

Control de Fortran

Escribe un programa en Fortran chamado `exame.f90` que lea un número enteiro n e logo n valores enteiros, que debes almacenar nun vector \mathbf{v} de lonxitude n . O programa debe crear un arquivo novo chamado `saida.txt`, e chamar repetidamente a un subprograma `reduce(...)`, do tipo e cos argumentos axeitados, que transforme o vector \mathbf{v} de modo que:

$$v_i = \begin{cases} v_i + v_{n-i+1} & 1 \leq i \leq m \\ v_{i-m} + v_i & m < i \leq n \end{cases} \quad (1)$$

onde $m = \lfloor n/2 \rfloor$. A chamada ao subprograma repetirase ata que a suma dos elementos do vector \mathbf{v} sexa maior que 100. En cada iteración, o programa debe engadir o vector \mathbf{v} calculado ao final do arquivo `saida.txt` (o vector \mathbf{v} debe ir nunha liña distinta en cada iteración). Proba con $n = 5$ e $\mathbf{v}=(1,2,3,4,5)$. Antes de cada execución (agás a primeira), on esquezas borrar o arquivo `saida.txt` co comando `rm -f saida.txt`. Tes que obter o seguinte arquivo `saida.txt`.

```
6 6 9 10 14
20 16 29 26 43
```

SOLUCIÓN:

```
program exame
integer , dimension (:), allocatable :: v
print '( "n? ", $ )'; read *, n
allocate (v(n)); m=n/2
print '( "v? ", $ )'; read *, v
open(1, file='saida.txt', status='new', err=1)
do
  call reduce(v,m,n)
  write (1,*) v
  if(sum(v)>100) exit
end do
close(1)
deallocate(v)
stop
! print *, 'erro: saida.txt xa existe'
stop
end program exame
```

```
subroutine reduce(v,m,n)
integer , dimension(n), intent(inout) :: v
integer , intent(in) :: n,m
do i=1,m
  v(i)=v(i)+v(n-i+1)
end do
do i=m+1,n
  v(i)=v(i-m)+v(i)
end do
return
end subroutine reduce
```