

The image features a dark navy blue background with four horizontal rows of yellow zigzag patterns. The top row consists of thin lines, while the subsequent three rows are progressively thicker. In the center, a white rounded rectangle contains the word "Fortran" in a bold, yellow, sans-serif font.

Fortran

```

program Tutorial escribe y lee
implicit none
real num
write(*,*) 'Escribe un numero'
read(*,*) num
write(*,*) 'Su numero es', num
end

```

```

program Suma de dos numeros
implicit none
real num1, num2, suma, rest, mul, div, pot, raz
write(*,*) 'programa que suma resta multiplica, ',
&' Divide dos numero dados por el usuario'
write(*,*) 'Escriba un numero'
read(*,*) num1
write(*,*) 'Escriba otro numero'
read(*,*) num2
if(num2.eq.0) then
write(*,*) 'Este numero no puede ser cero'
goto 10
end if
suma=num1+num2
mul=num1*num2
rest=num1-num2
div=num1/num2
pot=num2**2
raz=sqrt(pot)
write(*,*) 'La suma es :', suma
write(*,*) 'La resta es :', rest
write(*,*) 'La multiplicacion es :', mul
write(*,*) 'La division es :', div
write(*,*) 'El cuadrado del segundo numero es :', pot
write(*,*) 'La raiz de la potencia es :', raz
stop
end

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
programa que suma resta multiplicaDivide dos numero dados por el usuario
Escriba un numero
10
Escriba otro numero
2
La suma es : 12.0000000
La resta es : 8.00000000
La multiplicacion es : 20.0000000
La division es : 5.00000000
El cuadrado del segundo numero es : 4.00000000
La raiz de la potencia es : 2.00000000

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran asteriscos.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
imprime asteriscos
Teclee el numero
3
* * *

```

```

program asteriscos
implicit none
integer i,n
write(*,*) 'imprime asteriscos'
write(*,*) 'Teclee el numero'
read(*,*) n
write(*,*)(" ", i=1,n)
stop
end

```

```

program serie de Fibonanci
implicit none
integer a0, a1, a2, term, n
write(*,*) ' serie de Fibonanci'
write(*,*) '¿Cuantos terminos de la serie desea
tener?'

```

```

read(*,*) term
a0=0
a1=1
write(*,*) a0
write(*,*) a1
do n=3,term,1
  a2=a0+a1
  write(*,*) a2
a0=a1
a1=a2
end do
read(*,*)
stop
end

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran T3.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
serie de Fibonanci
¿Cuantos terminos de la serie desea tener?
10
0
1
1
2
3
5
8
13
21
34

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran T4.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34

```

```

program Numeros
integer x,i
x=0.0d0
do i=1,100,1
  x=x+1
  write(*,*) x
end do
end

```

```

program Trans de coord
implicit none
real rad,alf,x,y
write(*,*) 'Introduzca las coordenadas polares',
&'primero r y luego alfa'

```

```

read(*,*) rad, alf
x=rad*cos(alf)
y=rad*sin(alf)
write(*,*) 'Las coordenadas cartesianas son ', x, y
read(*,*)
stop
end

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran T5.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Introduzca las coordenadas polaresprimero r y luego alfa
2
45
Las coordenadas cartesianas son 1.05064392 1.70180702

```

```

program pronostico
implicit none
integer :: temp
write(*,*)'Programa que dice que tanto calor existe'
read(*,*) temp
select case(temp)
case(:-1)
write(*,*)'Por debajo de 0 celsius'
case(0)
write(*,*)'Esta helado'
case(1:10)
write(*,*)'Hace frio'
case(11:20)
write(*,*)'Esta templado'
case(21:)
write(*,*)'Hace calor'
end select
end

```

```

hector@hector-pc ~ $ cd Documentos
hector@hector-pc ~/Documentos $ cd programacion
hector@hector-pc ~/Documentos/programacion $ cd fortran
hector@hector-pc ~/Documentos/programacion/fortran $ gfortran case.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Programa que dice que tanto calor existe
40
Hace calor

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran aprox_pi.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Aprox pi hasta
99
La aprox es 3.12159443
hector@hector-pc ~/Documentos/programacion/fortran $ gfortran aprox_pi.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Aprox pi hasta
500
La aprox es 3.13759351
hector@hector-pc ~/Documentos/programacion/fortran $ gfortran aprox_pi.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Aprox pi hasta
1000
La aprox es 3.13959336
hector@hector-pc ~/Documentos/programacion/fortran $

```

```

program numpi
implicit none
integer i,n,signo
real pi
signo=1.0d0
pi=0.0d0
write(*,*) 'Aprox pi hasta'
read(*,*) n
do i=1,n,2
pi=pi+1.0d0/i*signo
signo=-signo
end do

```

```

write(*,*) 'La aprox es ',4*pi
end program

```

```

program
implicit none
integer cout
real num,acum,resul
character(LEN=2)::seguir
do
write(*,*) 'Escribe un numero'
read(*,*) num
cout=cout+1
acum=acum+num
write(*,*) '¿Desea sumar otro numero?'
write(*,*) 'Teclea SI o NO en mayusculas y entre apostrofes'
read(*,*) seguir
if(seguir/= 'SI') exit
end do
resul=acum / cout
write(*,*) 'La Media es: ', resul
end program

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran cap3_5.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Escribe un numero
5
¿Desea sumar otro numero?
Teclea SI o NO en mayusculas y entre apostrofes
SI
Escribe un numero
5
¿Desea sumar otro numero?
Teclea SI o NO en mayusculas y entre apostrofes
SI
Escribe un numero
90
¿Desea sumar otro numero?
Teclea SI o NO en mayusculas y entre apostrofes
NO
La Media es: 33.3333321

```

```

PROGRAM cap_3_7
CHARACTER (LEN= 1) :: seguir
INTEGER :: pos
REAL : : num
DO
WRITE(*,*) 'DAME EL NUMERO DE ELEMENTOS DE LA LISTA'
READ(*,*) n
pos= 0
DO i= 1,n
WRITE(*,*) ' INTRODUCER NUMERO' ,i
READ(*,*) num
IF(num>0)THEN
pos=pos+1
END IF
END DO
WRITE(*,*) 'CANTIDAD DE POSITIVOS DE LOS' ,n,' LEIDOS ES:' ,pos
WRITE(*,*) 'QUIERES SEGUIR CON OTRA LISTA (S/N) ?'
READ(*,*) seguir
IF(seguir/='S') EXIT
END DO
END PROGRAM

```

```

program cap4_4
implicit none
integer :: num,i,c
integer vec(5)
vec(1)=1
vec(2)=1
vec(3)=6
vec(4)=6
vec(5)=7
c=0.0d0
write(*,*)'Dame un numero'
read(*,*) num
do i=1,5
if(vec(i) == num)then
write(*,*)'Primera coincidencia en la posicion',i,'del vector'
c=1.0d0
EXIT
end if
end do
if(c == 0)then
write(*,*)'No existe numero en el vector'
end if
end program

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran cap3_3.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Escribe un numero
5
¿Desea sumar otro numero?
Teclea SI o NO en mayusculas y entre apostrofes
SI
Escribe un numero
5
¿Desea sumar otro numero?
Teclea SI o NO en mayusculas y entre apostrofes
SI
Escribe un numero
90
¿Desea sumar otro numero?
Teclea SI o NO en mayusculas y entre apostrofes
NO
La Media es: 33.3333321
hector@hector-pc ~/Documentos/programacion/fortran $ gfortran cap3_7.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
DAME EL NUMERO DE ELEMENTOS DE LA LISTA
3
INTRODUCE NUMERO 1
1
INTRODUCE NUMERO 2
2
INTRODUCE NUMERO 3
3
CANTIDAD DE POSITIVOS DE LOS 3 LEIDOS ES: 3
QUIERES SEGUIR CON OTRA LISTA (S/N) ?

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran cap4_4.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Dame un numero
6
Primera coincidencia en la posicion 3 del vector

```

```

program cap4_10
implicit none
integer x,sw,i,j
integer matriz(3,5)
write(*,*)'Escribe la matriz por columnas'
read(*,*)((matriz(i,j),i=1,3),j=1,5)
sw=0.0d0
write(*,*)'Escribe un numero'
read(*,*) x
write(*,*)'La matriz es: '
do i=1,3
write(*,*)(matriz(i,j),j=1,5)
end do

do i=1,3
do j=1,5
if(x== matriz(i,j))then
sw=1
write(*,*)'Encontrado en la fila:',i,'columna',j
end if
end do
end do
if(sw==0)write(*,*)'No Existe el Valor Buscado'
end program

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran cap4_10.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Escribe la matriz por columnas
1
2
3
4
5
6
7
8
9
10
11
12
13
14
Escribe un numero
6
La matriz es:
      1      4      7      10      6
      2      5      8      11      13
      3      6      9      12      14
Encontrado en la fila:      1 columna      5
Encontrado en la fila:      3 columna      2

```

```

program e
implicit none
integer i,n
real producto,suma
producto=1
suma=1
write(*,*) 'Aprox e hasta'
read(*,*) n
do i=1,n,1
producto=producto+i
suma=suma+(1/producto)
end do

write(*,*) 'La aprox es ',suma
end program

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran aproxE.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Aprox e hasta
1000000
La aprox es 2.37323880

```

```

program decremetoclear
implicit none
integer i,j,n, aux,mayor
real num(100)
mayor=0.0d0
write(*,*) '¿Cuantos numeros desea teclear?'
read(*,*) n
!ingrese valores
do i=1,n
write(*,*) 'Ingrese un numero'
read(*,*) num(i)
enddo
do i=1,n
do j=i,n
if(num(j).lt.num(i))then
aux=num(i)
num(i)=num(j)
num(j)=aux
endif
enddo
enddo
!imprimir los nums en orden
do i=1,n
write(*,*) num(i)
enddo
end program

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran burbuja.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
¿Cuantos numeros desea teclear?
10
Ingrese un numero
9
Ingrese un numero
2
Ingrese un numero
7
Ingrese un numero
4
Ingrese un numero
3
Ingrese un numero
6
Ingrese un numero
11
Ingrese un numero
8
Ingrese un numero
1
Ingrese un numero
5
1.00000000
2.00000000
3.00000000
4.00000000
5.00000000
6.00000000
7.00000000
8.00000000
9.00000000
11.00000000

```

```

program impar y par
  implicit none
  integer x, prim
  write(*,*) 'Escribe un numero y le',
  &'dire si es par o impar'
  read(*,*) x
  prim=mod(x,2)
  if(prim.eq.0) then
    write(*,*) 'el numero es par '
    goto 10
  end if
  write(*,*) 'el numero es impar'
  read(*,*)
end

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran T7.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Escribe un numero y ledire si es par o impar
1200
el numero es par

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran T7.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Escribe un numero y ledire si es par o impar
3335
el numero es impar

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran T9.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
1
2
3
4
5
6
7
8
9
10
hector@hector-pc ~/Documentos/programacion/fortran $

```

```

program array
  implicit none
  integer array(10),i
  open(1,file='numer.dad', status='old')
  read(1,*) (array(i), i=1,10)
  do i=1,10
    write(*,*)array(i)
  end do
  close(1)
end

```

```

program Numeros
  implicit none
  integer a(3,3),i,j
  do i=1,3
    write(*,*) 'Escribe la fila de la matriz'
    read(*,*) (a(i,j), j=1,3)
  end do
  do i=1,3
    write(*,*) (a(i,j), j=1,3)
  end do
end

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran M1.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Escribe la fila de la matriz
1
2
3
Escribe la fila de la matriz
4
5
6
Escribe la fila de la matriz
1
2
2
1 2 3
4 5 6
1 2 2

```



```

program Cap4_11
integer i,j
real mod
real v1(3),v2(3),v3(3),s(3)
write(*,*)'Teclee el Vector 1'
do i=1,3
write(*,*)'Posicion ',i
read(*,*) v1(i)
end do
write(*,*)'Teclee el Vector 2'
do i=1,3
write(*,*)'Posicion ',i
read(*,*) v2(i)
end do
write(*,*)'Teclee el Vector 3'
do i=1,3
write(*,*)'Posicion ',i
read(*,*) v3(i)
end do
write(*,*)'Vector Suma'
do i=1,3
s(i)=v1(i)+v2(i)+v3(i)
write(*,*) 'Componente',i,s(i)
end do
write(*,*)'Vector Suma es'

write(*,*)(s(i),i=1,3)
mod=0.0d0
do i=1,3
mod=mod+s(i)**2
end do
mod=SQRT(mod)
write(*,*)'El modulo del Vector suma es',mod

end program

```

```

hector@hector-pc ~/Documentos/prog $ gfortran cap4_11.f
hector@hector-pc ~/Documentos/prog $ ./a.out
Teclee el Vector 1
Posicion      1
1
Posicion      2
2
Posicion      3
3
Teclee el Vector 2
Posicion      1
1
Posicion      2
2
Posicion      3
3
Teclee el Vector 3
Posicion      1
1
Posicion      2
1
Posicion      3
1
Vector Suma
Componente     1   3.00000000
Componente     2   5.00000000
Componente     3   7.00000000
Vector Suma es
3.00000000    5.00000000    7.00000000
El modulo del Vector suma es  9.11043358

```

```

program mult_matrices
implicit none
integer A(3,3), B(3,3),C(3,3)
integer i,j,k
C=0.
call Matriz(A,B,C)
end program

```

```

subroutine Matriz(A,B,C)
implicit none
integer A(3,3), B(3,3), C(3,3)
integer i,j,k
C=.0

```

```

write(*,*) 'Matriz 1'
open(10,file='matriz1.dad', status='old')
do i=1,3
  read(10,*) (A(i,j), j=1,3)
end do
do i=1,3
  write(*,*) (A(i,j), j=1,3)
end do
close(10)
write(*,*)
write(*,*) 'Matriz 2'
open(11,file='matriz2.dad', status='old')
do i=1,3
  read(11,*) (B(i,j), j=1,3)
end do
do i=1,3
  write(*,*) (B(i,j), j=1,3)
end do
close(11)
write(*,*)
write(*,*) 'El producto de ',
&'Matriz 1 y la matriz 2 es'
write(*,*)

```

```

do i=1,3
  do j=1,3
    do k=1,3
      C(i,j)=C(i,j)+(A(i,k)*B(k,j))
    end do
  end do
end do
do i=1,3
  write(*,*) (C(i,j),j=1,3)
end do
end subroutine

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran multdematrices.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Matriz 1
  1      2      7
  3      4      8
  5      6      9

Matriz 2
  9      8      7
  6      5      4
  3      2      1

El producto de Matriz 1 y la matriz 2 es
  42     32     22
  75     60     45
 108     88     68

```

matriz1.dad x	multdematrices.f x	matriz2.dad
1 2 7	9 8 7	
3 4 8	6 5 4	
5 6 9	3 2 1	

```

program tranpuesta
implicit none
integer i,j
integer tmatriz(3,3)
write(*,*) 'Matriz Trapuesta de 3x3'
write(*,*) 'teclea la matriz por filas'
tmatriz=.0
do i=1,3
read(*,*) (tmatriz(i,j), j=1,3)
enddo
write(*,*) 'La Matriz es:'
do i=1,3
write(*,*) (tmatriz(i,j), j=1,3)
enddo
write(*,*) 'La Transpuesta es:.'
do i=1,3
write(*,*) (tmatriz(j,i), j=1,3)
enddo
end

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran transp.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Matriz Trapuesta de 3x3
teclea la matriz por filas
1
2
3
4
5
6
7
8
9
La Matriz es:
1 2 3
4 5 6
7 8 9
La Transpuesta es:
1 4 7
2 5 8
3 6 9

```

```

program multiplicacion
implicit none
integer i,j,k
integer A(3,3),B(3,3), S(3,3)
write(*,*) 'multiplicacion de Matrices'
write(*,*) 'teclea la matriz A por filas'
A=.0
B=.0
S=.0

```

!Teclear Una Matriz

```
do i=1,3
```

```
read(*,*) (A(i,j), j=1,3)
```

```
enddo
```

!Teclear otra Matriz

```
write(*,*) 'teclea la matriz B por filas'
```

```
do i=1,3
```

```
read(*,*) (B(i,j), j=1,3)
```

```
enddo
```

!Imprime en Pantalla La Primer Matriz

```
write(*,*) 'La Matriz A es:'
```

```
do i=1,3
```

!Imprime en Pantalla La segunda Matriz

```
write(*,*) (A(i,j), j=1,3)
```

```
enddo
```

```
write(*,*) 'La Matriz B es:'
```

```
do i=1,3
```

```
write(*,*) (B(i,j), j=1,3)
```

```
enddo
```

!Codigo De multiplicacion

```
do i=1,3
```

```
do j=1,3
```

```
do k=1,3
```

```
S(i,j)=S(i,j)+(A(i,k)*B(k,j))
```

```
enddo
```

```
enddo
```

```
enddo
```

!Imprimir el Codigo de la multiplicacion

```
write(*,*) 'La multiplicacion de A*B'
```

```
do i=1,3
```

```
write(*,*) (S(i,j), j=1,3)
```

```
enddo
```

```
end
```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran multMat.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
multiplicacion de Matrices
teclea la matriz A por filas
1
2
3
4
5
6
7
8
9
teclea la matriz B por filas
9
8
7
6
5
4
3
2
1
La Matriz A es:
1 2 3
4 5 6
7 8 9
La Matriz B es:
9 8 7
6 5 4
3 2 1
La multiplicacion de A*B
30 24 18
84 69 54
138 114 90

```

```

program multiplicacion
implicit none
integer i,j,k
integer A(3,3),B(3,3), S(3,3),M(3,3),R(3,3)
write(*,*) 'Operaciones con Matrices'
write(*,*) 'teclea la matriz A por filas'
A=.0
B=.0
S=.0
R=.0
M=.0
!Teclear Una Matriz
do i=1,3
read(*,*) (A(i,j), j=1,3)
enddo
!Teclear otra Matriz
write(*,*) 'teclea la matriz B por filas'
do i=1,3
read(*,*) (B(i,j), j=1,3)
enddo
!Imprime en Pantalla La Primer Matriz
write(*,*) 'La Matriz A es:'
do i=1,3
!Imprime en Pantalla La segunda Matriz
write(*,*) (A(i,j), j=1,3)
enddo
write(*,*) 'La Matriz B es:'
do i=1,3
write(*,*) (B(i,j), j=1,3)
enddo
!Codigo Suma de Matrices
do i=1,3
do j=1,3
S(i,j)=(A(i,j))+B(i,j))
enddo
enddo
!resta de Matrices
do i=1,3
do j=1,3
R(i,j)=(A(i,j))-B(i,j))
enddo
enddo
!Codigo De multiplicacion
do i=1,3
do j=1,3
do k=1,3
M(i,j)=M(i,j)+(A(i,k)*B(k,j))
enddo
enddo
enddo

```

```

!Imprimir el Codigo de la suma
write(*,*) 'La Suma de A+B'
do i=1,3
write(*,*) (S(i,j), j=1,3)
enddo
!Imprimir el Codigo de la multiplicacion
write(*,*) 'La resta de A-B'
do i=1,3
write(*,*) (R(i,j), j=1,3)
enddo
!Imprimir el Codigo de la multiplicacion
write(*,*) 'La multiplicacion de A*B'
do i=1,3
write(*,*) (M(i,j), j=1,3)
enddo
end

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran multMat.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Operaciones con Matrices
teclea la matriz A por filas
1
2
3
4
5
6
7
8
9
teclea la matriz B por filas
1
2
3
4
5
6
7
8
9
La Matriz A es:
1 2 3
4 5 6
7 8 9
La Matriz B es:
1 2 3
4 5 6
7 8 9
La Suma de A+B
10 10 10
10 10 10
10 10 10
La resta de A-B
-8 -6 -4
-2 0 2
4 6 8
La multiplicacion de A*B
30 24 18
84 69 54
138 114 90

```

```

program multiplicacion
implicit none
real A(100000,100000),B(100000,100000), M(100000,100000)
integer i,j,k,r,c, re,co
write(*,*) 'Operaciones con Matrices'
write(*,*) 'las filas de A son'
read(*,*) r
write(*,*) 'las columnas de A son'
read(*,*) c
write(*,*) 'las filas de B son'
read(*,*) re
write(*,*) 'las columnas de B son'
read(*,*) co
write(*,*) 'teclea la matriz A por filas'
M=.0
!Teclear Una Martiz
do i=1,r
read(*,*) (A(i,j), j=1,c)
enddo
!Teclear otra Matriz
write(*,*) 'teclea la matriz B por filas'
do i=1,re
read(*,*) (B(i,j), j=1,co)
enddo
!Imprime en Pantalla La Primer Matriz
write(*,*) 'La Matriz A es:'
do i=1,r
write(*,*) (A(i,j), j=1,c)
enddo
!Imprime en Pantalla La segunda Matriz
write(*,*) 'La Matriz B es:'
do i=1,re
write(*,*) (B(i,j), j=1,co)
enddo
!Codigo De multiplicacion
do i=1,r
do j=1,co
do k=1,c
M(i,j)=M(i,j)+(A(i,k)*B(k,j))
enddo
enddo
enddo
!Imprimir el Codigo de la multiplicacion
write(*,*) 'La multiplicacion de A*B'
do i=1,r
write(*,*) (M(i,j), j=1,co)
enddo
end

```

```

hector@hector-pc ~/Documentos/programacion/fortran $ gfortran mm.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
Operaciones con Matrices
las filas de A son
3
las columnas de A son
3
las filas de B son
3
las columnas de B son
3
teclea la matriz A por filas
1
2
3
4
5
6
7
8
9
teclea la matriz B por filas
9
8
7
6
5
4
3
2
1
La Matriz A es:
1.00000000  2.00000000  3.00000000
4.00000000  5.00000000  6.00000000
7.00000000  8.00000000  9.00000000
La Matriz B es:
9.00000000  8.00000000  7.00000000
6.00000000  5.00000000  4.00000000
3.00000000  2.00000000  1.00000000
La multiplicacion de A*B
30.0000000  24.0000000  18.0000000
84.0000000  69.0000000  54.0000000
138.000000  114.000000  90.0000000

```

Funciones y subrutinas

```
program Suma de dos numeros
real r
call s(r)
end program
```

```
subroutine s(suma)
implicit none
real num1, num2, suma
write(*,*) 'programa Funcion de Suma'
write(*,*) 'Escriba un numero'
read(*,*) num1
write(*,*) 'Escriba otro numero'
read(*,*) num2
suma=num1+num2
write(*,*) 'La suma es : ',suma
return
end
```

```
hector@hector-pc ~/Documentos/programacion/fortran $ gfortran F1.f
hector@hector-pc ~/Documentos/programacion/fortran $ ./a.out
programa Funcion de Suma
Escriba un numero
5
Escriba otro numero
5
La suma es : 10.0000000
```

```
hector@hector-pc ~/Documentos/prog $ gfortran cap5_1.f
hector@hector-pc ~/Documentos/prog $ ./a.out
Dame dos numeros
64
36
La suma es 100
```

```
program cap5_1
implicit none
integer a,b,suma
write(*,*) 'Dame dos numeros'
read(*,*) a,b
write(*,*) 'La suma es',suma(a,b)
end program
```

```
integer function suma(x,y)
implicit none
integer x,y
suma=x+y
end function
```

```
program cap5_4
implicit none
integer a,b,s
write(*,*) 'escribe dos numeros'
read(*,*) a,b
call suma(a,b,s)
write(*,*) 'La suma es',s
end program
```

```
subroutine suma(x,y,z)
implicit none
integer x,y,z
z=x+y
end subroutine
```

```
hector@hector-pc ~/Documentos/prog $ gfortran cap5_4.f
hector@hector-pc ~/Documentos/prog $ ./a.out
escribe dos numeros
64
36
La suma es 100
```

```

program cap5_3
implicit none
character(LEN=5)::seguir
integer fac,i,n
real sumador
do
write(*,*)'Numero de elementos de la suma'
read(*,*) n
sumador=0.0d0
do i=1,n
sumador=sumador+1./fac(i)
end do
write(*,*)'el resultado es :',sumador
write(*,*)'Desea continuar (S/N)'
read(*,*)seguir
if(seguir/='S')Exit
end do
end program

```

```

hector@hector-pc ~/Documentos/prog $ gfortran cap5_3.f
hector@hector-pc ~/Documentos/prog $ ./a.out
Numero de elementos de la suma
5
el resultado es : 1.71666658
Desea continuar (S/N)
S
Numero de elementos de la suma
10000
el resultado es : Infinity
Desea continuar (S/N)
N

```

```

function fac(x)
implicit none
integer x,fac,i
fac=1.0d0
do i=1,x
fac=fac*i
end do
end function

```

```

program cap5_3
implicit none
character(LEN=5)::seguir
integer fac,i,n
real sumador
do
write(*,*)'Numero de elementos de la suma'
read(*,*) n
sumador=0.0d0
do i=1,n
sumador=sumador+1./fac(i)
end do
write(*,*)'el resultado es :',sumador
write(*,*)'Desea continuar (S/N)'
read(*,*)seguir
if(seguir/='S')Exit
end do
end program

```

```

hector@hector-pc ~/Documentos/prog $ gfortran cap5_3.f
hector@hector-pc ~/Documentos/prog $ ./a.out
Numero de elementos de la suma
5
el resultado es : 1.71666658
Desea continuar (S/N)
S
Numero de elementos de la suma
10000
el resultado es : Infinity
Desea continuar (S/N)
N

```

```

function fac(x)
implicit none
integer x,fac,i
fac=1.0d0
do i=1,x
fac=fac*i
end do
end function

```

```

program cap5_6
implicit none
integer a,b
a=5.0d0
b=10.0d0
write(*,*)'Antes del intercambio'
write(*,*)'A=',a,'B=',b
call cambia(a,b)
write(*,*)'Despues del cambio en principal'
write(*,*)'A=',a,'B=',b
end program

```

```

subroutine cambia(x,y)
implicit none
integer x, y, aux
aux=x
x=y
y=aux
write(*,*)'Despues del intercambio en subrutina'
write(*,*)'X=',x,'Y=',y
end subroutine

```

```

hector@hector-pc ~/Documentos/prog $ gfortran cap5_6.f
hector@hector-pc ~/Documentos/prog $ ./a.out
Antes del intercambio
A=      5 B=     10
Despues del intercambio en subrutina
X=     10 Y=      5
Despues del cambio en principal
A=     10 B=      5

```