Programming and Modelling (week 37)

C. Thieulot

Institute of Earth Sciences

September 2017

The if-then-else construction (1)

```
program factorial
implicit none
integer :: fact
integer :: i,n
write(6,*) 'enter a number'
read(5,*) n
fact=1
do i=1,n
   fact=fact*i
  write(6,*) i,'! =',fact
end do
end program
```

The if-then-else construction (1)

```
program factorial
implicit none
integer :: fact
integer :: i,n
write(6,*) 'enter a number'
read(5,*) n
fact=1
do i=1.n
  fact=fact*i
  write(6,*) i,'! =',fact
end do
end program
```

 $\rightarrow n!$ exists for n > 0 only

The if-then-else construction (1)

```
program factorial
implicit none
integer :: fact
integer :: i,n
write(6,*) 'enter a number'
read(5,*) n
fact=1
do i=1.n
  fact=fact*i
  write(6,*) i,'! =',fact
end do
end program
```

- $\rightarrow n!$ exists for n > 0 only
- \rightarrow the computed n! makes sense up to n=12

The if-then-else construction (2)

```
program factorial
implicit none
integer :: fact
integer :: i,n
write(6,*) 'enter a number'
read(5,*) n
if (n>0 .and. n<13) then
   fact=1
   do i=1,n
     fact=fact*i
     write(6,*) i,'! =',fact
   end do
else
   write(6,*) 'the input value of n'
   write(6,*) 'is not correct. Aborting.'
end if
end program
```

```
thebeast:progmod geogarfield$ ./a.out
enter a number
           2!=
thebeast:progmod geogarfield$ ./a.out
enter a number
                         120
                         720
                        5040
thebeast:progmod geogarfield$ ./a.out
 enter a number
 the input value of n
 is not correct. Aborting.
thebeast:progmod geogarfield$ ./a.out
 enter a number
15
 the input value of n
 is not correct. Aborting.
```

The if-then-else construction (3)

The if-then-else construction can be extended as follows:

```
IF (x > 0) THEN
    WRITE(*,*) '+'
ELSE IF (x == 0) THEN
    WRITE(*,*) '0'
ELSE
    WRITE(*,*) '-'
END IF
```

The if-then-else construction (3)

The if-then-else construction can be extended as follows:

```
IF (x > 0) THEN
  WRITE(*,*) '+'
ELSE IF (x == 0) THEN
   WRITE(*,*) '0'
FL SF
 WRITE(*,*) '-'
END IF
INTEGER
CHARACTER(LEN=1) :: Grade
IF (x < 50) THEN
   Grade = 'F'
ELSE IF (x < 60) THEN
   Grade = 'D'
ELSE IF (x < 70) THEN
   Grade = 'C'
ELSE IF (x < 80) THEN
   Grade = 'B'
ELSE
   Grade = 'A'
END IF
```

The if-then-else construction (4)

Write a small fortran program which computes the smallest of three numbers.

The if-then-else construction (4)

Write a small fortran program which computes the smallest of three numbers.

```
program smallest
implicit none
real :: x,v,z,smallest nb
write(6,*) 'enter first number'
read(5,*) x
write(6,*)'enter second number'
read(5,*) y
write(6,*)'enter third number'
read(5,*) z
if (x < y .and. x < z) then
   smallest nb=x
else if (y < x .and. y < z) then
   smallest nb=v
else
   smallest nb=z
end if
write(6,*) 'the smallest number is', smallest nb
end program
```

The if-then-else construction (4)

$$ax^2 + bx + c = 0$$

Write a program which takes as input three numbers a, b, c and returns the solutions (if they exist) of the equation

relational operators

The relational operators act upon numbers:

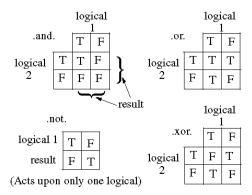
- == , is equal to
- < , is less than
- <= , is less than or equal to
- /= , is not equal to
- > , is greater than
- >= , is greater than or equal to

Logical operators

There are also logical operators, used only between complete logical expressions.

- .not., Changes the value of the expression to the opposite value.
- .and., Conditional is true only if both expressions are true.
- .or., Conditional is false only if both expressions are false.
- .xor., Exclusive or conditional is true if only one expression is true.

Truth Tables



if-then-else in other languages

C++ language

```
if (Condition_1)
{
    // Statement_1;
}
else if (Condition_2)
{
    // Statement_2;
}
else if (Condition_3)
{
    // Statement_3;
}
else
{
    // Statement_n;
}
```

Matlab

```
% Preallocate a matrix
nrows = 10:
ncols = 10:
myData = ones(nrows, ncols);
% Loop through the matrix
for r = 1:nrows
   for c = 1:ncols
      if r == c
         myData(r,c) = 2;
      elseif abs(r - c) == 1
         myData(r,c) = -1;
      else
         myData(r,c) = 0;
      end
   end
end
```

2D geometry

Write a program which reads the coordinates of three points defining a triangle, and the coordinates of a fourth point M.

- Compute whether M is in the triangle ABC
- Compute whether M is in the circumscribed circle of triangle ABC
- ► Compute whether M is in the inscribed circle of triangle ABC