

Control de Fortran

Crea un archivo de texto chamado `vector_matriz.dat` co seguinte contido:

```
4
2 7 1 9
3 1 5 6
2 4 10 7
1 9 5 6
7 3 1 8
```

Escribe un programa en Fortran chamado `exame.f90` que abra este arquivo e lea o número da primeira liña á variábel n . Logo, debe reservar memoria para un vector \mathbf{x} de lonxitude n , e para dúas matrices \mathbf{a} e \mathbf{b} cadradas de orde n , ambas enteiras. A segunda liña do arquivo debe lerse ao vector \mathbf{x} , e as 4 liñas restantes deben lerse á matriz \mathbf{a} . O programa debe invocar a un subprograma chamado `calcula_b(...)`, do tipo e cos argumentos axeitados, que calcule a matriz \mathbf{b} con elementos b_{ij} , con $i, j = 1, \dots, n$, de modo que b_{ij} sexa o número de elementos da columna j da matriz \mathbf{a} que hai que sumar para superar x_i . Se non se chega a cumprir a condición debe ser $b_{ij} = n$. Finalmente, o programa principal debe mostrar por pantalla a matriz \mathbf{b} , con formato enteiro de ancho 8.

SOLUCIÓN:

```
program exame36
integer , dimension (:), allocatable :: x
integer , dimension (:,:), allocatable :: a,b
open(1, file='vector_matriz.dat', status='old', err=1)
read (1,*) n
allocate(x(n), a(n,n), b(n,n))
read (1,*) x
do i=1,n
  read (1,*) (a(i,j), j=1,n)
end do
close(1)
call calcula_b(x,a,b,n)
print *, 'matriz b:'
do i=1,n
  do j=1,n
    print '(i8," ",$)', b(i,j)
  end do
  print *, ''
end do
deallocate(x,a,b)
stop
1 stop 'erro: vector_matriz.dat non existe'
end program exame36
```

```
!-----
subroutine calcula_b(x,a,b,n)
integer , dimension(n), intent(in) :: x
integer , dimension(n,n), intent(in) :: a
integer , dimension(n,n), intent(out) :: b
integer , intent(in) :: n
do i=1,n
  do j=1,n
    s=0
    do k=1,n
      s=s+a(k,j)
      if(s>=x(i)) exit
    end do
  end do
end do
```

```
        b(i,j)=k
    end do
end do
return
end subroutine calcula_b
```