

# Reporte: Actividad 3

Valenzuela Carrillo María Inés

20 de Febrero de 2015

## 1 Fortran

Como parte de la tercera actividad para el curso de programación se compilieron diversos programas en Fortran. A continuación se describen cada uno de ellos acompañados de un ejemplo del programa en código Fortran y una foto al momento de compilarlos.

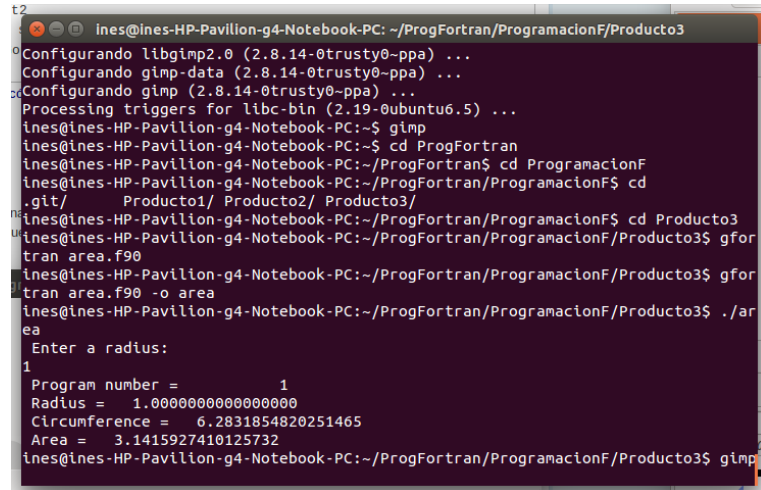
### 1.1 Área

En primer programa se pretende calcular el área de un círculo mediante la entrada de un radio.

#### 1.1.1 Código en Fortran

```
! Area . f90 : Calculates the area of a circle, sample program
! -----
Program area_circulo ! Begin main program
Implicit None ! Declare all variables
Real *8 :: radius , circum , area ! Declare Reals
Real *8 :: PI = 4.0 * atan(1.0) ! Declare , assign Real
Integer :: model_n = 1 ! Declare , assign Ints
print * , 'Enter a radius:' ! Talk to user
read * , radius ! Read into radius
circum = 2.00 * PI * radius ! Calc circumference
area = radius * radius * PI ! Calc area
print * , 'Program number =' , model_n ! Print program number
print * , 'Radius =' , radius ! Print radius
print * , 'Circumference =' , circum ! Print circumference
print * , 'Area =' , area ! Print area
End Program area_circulo ! End main program code
```

### 1.1.2 Captura en la Terminal



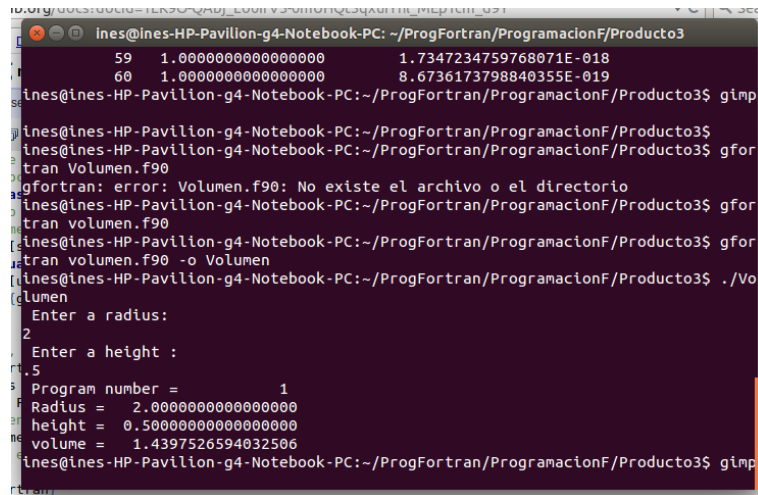
```
ines@ines-HP-Pavilion-g4-Notebook-PC: ~/ProgFortran/ProgramacionF/Producto3
Configurando libgimp2.0 (2.8.14-0trusty0~ppa) ...
Configurando gimp-data (2.8.14-0trusty0~ppa) ...
Configurando gimp (2.8.14-0trusty0~ppa) ...
Processing triggers for libc-bin (2.19-0ubuntu6.5) ...
ines@ines-HP-Pavilion-g4-Notebook-PC:~$ gimp
ines@ines-HP-Pavilion-g4-Notebook-PC:~$ cd ProgFortran
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran$ cd ProgramacionF
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF$ cd
.git/ Producto1/ Producto2/ Producto3/
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF$ cd Producto3
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfor
tran area.f90
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfor
tran area.f90 -o area
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ ./ar
ea
Enter a radius:
1
Program number =          1
Radius =  1.0000000000000000
Circumference =  6.2831854820251465
Area =  3.1415927410125732
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gimp
```

## 1.2 Volumen

El segundo programa es una modificación del primero, en este caso en lugar de calcular el área se calculará el volumen de una esfera con un líquido dentro, dicho volumen dependerá del radio y de la altura del líquido.

```
! Volumen . f90 : Calcular el volumen
! -----
Program Volumen_circulo ! Begin main program
Implicit None ! Declare all variables
Real *8 :: radius , height , volume ! Declare Reals
Real *8 :: PI = 4.0 * atan(1.0) ! Declare , assign Real
Integer :: model_n = 1 ! Declare , assign Ints
print * , 'Enter a radius:' ! Talk to user
read * , radius ! Read into radius
print * , 'Enter a height :' ! Talk to user
read * , height ! Read into height
volume = 0.3333 * PI * height * height * (3 * radius - height) ! Calc volume
print * , 'Program number =' , model_n ! Print program number
print * , 'Radius =' , radius ! Print radius
print * , 'height =' , height ! Print height
print * , 'volume =' , volume ! Print volume
End Program Volumen_circulo ! End main program code
```

### 1.2.1 Captura en la Terminal



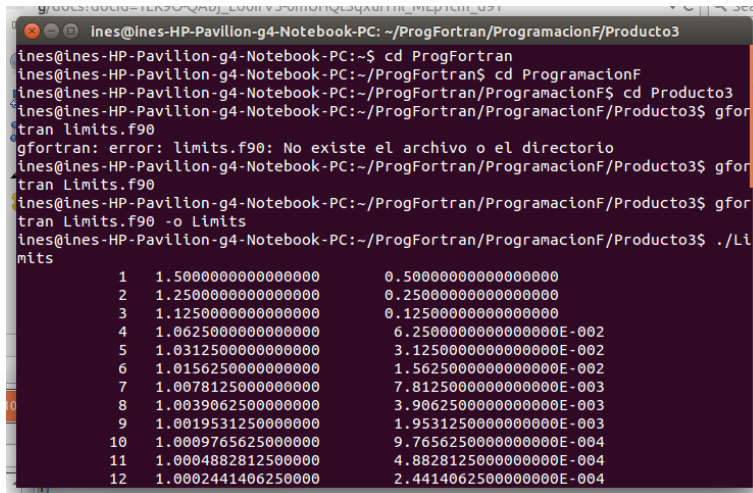
```
ines@ines-HP-Pavilion-g4-Notebook-PC: ~/ProgFortran/ProgramacionF/Producto3
59 1.0000000000000000 1.7347234759768071E-018
60 1.0000000000000000 8.6736173798840355E-019
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gimp
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfor
tran Volumen.f90
gfortran: error: Volumen.f90: No existe el archivo o el directorio
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfor
tran volumen.f90
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfor
tran volumen.f90 -o Volumen
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ ./Vo
lumen
Enter a radius:
2
Enter a height :
0.5
Program number = 1
Radius = 2.0000000000000000
height = 0.5000000000000000
volume = 1.4397526594032506
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gimp
```

## 1.3 Precisión

La tercera actividad consistió en determinar la precisión del ordenador, utilizando precisión doble.

```
! Limits . f90 : Determines machine precision
! -----
Program Limits
Implicit None
Integer :: i , n
Real *8 :: epsilon_m , one
n=60 ! Establish the number of iterations
! Set initial values :
epsilon_m = 1.0
one = 1.0
! Within a DO-LOOP, calculate each step and print .
! This loop will execute 60 times in a row as i is
! incremented from 1 to n ( since n = 60) :
do i = 1, n , 1 ! Begin the do-loop
epsilon_m = epsilon_m / 2.0 ! Reduce epsilon m
one = 1.0 + epsilon_m ! Re-calculate one
print * , i , one , epsilon_m ! Print values so far
end do ! End loop when i>n
```

### 1.3.1 Captura en la Terminal



```
ines@ines-HP-Pavilion-g4-Notebook-PC: ~/ProgFortran/ProgramacionF/Producto3
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran$ cd ProgramacionF
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF$ cd Producto3
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfortran Limits.f90
gfortran: error: Limits.f90: No existe el archivo o el directorio
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfortran Limits.f90 -o Limits
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ ./Limits
1 1.5000000000000000 0.5000000000000000
2 1.2500000000000000 0.2500000000000000
3 1.1250000000000000 0.1250000000000000
4 1.0625000000000000 6.250000000000000E-002
5 1.0312500000000000 3.125000000000000E-002
6 1.0156250000000000 1.562500000000000E-002
7 1.0078125000000000 7.812500000000000E-003
8 1.0039062500000000 3.906250000000000E-003
9 1.0019531250000000 1.953125000000000E-003
10 1.0009765625000000 9.765625000000000E-004
11 1.0004882812500000 4.882812500000000E-004
12 1.0002441406250000 2.441406250000000E-004
```

## 1.4 Precisión 4

En esta actividad se hizo lo mismo que en la anterior, pero ahora con precisión sencilla.

```
! Limits . f90 : Determines machine precision
! -----
Program Limits
  Implicit None
  Integer :: i , n
  Real *4 :: epsilon_m , one
  n=60 ! Establish the number of iterations
  ! Set initial values :
  epsilon_m = 1.0
  one = 1.0
  ! Within a DO-LOOP, calculate each step and print .
  ! This loop will execute 60 times in a row as i is
  ! incremented from 1 to n ( since n = 60 ) :
  do i = 1, n , 1 ! Begin the do-loop
    epsilon_m = epsilon_m / 2.0 ! Reduce epsilon m
    one = 1.0 + epsilon_m ! Re-calculate one
    print *, i , one , epsilon_m ! Print values so far
  end do ! End loop when i>n
End Program Limits
```

### 1.4.1 Captura en la Terminal

```
ines@ines-HP-Pavilion-g4-Notebook-PC: ~/ProgFortran/ProgramacionF/Producto3
Radius = 2.0000000000000000
height = 0.5000000000000000
volume = 1.4397526594032506
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gimp
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfor
tran Limits4.f90
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfor
tran Limits4.f90 -o Limits4
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ ./Li
mits4
1 1.50000000 0.50000000
2 1.25000000 0.25000000
3 1.12500000 0.12500000
4 1.06250000 6.25000000E-02
5 1.03125000 3.12500000E-02
6 1.01562500 1.56250000E-02
7 1.00781250 7.81250000E-03
8 1.00390625 3.90625000E-03
9 1.00195312 1.95312500E-03
10 1.00097656 9.76562500E-04
11 1.00048828 4.88281250E-04
12 1.00024414 2.44140625E-04
13 1.00012207 1.22070312E-04
14 1.00006104 6.10351562E-05
```

## 1.5 Math

Se muestran distintas funciones con las cuales se puede trabajar en fortran.

```
! Math . f90 : demo some Fortran math functions
! -----
Program Mathtest ! Begin main program
  Real *8 :: x = 1.0 , y, z ! Declare variables x, y, z
  y = sin (x) ! Call the sine function
  z = exp (x) + 1.0 ! Call the exponential function
  print * , x, y, z ! Print x, y, z
End Program Mathtest ! End main program
```

### 1.5.1 Captura en la Terminal

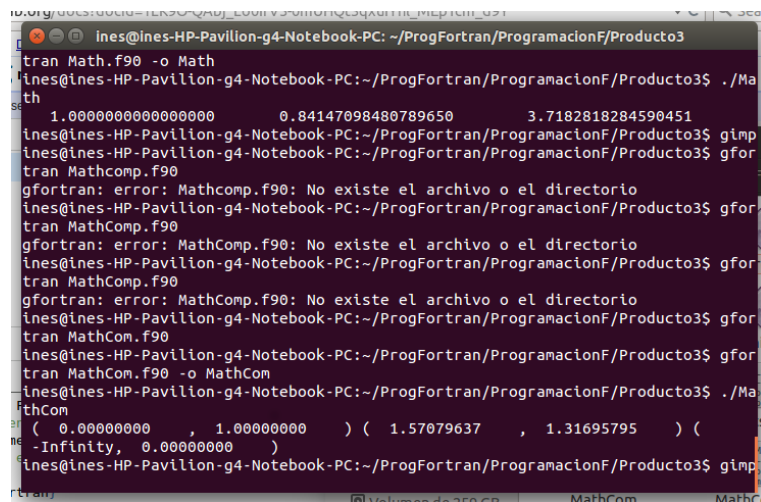
```
ines@ines-HP-Pavilion-g4-Notebook-PC: ~/ProgFortran/ProgramacionF/Producto3
47 1.00000000 7.10542736E-15
48 1.00000000 3.55271368E-15
49 1.00000000 1.77635684E-15
50 1.00000000 8.88178420E-16
51 1.00000000 4.44089210E-16
52 1.00000000 2.22044605E-16
53 1.00000000 1.11022302E-16
54 1.00000000 5.55111512E-17
55 1.00000000 2.77555756E-17
56 1.00000000 1.38777878E-17
57 1.00000000 6.93889390E-18
58 1.00000000 3.46944695E-18
59 1.00000000 1.73472348E-18
60 1.00000000 8.67361738E-19
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gimp
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfor
tran Math.f90
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfor
tran Math.f90 -o Math
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ ./Ma
th
1.0000000000000000 0.84147098480789650 3.7182818284590451
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gimp
```

## 1.6 Math Complex

Al igual que en la actividad anterior se muestran funciones, pero que arrojan resultados con números complejos.

```
! Math . f90 : demo some Fortran math functions
! -----
Program Math2! Begin main program
Complex *8 :: x=- 1.0 , y=2, z=0 ! Declare variables x, y, z
x = sqrt (x)
y = asin (y) ! Call the sine function
z = log (z) ! Call the exponential function
print * , x, y, z ! Print x, y, z
End Program Math2 ! End main program
```

### 1.6.1 Captura en la Terminal



```
ines@ines-HP-Pavilion-g4-Notebook-PC: ~/ProgFortran/ProgramacionF/Producto3
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ ./Math
1.0000000000000000 0.84147098480789650 3.7182818284590451
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gimp
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfortran MathComp.f90
gfortran: error: MathComp.f90: No existe el archivo o el directorio
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfortran MathComp.f90
gfortran: error: MathComp.f90: No existe el archivo o el directorio
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfortran MathComp.f90
gfortran: error: MathComp.f90: No existe el archivo o el directorio
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfortran MathComp.f90 -o MathCom
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ ./MathCom
( 0.00000000 , 1.00000000 ) ( 1.57079637 , 1.31695795 ) (
-Infinity, 0.00000000 )
ines@ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gimp
```

## 1.7 función

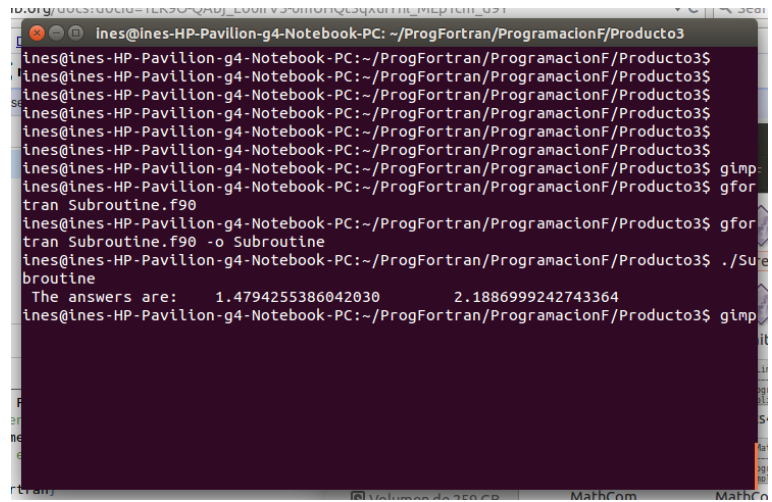
Se declara una función trigonométrica la cual depende de dos variables dadas por el usuario y se utiliza dicha función en un programa para calcular valores en base a ella.

```
! Funcion . f90 : Creando funciones
Real *8 Function f (x,y)
Implicit None
Real *8 :: x, y
f = 1.0 + sin (x*y )
End Function f

Program Main
Implicit None
Real *8 :: Xin =0.25 , Yin =2. , c , f
c = f ( Xin , Yin )
write ( * , * ) 'f(Xin, Yin) = ' , c
```



### 1.8.1 Captura en la Terminal



A screenshot of a terminal window on a Linux system. The window title is 'Ines@Ines-HP-Pavilion-g4-Notebook-PC: ~/ProgFortran/ProgramacionF/Producto3'. The terminal shows a series of commands and their outputs. The user enters 'gimp' multiple times, which appears to be a typo for 'gfortran'. The final output shows the results of a calculation: 'The answers are: 1.4794255386042030 2.1886999242743364'.

```
Ines@Ines-HP-Pavilion-g4-Notebook-PC: ~/ProgFortran/ProgramacionF/Producto3
Ines@Ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$
Ines@Ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$
Ines@Ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$
Ines@Ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$
Ines@Ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gimp:
Ines@Ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfor
tran Subroutine.f90
Ines@Ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gfor
tran Subroutine.f90 -o Subroutine
Ines@Ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ ./Subr
outine
The answers are: 1.4794255386042030 2.1886999242743364
Ines@Ines-HP-Pavilion-g4-Notebook-PC:~/ProgFortran/ProgramacionF/Producto3$ gimp
```