XIE Dong (谢栋)'s TA report for assignment06

SID: 12149022

Github: https://github.com/X-Giraffe/ESE5023_Assignment_12149022

Responsible TA: HUANG Hao

Grade: 40

1.2 定义方程

Good for 1(15/15)

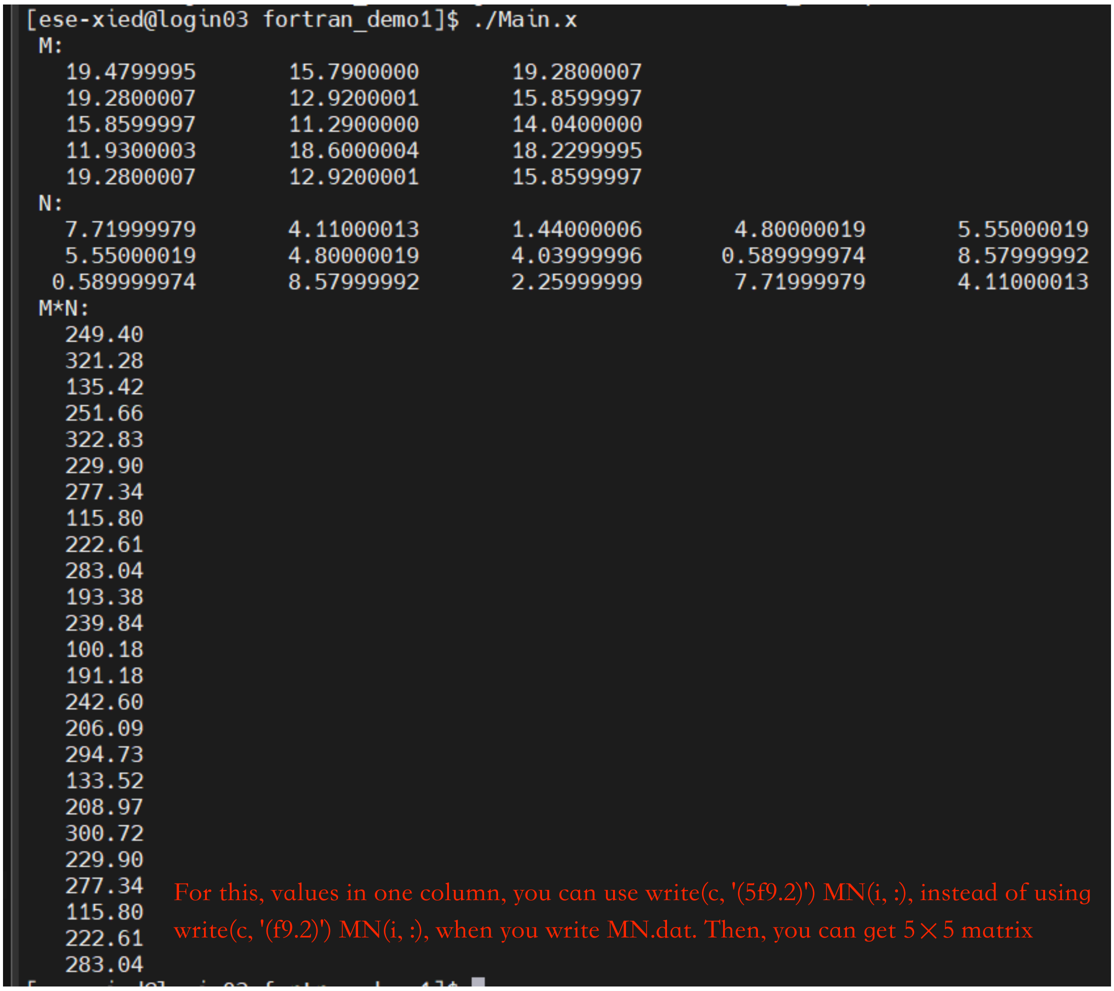
```
subroutine Matrix_multip(x,y,z)
        real(4)::x(5,3),y(3,5),z(5,5)
    !M * N
        z = MATMUL(x,y)

    end subroutine    Matrix_multip
```

1.1 和 1.3 读取矩阵，并完成计算

```
program Main
    implicit none

    integer::i,j,k,m,n,u
    real(4)::x(5,3),y(3,5),z(5,5)



    u=1
    open(u,file="/work/ese-xied/fortran_demo1/M.dat",status='old')

    do i=1,5
        read(u,*) (x(i,j),j=1,3)

    end do
    close(u)
    write(*,*)'M:'
    write(*,*)x(1,:)
    write(*,*)x(2,:)
    write(*,*)x(3,:)
    write(*,*)x(4,:)
    write(*,*)x(5,:)


     u=2
    open(u,file="/work/ese-xied/fortran_demo1/N.dat",status='old')

    do i=1,3
        read(u,*) (y(i,j),j=1,5)

    end do
    close(u)
    write(*,*)'N:'
    write(*,*)y(1,:)
    write(*,*)y(2,:)
    write(*,*)y(3,:)
```

```
        call Matrix_multip(x,y,z)
        write(*,*)'M*N:'
        do i=1,5
        write(*,'(f9.2)'),z(i,:)
        end do

end program Main
```

1.3 输出结果



```
[ese-xied@login03 fortran_demo1]$ ./Main.x
 M:
   19.4799995       15.7900000       19.2800007
   19.2800007       12.9200001       15.8599997
   15.8599997       11.2900000       14.0400000
   11.9300003       18.6000004       18.2299995
   19.2800007       12.9200001       15.8599997
 N:
   7.71999979       4.11000013       1.44000006       4.80000019       5.55000019
   5.55000019       4.80000019       4.03999996      0.589999974       8.57999992
  0.589999974       8.57999992       2.25999999       7.71999979       4.11000013
 M*N:
   249.40
   321.28
   135.42
   251.66
   322.83
   229.90
   277.34
   115.80
   222.61
   283.04
   193.38
   239.84
   100.18
   191.18
   242.60
   206.09
   294.73
   133.52
   208.97
   300.72
   229.90
   277.34
   115.80
   222.61
   283.04
```

For this, values in one column, you can use write(c, '(5f9.2)') MN(i, :), instead of using write(c, '(f9.2)') MN(i, :), when you write MN.dat. Then, you can get 5×5 matrix

2.1 编写计算赤角的模块

Correct for 2 (25/25)

```
module Declination_angle

contains

   subroutine cal_declination_angle(day,delta)
```

```
        implicit none

        real(4), parameter          :: pi = 3.14
        integer(4), intent(in)    :: day
        real(4), intent(out)       :: angel
```

I suggest you to use asind and sin, replacing asin(*180/pi) and sin(pi/180*).

```
        angel = asin(sin(-23.44*pi/180)*cos(pi/180*(360/365.24*(day+10)+360/pi*0.0167*sin(pi/180*360/365.24*(day-2)))))
        print*, 'angel = ', angel*180/pi

    end subroutine cal_declination_angle

end module Declination_angle
```

2.2 计算给定日期时间和位置的太阳小时角度

```
module Solar_hour_angle

contains

    subroutine cal_solar_hour_angle(lon,Z,day,LST,h)

        implicit none

        real(4), parameter         :: pi = 3.14
        real(4), intent(in)        :: lon, Z, LST
        integer(4), intent(in) :: day
        real(4), intent(out)      :: h
        real(4)                              :: y, EoT, LST_crt




        y = 2*pi/365*(day-1+(LST-12)/24)


        EoT = 229.18*(0.000075+0.001868*cos(y)-0.032077*sin(y)-0.014615*cos(2*y)-0.040849*sin(2*y))

        LST_crt = LST + (EoT + 4*(lon - 15*Z))/60

        h = 15*(LST_crt - 12)
        print*, 'h = ', h

    end subroutine cal_solar_hour_angle

end module Solar_hour_angle
```

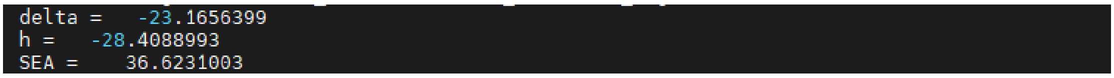
2.3

```
program Solar_elevation_angle

use Solar_hour_angle
use Declination_angle

implicit none

real(4), parameter :: pi = 3.14
real(4)                  :: lat, lon, Z, LST, h, delta, SEA
integer(4)               :: day


lat = 38.87
lon = 115.47
Z = 8
LST = 10.533333
day = 364


call cal_declination_angle(day, delta)

call cal_solar_hour_angle(lon, Z, day, LST, h)


SEA = asin(sin(lat*pi/180)*sin(delta)+cos(lat*pi/180)*cos(delta)*cos(h*pi/180))*180/pi
print*, 'SEA = ', SEA

end program Solar_elevation_angle
```

2.4

```
[ese-xied@login03 fortran_demo1]$ gfortran -c Declination_angle.f90
[ese-xied@login03 fortran_demo1]$ gfortran -c Solar_hour_angle.f90
[ese-xied@login03 fortran_demo1]$ gfortran -c Solar_elevation_angle.f90
[ese-xied@login03 fortran_demo1]$ gfortran Solar_elevation_angle.f90
```

```
Declination_angle.o Solar_hour_angle.o -o Solar_elevation_angle.x
[ese-xied@login03 fortran_demo1]$ gfortran Solar_elevation_angle.f90 -o Solar_elevation_angle.x -L. -lsea
[ese-xied@login03 fortran_demo1]$ ./Solar_elevation_angle.x
```



```
 delta =   -23.1656399
 h =   -28.4088993
 SEA =    36.6231003
```