

1.

PROGRAM Primes

IMPLICIT NONE

INTEGER :: Number, Divisor, Count, N, L, R

Count = 0

DO Number = 11, 10000, 2

Divisor = 3

DO

IF (Divisor*Divisor > Number .OR. MOD(Number,Divisor) == 0) EXIT

Divisor = Divisor + 2

END DO

IF (Divisor*Divisor > Number) THEN

L = Number

N = Number

R = 0

DO

R = R*10

R = R+MOD(N,10)

N = N/10

IF(N==0) EXIT

END DO

IF(R==L) THEN

Count = Count + 1

WRITE(*,*) 'Palindromic Prime number #', Count, ': ', Number

END IF

END IF

END DO

END PROGRAM Primes

The result of problem 1.

```
yewon@yewon-VirtualBox:~$ gfortran -ffree-form primes.f90
yewon@yewon-VirtualBox:~$ ./a.out
```

Palindromic Prime number #	1 :	11
Palindromic Prime number #	2 :	101
Palindromic Prime number #	3 :	131
Palindromic Prime number #	4 :	151
Palindromic Prime number #	5 :	181
Palindromic Prime number #	6 :	191
Palindromic Prime number #	7 :	313
Palindromic Prime number #	8 :	353
Palindromic Prime number #	9 :	373
Palindromic Prime number #	10 :	383
Palindromic Prime number #	11 :	727
Palindromic Prime number #	12 :	757
Palindromic Prime number #	13 :	787
Palindromic Prime number #	14 :	797
Palindromic Prime number #	15 :	919
Palindromic Prime number #	16 :	929

2.

PROGRAM Matrix

```
IMPLICIT NONE
REAL*8 :: ans, A1norm
INTEGER :: r,c

WRITE(*,*) 'Enter row and column.'
WRITE(*,*) 'Row: '
READ(*,*) r
WRITE(*,*) 'Col: '
READ(*,*) c
WRITE(*,*) 'Enter the matrix.'
ans = A1norm(r, c)
WRITE(*,*) 'A1norm : ', int(ans)
```

END PROGRAM Matrix

real*8 function A1norm(M, N)

```
IMPLICIT NONE
INTEGER :: i, j, M, N
real*8 A(M, N)
REAL(8) :: s
REAL y(20)

Do i = 1, M
    READ *, (A(i, j), j=1,N)
ENDDO

DO i = 1,N
    y(i) = 0
ENDDO
```

```

DO i=1, N
    DO j=1, M

        y(i) = y(i) + A(j, i)
    END DO
END DO

DO i=1, N-1
    s = max(y(i), y(i+1))
END DO

A1norm = s
RETURN

END

```

The result of problem 2.

```

yewon@yewon-VirtualBox:~$ gfortran -ffree-form test.f90
yewon@yewon-VirtualBox:~$ ./a.out
Enter row and column.
Row:
2
Col:
3
Enter the matrix.
1 2 3
4 5 6
A1norm :          9
yewon@yewon-VirtualBox:~$ ./a.out
Enter row and column.
Row:
3
Col:
3
Enter the matrix.
1 2 3
9 5 4
5 3 7
A1norm :          14

```

3.

```
PROGRAM Taylor
  IMPLICIT NONE
  DOUBLE PRECISION :: sum, x, a
  INTEGER :: k

  PRINT *, "give x: "
  READ *, x

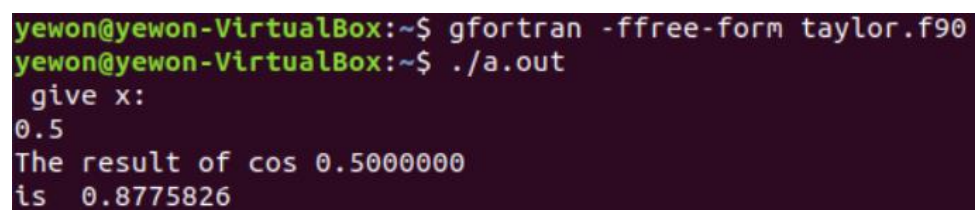
  sum = 0.0d0
  k = 0
  a = 1.0d0

  DO
    sum = sum + a
    !      next term is ...
    k = k+1
    a = a * (-x*x) / (2*k*(2*k-1))
    IF (sum + a == sum) EXIT
  ENDDO

  WRITE(*,100) 'The result of cos', x, 'is ', sum
100 FORMAT(A, F10.7)

END PROGRAM Taylor
```

The result of problem 3.



```
yewon@yewon-VirtualBox:~$ gfortran -ffree-form taylor.f90
yewon@yewon-VirtualBox:~$ ./a.out
give x:
0.5
The result of cos 0.5000000
is 0.8775826
```