n

!Reading values from the input file
    open (unit=1, file="P1Binput.dat", status="old", action="read")
        read (1,*) x1, y1, z1
        read (1,*) x2, y2, z2
    close (1)

!Applying standard equations
    d = sqrt((x1-x2)**2 + (y1-y2)**2 + (z1-z2)**2 ) !d = distance between points S and T
    l = (x2-x1)/d !l = direction cosine along X axis
    m = (y2-y1)/d !m = direction cosine along Y axis
    n = (z2-z1)/d !n = direction cosine along Z axis

!Displaying output
    open (unit=2, file="P1Boutput.txt", status="replace", action="write")
        write (2,3) "The distance between S (", x1, ",", y1, ",", z1, ") and T (", x2,
",",y2,",",z2, ") is ", d
            3 format (a, f4.2, a, f4.2, a, f4.2, a, f4.2, a, f4.2, a, f4.2, a, f10.7)
        write (2,4) "The direction cosine of ST along X axis is", l
        write (2,4) "The direction cosine of ST along Y axis is", m
        write (2,4) "The direction cosine of ST along Z axis is", n
            4 format (a, f9.6)
    close (2)

end program P1B
```

Input File:

P1Binput - Notepad

File Edit Format View Help

```
0, 0, 0
6.25, 5.75, 3.10
```

Output File:

P1Boutput - Notepad

File Edit Format View Help

```
The distance between S (0.00,0.00,0.00) and T (6.25,5.75,3.10) is  9.0407410
The direction cosine of ST along X axis is 0.691315
The direction cosine of ST along Y axis is 0.636010
The direction cosine of ST along Z axis is 0.342892
```

PROGRAM 2

Problem:

Problem-2: Calculating Orbits

When a satellite orbits the Earth, the satellite's orbit will form an ellipse with the Earth located at one of the focal points of the ellipse. The satellite's orbit can be expressed in polar coordinates as

$$r = \frac{p}{1 - \varepsilon \cos\theta}$$

where r and θ are the distance and angle of the satellite from the center of the Earth, p is a parameter specifying the size of the orbit, and ε is a parameter representing the eccentricity of the orbit.

A circular orbit has an eccentricity ε of zero. An elliptical orbit has an eccentricity of $0 \le \varepsilon \le 1$. If $\varepsilon > 1$, the satellite follows a hyperbolic path and escapes from the Earth's gravitational field.

Consider a satellite with a size parameter p = 1200 km. Write a program to calculate the distance of the satellite from the center of the Earth as a function of θ if the satellite has an eccentricity of (a) ε = 0; (b) ε = 0.25; (c) ε = 0.5.

Write a single program in which r and ε are both input values.

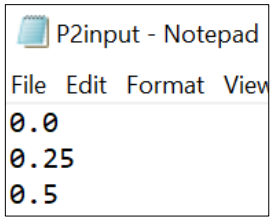
How close does each orbit come to the Earth? How far away does each orbit get from the Earth?

Source Code:

```
program P2
implicit none

    integer :: theta, i
    real :: r
    real, parameter :: p = 1200.0, pi = 4*atan(1.0)
    integer, dimension (3):: theta_min, theta_max
    real, dimension (3):: e, r_min, r_max

    open (unit=1, file="P2input.dat", status="old", action="read")
        do i = 1,3
            read (1,*) e(i)
        end do
    close (1)

!Displaying required outputs
    open ( unit = 2, file = "P2output1.txt", status = "replace")
    write (2, 20) "For epsilon = ", e(1)
    write (2, *) "Theta           r"
    open ( unit = 3, file = "P2output2.txt", status = "replace")
    write (3, 20) "For epsilon = ", e(2)
    write (3, *) "Theta           r"
    open ( unit = 4, file = "P2output3.txt", status = "replace")
    write (4, 20) "For epsilon = ", e(3)
    write (4, *) "Theta           r"
    open ( unit = 5, file = "P2output4.txt", status = "replace")

    do i = 1,3

    theta_min(i)=0
    r_min(i) = p / (1 - e(i)* cos(real(theta_min(i)) * pi/180.0))
    theta_max(i)=1
    r_max(i) = p / (1 - e(i)* cos(real(theta_max(i)) * pi/180.0))

        do theta = 0, 359
```

```
            r = p / (1 - e(i)* cos(real(theta) * pi/180.0)) !Given equation

        write ((i+1), 10) theta,"    ", r
            10 format (i4,a,f10.5)

            !Finding minimum r and locating corresponding theta
            if (r <= r_min(i)) then
                r_min(i) = r
                theta_min(i) = theta
            end if

            !Finding maximum r and locating corresponding theta
            if (r >= r_max(i)) then
                r_max(i) = r
                theta_max(i) = theta
            end if

        end do

        write (5, 20) "For epsilon =", e(i)
            20 format (a,f5.2)
        write (5, 30) " The orbit is closest to Earth at distance ", r_min(i), " km, at
       theta = ", theta_min(i)
        write (5, 30) " The orbit is farthest from Earth at distance ", r_max(i), " km, at
       theta = ", theta_max(i)
            30 format (a,f9.4,a,i3)
    end do

    close(2)
    close(3)
    close(4)
    close(5)

end program P2
```

Input File:



Output File:

P2output1 - Notepad

File Edit Format View Help

```
For epsilon =  0.00
 Theta         r
   0      1200.00000
   1      1200.00000
   2      1200.00000
   3      1200.00000
   4      1200.00000
   5      1200.00000
   6      1200.00000
   7      1200.00000
   8      1200.00000
   9      1200.00000
  10      1200.00000
  11      1200.00000
  12      1200.00000
  13      1200.00000
  14      1200.00000
  15      1200.00000
  16      1200.00000
  17      1200.00000
  18      1200.00000
  19      1200.00000
  20      1200.00000
  21      1200.00000
  22      1200.00000
  23      1200.00000
  24      1200.00000
  25      1200.00000
```

P2output2 - Notepad

File Edit Format View Help

```
For epsilon =  0.25
 Theta         r
   0      1600.00000
   1      1599.91882
   2      1599.67517
   3      1599.26941
   4      1598.70190
   5      1597.97302
   6      1597.08362
   7      1596.03442
   8      1594.82642
   9      1593.46057
  10      1591.93823
  11      1590.26086
  12      1588.42969
  13      1586.44653
  14      1584.31299
  15      1582.03125
  16      1579.60291
  17      1577.03040
  18      1574.31580
  19      1571.46155
  20      1568.46985
  21      1565.34351
  22      1562.08496
  23      1558.69702
  24      1555.18250
  25      1551.54419
```

P2output3 - Notepad

File Edit Format View Help

```
For epsilon =  0.50
 Theta         r
   0      2400.00000
   1      2399.63452
   2      2398.53882
   3      2396.71533
   4      2394.16797
   5      2390.90186
   6      2386.92407
   7      2382.24316
   8      2376.86841
   9      2370.81128
  10      2364.08423
  11      2356.70068
  12      2348.67578
  13      2340.02539
  14      2330.76636
  15      2320.91675
  16      2310.49536
  17      2299.52173
  18      2288.01660
  19      2276.00024
  20      2263.49463
  21      2250.52124
  22      2237.10278
  23      2223.26147
  24      2209.02026
  25      2194.40161
```

Note: Sir, these three output files (P2output1.txt, P2output2.txt, P2output3.txt) have all values of r corresponding to 360 values of theta. Since the files are very long, I am attaching only upto some values here, as mentioned by you during class.

P2output4 - Notepad

File Edit Format View Help

```
For epsilon = 0.00
 The orbit is closest to Earth at distance 1200.0000 km, at theta = 359
 The orbit is farthest from Earth at distance 1200.0000 km, at theta = 359
For epsilon = 0.25
 The orbit is closest to Earth at distance  960.0000 km, at theta = 180
 The orbit is farthest from Earth at distance 1600.0000 km, at theta =   0
For epsilon = 0.50
 The orbit is closest to Earth at distance  800.0000 km, at theta = 180
 The orbit is farthest from Earth at distance 2400.0000 km, at theta =   0
```

PROGRAM 3

Problem:

Problem 3: Current through a Diode

The current flowing through the semiconductor diode shown is given by the equation

$$i_D = i_0(e^{\frac{qV_D}{kT}} - 1)$$

where
V_D = the voltage across the diode, in volts
i_D = the current flow through the diode, in amperes
i_0 = the leakage current of the diode, in amperes
q = the charge on an electron, 1.602×10^{-19} C
k = Boltzmann's constant, 1.38×10^{-23} J/K
T = temperature, in kelvins (K)
The leakage current i_0 of the diode is 2.0 μA. Write a computer program to calculate the current flowing through this diode for all voltages from −1.0 V to + 0.6 V, in 0.1 V steps. Repeat this process for the following temperatures: 75 °F , 100 °F, and 125 °F. Convert the temperatures from °F to kelvins.

Source Code:

```
program P3
implicit none

    integer :: i, Vo
    real, dimension (3) :: Tf, Tk
    real :: Id, Vd
    real, parameter :: Io = 2.0e-6 , q = 1.602e-19, k = 1.38e-23

!Saving given temperatures (F) in an array
    Tf = (/75, 100, 125 /)

!Converting Temperature from fahrenheit to kelvin
    do i = 1,3
        Tk(i)= ( Tf(i) - 32 ) * ( 5.0 / 9.0 ) + 273.15
    end do

!Displaying output of Current corresponding to Voltage for each Temperature
    open (unit = 2, file = "P3output.txt", status = "replace")

    do i = 1,3
        write (2,10) "At T=",Tf(i), " F"
            10 format (a, f6.1 ,a)
        write (2,*)
        write (2,*)"Voltage (V)                    Current (A)"
        write (2,*)

            do Vo = -10, 6, 1
                Vd = Vo/10.0
                Id = Io * ( exp(( q * (Vd) ) / ( k * Tk(i) )) - 1 ) !Given equation

                write (2, 20) Vd, "           ", Id
                    20 format (f5.1, a, f30.15)
            end do

        write (2, *)
    end do

    close(2)

end program P3
```

Note: Sir, since in do loop, only integer type is accepted and the Voltage(Vd) type was real, so I introduced another integer type variable called Vo for my do loop, and substituted value of Vd in terms of Vo. (`Vd = Vo/10.0`*)*

Input File:

Input File was not required in this question.

Output File:

P3output - Notepad

File Edit Format View Help

```
At T=  75.0 F

 Voltage (V)                    Current (A)

 -1.0                    -0.000001999999995
 -0.9                    -0.000001999999995
 -0.8                    -0.000001999999995
 -0.7                    -0.000001999999995
 -0.6                    -0.000001999999995
 -0.5                    -0.000001999999995
 -0.4                    -0.000001999999768
 -0.3                    -0.000001999983851
 -0.2                    -0.000001999193728
 -0.1                    -0.000001959844440
  0.0                     0.000000000000000
  0.1                     0.000097612493846
  0.2                     0.004959324840456
  0.3                     0.247103020548820
  0.4                    12.307368278503418
  0.5                   612.983215332031250
  0.6                 30530.443359375000000
```

P3output - Notepad

File Edit Format View Help

```
At T= 100.0 F

 Voltage (V)                    Current (A)

 -1.0                    -0.000001999999995
 -0.9                    -0.000001999999995
 -0.8                    -0.000001999999995
 -0.7                    -0.000001999999995
 -0.6                    -0.000001999999995
 -0.5                    -0.000001999999995
 -0.4                    -0.000001999999313
 -0.3                    -0.000001999972710
 -0.2                    -0.000001998856760
 -0.1                    -0.000001952185130
  0.0                     0.000000000000000
  0.1                     0.000081656085968
  0.2                     0.003497170517221
  0.3                     0.146361410617828
  0.4                     6.122094631195068
  0.5                   256.075378417968750
  0.6                 10711.124023437500000
```

P3output - Notepad

File Edit Format View Help

```
At T= 125.0 F

 Voltage (V)                    Current (A)

 -1.0                    -0.000001999999995
 -0.9                    -0.000001999999995
 -0.8                    -0.000001999999995
 -0.7                    -0.000001999999995
 -0.6                    -0.000001999999995
 -0.5                    -0.000001999999995
 -0.4                    -0.000001999998858
 -0.3                    -0.000001999955884
 -0.2                    -0.000001998426796
 -0.1                    -0.000001943908728
  0.0                     0.000000000000000
  0.1                     0.000069312387495
  0.2                     0.002540728310123
  0.3                     0.090661995112896
  0.4                     3.232731819152832
  0.5                   115.266891479492188
  0.6                  4109.979492187500000
```

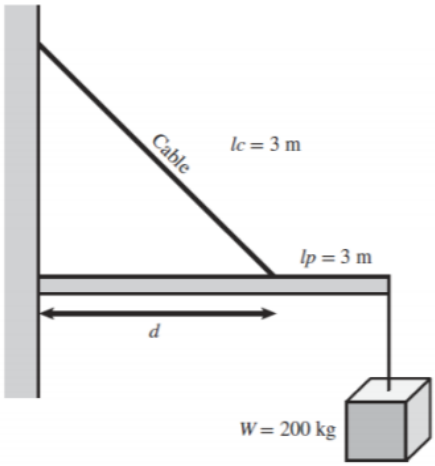
PROGRAM 4

Problem:

Problem 4: Tension on a Cable

A 200 kilogram object is to be hung from the end of a rigid 3-m horizontal pole of negligible weight, as shown in the Figure. The pole is attached to a wall by a pivot and is supported by a 3-m cable that is attached to the wall at a higher point. The tension on this cable is given by the equation

$$T = \frac{W \cdot lc \cdot lp}{d\sqrt{lp^2 - d^2}}$$



where T is the tension on the cable, W is the weight of the object, lc is the length of the cable, lp is the length of the pole, and d is the distance along the pole at which the cable is attached.

Write a program to determine the distance d at which to attach the cable to the pole in order to minimize the tension on the cable. To do this, the program should calculate the tension on the cable at 0.1 m intervals from d = 0.5 m to d = 2.8 m, and should locate the position d that produces the minimum tension.

Source Code:

```
program P4
implicit none

    real :: d, T, d_min, T_min
    real, parameter :: Lc = 3.0, Lp = 3.0, W = 1960.0 !W = mg, m=200, g=9.8

!Initializing variables
    d = 0.5
    d_min = 0.5
    T_min = (W * Lp * Lc)/(d_min * sqrt((lp ** 2) - (d_min ** 2)))

!Displaying required outputs
    open ( unit = 1, file = "p4output.txt", status = "replace")
    write (1, *) "Value of d     Tension (T)"

    do while (d <= 2.8)
        T = (W * Lp * Lc)/(d * sqrt((Lp ** 2) -(d ** 2))) !Given equation

        if (T <= T_min) then !Finding minimum Tension and locating corresponding distance
            T_min = T
            d_min = d
        end if

        write (1, 10) d, "           ", T !Displaying respective Tension for each distance
            10 format (f5.1, a, f10.4)

        d = d + 0.1

    end do

    !Displaying minimum tension and corresponding distance
    write (1,20) "The minumum value of Tension is", T_min, " N at distance of", d_min, " m."
        20 format (a, f10.4, a, f4.1, a)

    close (1)

end program P4
```

Input File:

Input File was not required in this question.

Output File:

p4output - Notepad

File Edit Format View Help

```
 Value of d      Tension (T)
  0.5            11926.8164
  0.6            10002.0830
  0.7             8638.4482
  0.8             7626.1514
  0.9             6848.7935
  1.0             6236.6812
  1.1             5745.6230
  1.2             5346.3379
  1.3             5018.7607
  1.4             4748.8032
  1.5             4526.4258
  1.6             4344.4580
  1.7             4197.8701
  1.8             4083.3330
  1.9             3998.9924
  2.0             3944.4238
  2.1             3920.7842
  2.2             3931.2371
  2.3             3981.8398
  2.4             4083.3333
  2.5             4254.9277
  2.6             4533.1606
  2.7             4996.1621
  2.8             5849.3970
The minumum value of Tension is 3920.7842 N at distance of 2.1 m.
```

PROGRAM 5

Problem:

Problem 5: Geometric Mean

The geometric mean of a set of numbers x1 through xn is defined as the nth root of the product of the numbers:

geometric mean = $\sqrt[n]{x_1 x_2 x_3 \ldots . x_n}$

Write a Fortran program that will accept an arbitrary number of positive input values and calculate both the arithmetic mean (i.e., the average) and the geometric mean of the numbers.

Use a while loop to get the input values, and terminate the inputs a user enters a negative number.

Test your program by calculating the average and geometric mean of the four numbers 10, 5, 4, and 5.

Source Code:

```
program P5
implicit none

        integer :: i, j, n, flag
        real :: sum, product, am, gm
        real, dimension(:), allocatable :: a

        sum = 0
        product = 1

!Reading from input file
        open(unit = 1, file = "P5input.dat", status = 'old', action = "read")

            read (1,*) n !Reading number of inputs
            allocate(a(n))

            i = 1
            do while (i <= n) !Reading elements from file into array
                read (1,*) a(i)
                i=i+1
            end do
        close(1)

!Checking inputs
        i = 1
        flag = 0
        do while (i <= n)
            if (a(i) < 0) then
                flag = 1 !Used to indicate negative input
            exit !Terminating loop on negative user input

            else
                i = i+1

            end if
        end do

!Displaying output
        open(unit=2 , file="P5output.txt", status = "replace", action = "write")

        if (flag == 1) then
            write (2, *) "You cannot input negative number."

        else
            j = 1
            do while (j <= n) !loop to find sum and product
                sum = sum + a(j)
                product = product*(a(j))
                j = j+1
```

```
            end do

            am = sum / n !Arithmetic Mean
            gm = (product) ** (1.0/ n) !Geometric Mean

            write (2, *) "The arithmetic mean is", am
            write (2, *) "The geometric mean is", gm

        end if
        close (2)

            deallocate(a)

end program P5
```

Input File:

P5input - Notepad

File Edit Format View

```
4
10
5
4
5
```

Output File:

P5output - Notepad

File Edit Format View Help

```
 The arithmetic mean is    6.00000000
 The geometric mean is    5.62341309
```

PROGRAM 6

Problem:

Problem:6

Antenna Gain Pattern The gain *G* of a certain microwave dish antenna can be expressed
as a function of angle by the equation

$$G(\theta) = |sinc\ 6\theta| \qquad \text{for } -\frac{\pi}{2} \le \theta \le \frac{\pi}{2}$$

where θ is measured in radians from the boresite of the dish, and the sinc function is defined as follows:

$$sinc\ x = \begin{cases} \frac{\sin x}{x} & x \neq 0 \\ 0 & x = 0 \end{cases}$$

Calculate a table of gain versus the angle off boresite *in degrees* for this antenna for the range $0° \le \theta \le 90°$ in 1° steps.
Label this table with the title "Antenna Gain vs Angle (deg)", and include column headings on the output.

Source Code:

```
program P6
implicit none

    integer :: theta
    real, dimension(91) :: G
    real, parameter :: pi = 4 * atan(1.0)

    theta = 0 !Initializing theta

    open(unit=1 , file="P6output.txt")
            write (1, *) "  Antenna Gain vs Angle (deg)" !Labelling table with title
            write (1, *)
            write (1, *) "  Antenna Gain    Angle (deg)" !Column headings
            write (1, *)

        do while (theta <= 90)

        !Calculating value of antenna gain
            if ( theta == 0 ) then
                G(theta) = 0.0
            else
                G(theta) = abs((sin(6.0  * theta * pi / 180.0 )) / (6.0  * theta * pi / 180.0))

            write (1,10) G(theta),"          ", theta !Displaying table in output
                10 format (f10.5,a,i4)

            end if

        theta = theta + 1
        end do
    close(1)

end program P6
```

Input File:

Input File was not required in this question.

Output File:

P6output - Notepad

File Edit Format View Help

```
Antenna Gain vs Angle (deg)

Antenna Gain    Angle (deg)

0.99817             1
0.99271             2
0.98363             3
0.97101             4
0.95493             5
0.93549             6
0.91282             7
0.88706             8
0.85839             9
0.82699            10
0.79307            11
0.75683            12
0.71851            13
0.67836            14
0.63662            15
0.59356            16
0.54945            17
0.50455            18
0.45914            19
0.41350            20
0.36788            21
0.32257            22
0.27781            23
0.23387            24
0.19099            25
```

Note: Sir, this output file (P6output.txt) has all values of Antenna Gain corresponding to 90 values of Angle. Since the file is very long, I am attaching only upto some values here.

PROGRAM 7

Problem:

Problem: 7

Sum the series

$$x = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots$$

for 20, 50 and 100 terms.

Source Code:

```
program P7
implicit none

    integer :: i, j
    integer, dimension(1:3) :: n
    real :: term
    real, dimension(1:3) :: sum

!Initializing variables
    term = 1
    sum = 0

!Reading from input file
    open(unit=1, file= "P7input.dat", status= "old", action="read")
        read(1,*) n
    close (1)

    do i= 1, 3
        do j= 1,(n(i)+1)
            term = ((-1)**(real(j)+1))/((2*real(j)-1)) !General term of series
            sum(i)=sum(i)+term !sum of series
        end do
    end do

!Displaying output
    open(unit=2, file= "P7output.txt", status= "replace", action="write")
        do i = 1, 3
            write (2, 3) "Sum of series up to", n(i), "terms is", sum(i)
                3 format (a20, i5, a10, f13.9)
        end do
    close (2)

end program P7
```

Input File:

P7input - Notepad

File Edit Format View

```
20, 50, 100
```

Output File:

P7output - Notepad

File Edit Format View Help

```
 Sum of series up to   20  terms is   0.797296166
 Sum of series up to   50  terms is   0.790299535
 Sum of series up to  100  terms is   0.787873268
```

PROGRAM 8

Problem:

Problem 8:

Write a program that can read two matrices of arbitrary size from two input disk files, and multiply them if they are of compatible sizes. If they are of incompatible sizes, an appropriate error message should be printed. The number of rows and columns in each matrix will be specified by two integers on the first line in each file, and the elements in each row of the matrix will be found on a single line of the input file.
Take example of the matrices A and B below.

$$A = \begin{bmatrix} 1. & -5. & 4. & 2. \\ -6. & -4. & 2. & 2. \end{bmatrix}$$

$$B = \begin{bmatrix} 1. & -2. & -1. \\ 2. & 3. & 4. \\ 0. & -1. & 2. \\ 0. & -3. & 1. \end{bmatrix}$$

Source Code:

```
program P8
implicit none

integer :: row1, col1, row2, col2, i, j, k, sum
real, dimension(:,:), allocatable :: m1, m2, m3

!Reading from input files
    !Reading matrix A
    open(unit = 1, file = "P8input1.dat", status = "old", action = "read")
        read (1,*) row1, col1
        allocate(m1(row1, col1)) !Matrix A

        do i = 1,row1
            read(1, *, end=10) (m1(i,j), j = 1, col1)
        end do
    10 close(1)

    !Reading matrix B
    open(unit = 2, file = "P8input2.dat", status = "old", action = "read")
        read (2,*) row2, col2
        allocate(m2(row2, col2)) !Matrix B

        do i = 1,row2
            read(2, *, end=20) (m2(i,j), j = 1, col2)
        end do
    20 close(2)


!Displaying output file
    open(unit = 3, file = "P8output.txt", status = "replace", action = "write")
        if (col1 /= row2) then
            write (3,*) "Invalid input : Matrices incompatible." !Error message
        else
            allocate(m3(row1,col2)) !Product Matrix
            sum = 0
                do i =1,row1
                    do j = 1,col2
                        do k = 1,col1
                            sum = sum + (m1(i,k)*m2(k,j))
```

```
                        end do
                        m3(i,j) = sum
                        sum = 0
                    end do
                end do

    write(3, *) "The resulting product matrix is"
            do i = 1, row1
                write(3,*) (m3(i,j), j = 1, col2)
            end do
        end if
    close (3)

    deallocate(m1)
    deallocate(m2)

end program P8
```

Input File:

P8input1 - Notepad

File Edit Format View

```
2,4
1, -5, 4, 2
-6, -4, 2, 2
```

P8input2 - Notepad

File Edit Format View Help

```
4,3
1, -2, -1
2, 3, 4
0, -1, 2
0, -3, 1
```

Output File:

P8output - Notepad

File Edit Format View Help

```
The resulting product matrix is
  -9.00000000       -27.0000000       -11.0000000
  -14.0000000       -8.00000000       -4.00000000
```

PROGRAM 9

Problem:

Problem 9

Write a program to compute and print the values of the safe loading S for different R=25, 50, 75, ……..250 using following criteria.

$S = 17000 - 0.485\ R^2$ for R < 120

$S = 18000/\ (1 + R2\ /1800)$ for R = or >120

Source Code:

```
program P9
implicit none

    integer :: R
    real :: S !Safe loading

    open (unit = 1, file = "P9output.txt", status = "replace", action = "write")

        write (1,*) " Value of R        Safe Loading (S)"

        !given Safe loading equations
        do R = 25, 250, 25
            if ( R < 120 ) then
                S = 17000.0 - 0.485 * (R ** 2.0)
            else
                S = 18000.0 / ( 1 + (( R ** 2.0) / 18000.0))
            end if
            write (1, 2) R, S
            2 format (i8, f25.4)
        end do
    close (1)

end program P9
```

Input File:

Input File was not required in this question.

Output File:

P9output - Notepad

File Edit Format View Help

Value of R	Safe Loading (S)
25	16696.8750
50	15787.5000
75	14271.8750
100	12150.0000
125	9635.6875
150	8000.0000
175	6663.2393
200	5586.2070
225	4721.3115
250	4024.8447

PROGRAM 10

Problem:

Problem 10

The wavelengths in Angstrom of the hydrogen spectrum are given by the following formula

$$\text{Wavelength} = 911.8 / (1/n^2 - 1/m^2)$$

Write a program to produce a table of values of wavelengths for all combinations of

$$m = 2, 3, 4, 5, \ldots\ldots\ldots, 50$$
$$\text{and} \quad n = 1, 2, 3, 4, 5, \ldots\ldots\ldots, (m-1)$$

The program should compute and print the wavelengths for all the desired combinations of n and m

Source Code:

```
program P10
implicit none

    integer :: n, m
    real :: W !wavelength

    open (unit = 1, file = "P10output.txt", status = "replace", action = "write")

        write (1,*) "(m)       (n)       Wavelength(W)"

        do m = 2, 50, 1
            do n = 1, (m-1), 1
                W = 911.8 / ((1.0/(n ** 2)) - (1.0/(m ** 2))) !Given equation

                write (1,2) m, n, W
                2 format (i3, i9, f20.5)

            end do
        end do
    close (1)

end program P10
```

Input File:

Input File was not required in this question.

Output File:

P10output - Notepad

File Edit Format View Help

(m)	(n)	Wavelength(W)
2	1	1215.73328
3	1	1025.77502
3	2	6564.95996
4	1	972.58667
4	2	4862.93311
4	3	18757.02734
5	1	949.79163
5	2	4341.90479
5	3	12822.18750
5	4	40524.44531
6	1	937.85144
6	2	4103.10010
6	3	10941.59961
6	4	26259.83984
6	5	74601.82031
7	1	930.79584
7	2	3971.39551
7	3	10052.59473
7	4	21662.15820
7	5	46539.79297
7	6	123724.24219
8	1	926.27301
8	2	3890.34668
8	3	9549.03223
8	4	19451.73242
8	5	37407.17969
8	6	75028.10938
8	7	190626.98438

P10output - Notepad

File Edit Format View Help

50	15	225445.04688
50	16	260049.90625
50	17	297953.62500
50	18	339410.84375
50	19	384712.25000
50	20	434190.46875
50	21	488227.03125
50	22	547260.87500
50	23	611798.81250
50	24	682428.25000
50	25	759833.31250
50	26	844814.68750
50	27	938314.81250
50	28	1041449.87500
50	29	1155551.25000
50	30	1282218.75000
50	31	1423391.50000
50	32	1581441.75000
50	33	1759302.25000
50	34	1960641.37500
50	35	2190107.75000
50	36	2453681.00000
50	37	2759182.50000
50	38	3117043.50000
50	39	3541490.75000
50	40	4052444.50000
50	41	4678680.50000
50	42	5463367.00000
50	43	6474340.00000
50	44	7824666.50000
50	45	9717868.00000
50	46	12560995.00000
50	47	17303834.00000
50	48	26795754.00000
50	49	55283632.00000

Note: *Sir, this output file (P10output.txt) has all values of Wavelength corresponding to desired values of m and n. Since the file is very long, I am attaching snippets of the start and end of this file.*

PROGRAM 11

Problem:

Problem 11

Prepare a program to read a matrix A and B of order 6 x 6 and calculate their sum and print it.

Also in the same program obtain the trace of each matrix (i.e. sum of all the diagonal elements) and print their values using an E format.

Source Code:

```
program P11
implicit none

    integer :: i, j, k
    real :: sum, T1, T2
    real, dimension(6,6):: m1, m2, m3

    sum = 0.0
    t1 = 0.0
    t2 = 0.0

!Reading from input files
    open(unit = 1, file = "P11input1.dat", status = "old", action = "read")
        do i = 1,6
            read(1, *, end=10) (m1(i,j), j = 1, 6) !matrix 1
        end do
    10 close(1)

    open(unit = 2, file = "P11input2.dat", status = "old", action = "read")
        do i = 1,6
            read(2, *, end=20) (m2(i,j), j = 1, 6) !matrix 2
        end do
    20 close(2)

!Calculating sum and trace
        do i = 1,6
            do j =1,6

            !sum
            sum = m1(i,j) + m2(i,j)
            m3(i,j) = sum !Sum matrix
            sum = 0

            !trace
            if (i == j) then
                T1 = T1 + m1(i,j) !trace of matrix 1
                T2 = T2 + m2(i,j) !trace of matrix 1
            end if

            end do
        end do

!Displaying output file
    open(unit = 3, file = "P11output.txt", status = "replace", action = "write")
        write(3, *) "The resulting sum matrix is"
            do i =1, 6
                write(3,*) (m3(i,j), j = 1, 6)
            end do

        write(3, 30) "Trace of Matrix 1 is", T1
        write(3, 30) "Trace of Matrix 2 is", T2
            30 format (a20, e13.5) !E format
    close(3)

end program P11
```

Input File:

P11input1 - Notepad

File Edit Format View

```
2 6 2 3 0 7
0 1 6 4 0 1
9 9 0 3 9 0
3 8 5 0 3 1
4 4 9 2 5 7
3 0 2 5 6 7
```

P11input2 - Notepad

File Edit Format View

```
6 3 1 1 4 9
6 1 5 7 1 7
1 7 4 8 0 8
2 0 4 6 8 0
4 5 3 0 0 9
2 1 8 1 3 8
```

Note: I have used arbitary matrices as the question did not mention any specific matrices.

Output File:

P11output - Notepad

File Edit Format View Help

```
 The resulting sum matrix is
   8.00000000       9.00000000       3.00000000       4.00000000       4.00000000       16.0000000
   6.00000000       2.00000000       11.0000000       11.0000000       1.00000000       8.00000000
   10.0000000       16.0000000       4.00000000       11.0000000       9.00000000       8.00000000
   5.00000000       8.00000000       9.00000000       6.00000000       11.0000000       1.00000000
   8.00000000       9.00000000       12.0000000       2.00000000       5.00000000       16.0000000
   5.00000000       1.00000000       10.0000000       6.00000000       9.00000000       15.0000000
Trace of Matrix 1 is  0.15000E+02
Trace of Matrix 2 is  0.25000E+02
```

PROGRAM 12

Problem:

Problem 12

Make a computer program by taking three matrices A, B and C of appropriate orders and read their elements and then multiply them to obtain a matrix D. Print its elements.

Source Code:

```
module multiplication
contains

subroutine multiply(a,b,r1,c1,r2,c2,t1)
implicit none
    integer :: i, j, k
    real :: sum
    integer :: r1, c1, r2, c2
    real, dimension(:,:) :: a, b
    real, dimension(:,:) :: t1

    sum = 0
    do i =1,r1
        do j = 1,c2
            do k = 1,c1
                sum = sum + (a(i,k) * b(k,j))
            end do
            t1(i,j) = sum
            sum = 0
        end do
    end do
end subroutine multiply
end module multiplication

program P12
use multiplication
implicit none

    integer :: row1, col1, row2, col2, row3, col3, i, j, k
    real, dimension(:,:), allocatable :: m1, m2, m3
    real, dimension(:,:), allocatable :: m4, m5

    !Reading from input files
        !Matrix A
        open(unit = 1, file = "P12input1.dat", status = "old", action = "read")
            read (1,*) row1, col1
            allocate(m1(row1, col1)) !Matrix A

            do i = 1,row1
                read(1, *, end=10) (m1(i,j), j = 1, col1)
            end do
        10 close(1)

        !Matrix B
        open(unit = 2, file = "P12input2.dat", status = "old", action = "read")
            read (2,*) row2, col2
            allocate(m2(row2, col2)) !Matrix B

            do i = 1,row2
                read(2, *, end=20) (m2(i,j), j = 1, col2)
            end do
        20 close(2)

        !Matrix C
            open(unit = 3, file = "P12input3.dat", status = "old", action = "read")
            read (3,*) row3, col3
            allocate(m3(row3, col3)) !Matrix C
```

```
            do i = 1,row3
                read(3, *, end=30) (m3(i,j), j = 1, col3)
            end do
        30 close(3)

    !Displaying output file
        open(unit = 4, file = "P12output.txt", status = "replace", action = "write")
            if (col1 /= row2 .or. col2 /= row3 ) then
                write (4,*) "Invalid input : Matrices incompatible."
            else
                allocate(m4(row1, col2)) !Matrix A*B
                call multiply(m1,m2,row1,col1,row2,col2,m4)

                allocate(m5(row1, col3)) !Matrix D
                call multiply(m4,m3,row1,col2,row3,col3,m5)

            !Printing elements of Matrix D
                write(4, *) "The resulting product matrix D is"
                do i = 1, row1
                    write(4,*) (m5(i,j), j = 1,col3)
                end do
            end if
        close(4)

        deallocate(m1)
        deallocate(m2)
        deallocate(m3)
        deallocate(m4)
        deallocate(m5)

end program P12
```

Note: Sir, as you explained in class, one way to return an array from a subprogram could be COMMON statement to avoid rank mismatch error. However, I used the other method. I allocated the dimensions of my result array before calling the subprogram, since it was a dynamic array and its dimensions depend on the matrices in input files.

Input File:

P12input1 - Notepad

File Edit Format View

```
2,3
6, 1, 1
1, 7, 4
```

P12input2 - Notepad

File Edit Format View

```
3,4
1, 0, 8, 8
1, 6, 3, 7
8, 7, 9, 2
```

P12input3 - Notepad

File Edit Format View

```
4,5
5, 2, 4, 6, 4
6, 4, 6, 1, 6
3, 1, 0, 7, 6
3, 6, 8, 2, 5
```

Output File:

P12output - Notepad

File Edit Format View Help

```
 The resulting product matrix D is
   504.000000       484.000000       594.000000       637.000000       783.000000
   1010.00000       815.000000       1100.00000       895.000000       1295.00000
```

PROGRAM 13

Problem:

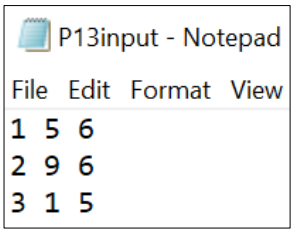
Problem 13

Make a function subprogram to evaluate second order determinant and utilize it to obtain the value of a third order determinant.

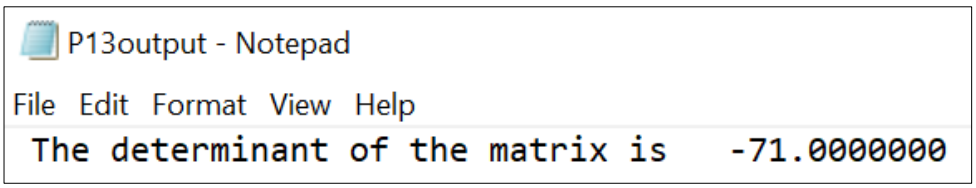
Source Code:

```
module determinant
contains

real function det(a)
implicit none
    real, dimension(2,2) :: a

    det = (a(1,1) * a(2,2)) - (a(1,2) * a(2,1))

end function det
end module determinant

program P13
use determinant
implicit none

    integer :: i,j
    real, dimension(3,3) :: m
    real, dimension(2,2) :: m11, m12, m13
    real :: d, d11, d12, d13

!Reading from input files
    open(unit = 1, file = "P13input.dat", status = "old", action = "read")

        !Loop to read matrix
        do i = 1,3
            read(1, *, end=10) (m(i,j), j = 1, 3)
        end do
    10 close(1)

!Finding minor matrix corresponding to each element in first row
    m11 = m(2:3,2:3)
    m12(:,1) = m(2:3,1)
    m12(:,2) = m(2:3,3)
    m13 = m(2:3,1:2)

!Finding determinant of minor matrices using function
    d11 = det(m11)
    d12 = det(m12)
    d13 = det(m13)

!calculating determinant of main 3x3 matrix
    d = ((m(1,1))*d11)-((m(1,2))*d12)+((m(1,3))*d13)

!Displaying determinant in output
    open(unit = 2, file = "P13output.txt", status = "replace", action = "write")
        write (2, *) "The determinant of the matrix is ", d
    close(2)
end program P13
```

Input File:



Output File:



PROGRAM 14

Problem:

Problem 14

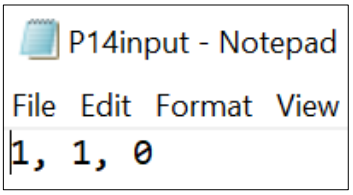
Write a subprogram which calculates the spherical coordinates (r, theta and phi) of a given point in Cartesian coordinates (x, y, z) using their usual relations

Source Code:

```
program P14
implicit none

    real :: x, y, z, r, theta, phi

!Reading Cartesian coordinates from input file
    open (unit = 1, file = "P14input.dat", status = "old")
        read (1,*) x, y, z
    close(1)

!Using subroutine to find Spherical coordinates
    call coordinates(x,y,z,r,theta,phi)

    open (unit = 2, file = "P14output.txt", status = "replace")
        write (2,*) "r = ", r
        write (2,*) "theta = ", theta
        write (2,*) "phi = ", phi
    close(2)

end program P14

subroutine coordinates(x,y,z,r,theta,phi)
implicit none
    real, intent(in) :: x, y, z
    real, intent(out) :: r, theta, phi

    r = sqrt(x**2 + y**2 + z**2)
    theta = acos(z/sqrt(x**2 + y**2 + z**2))
    phi = atan(y/x)

end subroutine coordinates
```

Note: Sir, the question asks for a subprogram, so here I have used the subroutine subprogram (as I needed the subprogram to return three results – r, theta, phi).
I have also written a main program and taken arbitary values of cartesian coordinates in input file, to confirm if my subroutine is working correctly.

Input File:



Output File: