

## Assignment 4

1. To the previous assignment with `printnice_r`, add function `printnice_i` that prints integer matrices. Overload so that `printnice` prints both integer and real matrices. Test.
2. The program contains the data on animals in structure `animal`. Create a subroutine that given a variable of type `animal` will print the information on that animal.

```
module animals
type calf
  character (10) :: id
  integer :: year_of_birth
  character:: sex
end type
end module

program registration
use animals
implicit none
type (calf) :: a,b,c

a=calf('small',1997,'M')
! b is also a male born in 1997 but has a different name
b=a
b%id='large'

! print details on both animals

print*, 'first animal'
print*, 'name:',a%id
print*, 'year of birth:',a%year_of_birth
print*, 'sex:',a%sex

print*, 'second animal'
print*, 'name:',b%id
print*, 'year of birth:',b%year_of_birth
print*, 'sex:',b%sex
end
```

Replace the print statements above by subroutine `print_am`, as below:

```
call print_am(a)
call print_am(b)
```

## Optional

3. This assignment tests data structures for sparse vectors, operations on data vectors, and operator overloading. All definitions and procedures need to reside in module `sparse_vec`. Write:

- a) data structure for sparse vector
- b) subroutine that creates sparse vector from a dense vector
- c) subroutine that prints a sparse vector
- d) subroutine that multiplies two sparse vectors.

Provide interface to c) with “=” operator

Provide interface to d) with “+” operator

To the last assignment, add interfaces to “+” and “\*” that allow to add or multiply sparse vector by a regular vector.