

# Tiro parabólico con FORTRAN y gnuplot

Rios Quijada Danira

10 de Marzo de 2015

## 1. Tiro parabólico con FORTRAN

En esta práctica, primeramente realizamos un programa en FORTRAN para calcular la posición de un objeto lanzado con tiro parabólico, en ciertos instantes de tiempo, el programa también calcula la y máxima y la x máxima, así como el tiempo total en el aire; todo esto dependiendo de la velocidad inicial y del ángulo de tiro. Todos estos cálculos se basan en las ecuaciones de tiro parabólico,

descritas a continuación:

$$\begin{aligned}x &= v_0 t \cos(\theta) \\ y &= v_0 t \sin(\theta) - \frac{1}{2} g t^2\end{aligned}$$

## 1.1. Código

El código que utilizamos para realizar el programa fue el siguiente:

```
!*****
!This program plots projectile motion of an object.
!The program requires user input for initial velocity
!and angle of the object.The algorithm uses a time
!step of 0.01 second i.e. it calculates object's
!location in the x and y plane every 0.01 second.
!*****By: Waleed Ishaque, 2013*****
program tiro_parabolico
  implicit none
  !Defining constants:
  real, parameter :: pi = 4.0*atan(1.0)
  real :: v, a, t, r, vx, vy, xm, ym
  real, parameter :: g = 9.81
  real:: x(150),y(150)
  integer :: i
  !donde g es la aceleracion de la gravedad, pi is "pi"
  !v es la velocidad inicial del objeto
  !a es el angulo de tiro, r es el mismo angulo, pero en radianes
  !t es el tiempo
  !x and y son cordenadas del objeto durante el tiro
  !Seek user input
  write(*,*) 'Dame el ángulo inicial de tiro del proyectil en grados (Real)'
  read *, a
  write(*,*) 'Dame la velocidad inicial del proyectil en m/s (Real) '
  read *, v
  !Convirtiendo los grados a radianes
  r = a*pi/180.0
  ! Calculando las velocidades en los 2 ejes
  vx=(v)*(cos(r))
  vy=(v)*(sin(r))
  !open .dat file and start writing on it using the algorithm
  open(1, file='tiro.dat')
  x=0
  y=0
  do i=1,100
    !displacement of object in x and y direction
    t = (float(i)*0.01)
    x(i) = vx*t
    y(i) = (vy*t) - (0.5*g*t*t)
    !write output in file "proj.dat" for plotting
    write(1,*) x(i), y(i)
    !kill the loop when the object hits the ground
    if (y(i)<0) exit
  end do
  close(1)
  !close file
  ym = (vy**2)/(19.6)
  xm = x(i)
  if (vx<0) then
    xm = 0
  end if
  !resultados en pantalla
  write(*,*) '-----'
  write(*,*) 'Un proyectil con una velocidad inicial de',v,'m/s'
  write(*,*) 'y con un ángulo de tiro de',a, 'grados'
  write(*,*) 'Alcanzará una h máxima de',ym, 'metros'
  write(*,*) 'Su alcance horizontal sera de',xm, 'metros'
  write(*,*) 'y durará en el aire un tiempo de',t,'segundos'
end program tiro_parabolico
```

## 1.2. Resultados

Cabe denotar que en el código se cambia la salida estándar de los datos de posición a un archivo, llamado “tiro.dat”, el cual utilizaremos posteriormente para graficar con gnuplot. una vez concluido el programa, lo compilamos, he aquí los resultados al ejecutarlo con los ángulos de 90,0, 60 y 30 grados.

### 1.2.1. 90 grados

```
danira@danirapc: ~/ProgFortran/ProgramacionF/Producto5
danira@danirapc:~/ProgFortran/ProgramacionF/Producto5$ ./tiro_parabolico
Dame el ángulo inicial de tiro del proyectil en grados (Real)
90
Dame la velocidad inicial del proyectil en m/s (Real)
1
Un proyectil con una velocidad inicial de 1.00000000 m/s
y con un ángulo de tiro de 90.00000000 grados
Alcanzará una h máxima de 5.10204062E-02 metros
Su alcance horizontal sera de 0.00000000 metros
y durará en el aire un tiempo de 0.209999993 segundos
danira@danirapc:~/ProgFortran/ProgramacionF/Producto5$
```

### 1.2.2. 0 grados

```
danira@danirapc: ~/ProgFortran/ProgramacionF/Producto5
danira@danirapc:~/ProgFortran/ProgramacionF/Producto5$ ./tiro_parabolico
Dame el ángulo inicial de tiro del proyectil en grados (Real)
0
Dame la velocidad inicial del proyectil en m/s (Real)
1
Un proyectil con una velocidad inicial de 1.00000000 m/s
y con un ángulo de tiro de 0.00000000 grados
Alcanzará una h máxima de 0.00000000 metros
Su alcance horizontal sera de 9.99999978E-03 metros
y durará en el aire un tiempo de 9.99999978E-03 segundos
danira@danirapc:~/ProgFortran/ProgramacionF/Producto5$
```

1.2.3. 60 grados

```

danira@danirapc: ~/ProgFortran/ProgramacionF/Producto5$ ./ tiro_parabolico
Dame el ángulo inicial de tiro del proyectil en grados (Real)
60
Dame la velocidad inicial del proyectil en m/s (Real)
1
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
Un proyectil con una velocidad inicial de 1.00000000 m/s
y con un ángulo de tiro de 60.00000000 grados
Alcanzará una h máxima de 3.82653102E-02 metros
Su alcance horizontal sera de 8.99999887E-02 metros
y durará en el aire un tiempo de 0.179999992 segundos
danira@danirapc:~/ProgFortran/ProgramacionF/Producto5$ █

```

### 1.2.4. 30 grados

```
danira@danirapc: ~/ProgFortran/ProgramacionF/Productos$  
danira@danirapc:~/ProgFortran/ProgramacionF/Productos$ ./tiro_parabolico  
Dame el ángulo inicial de tiro del proyectil en grados (Real)  
30  
Dame la velocidad inicial del proyectil en m/s (Real)  
1  
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^  
Un proyectil con una velocidad inicial de    1.00000000      m/s  
y con un ángulo de tiro de   30.0000000     grados  
Alcanzará una h máxima de    1.27551015E-02 metros  
Su alcance horizontal sera de   9.52627957E-02 metros  
y durará en el aire un tiempo de 0.109999999 segundos  
danira@danirapc:~/ProgFortran/ProgramacionF/Productos$
```

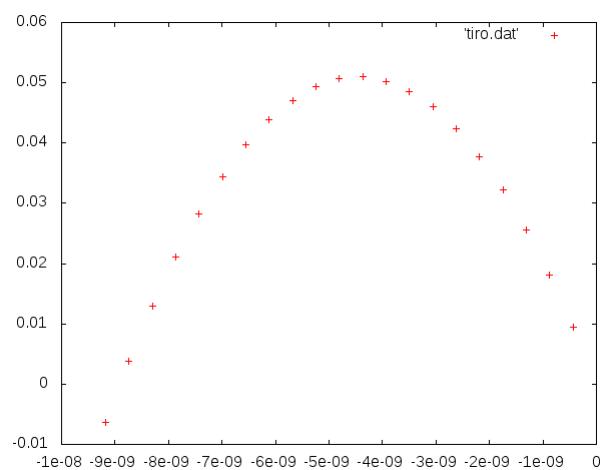
## 2. Graficando con Gnuplot

En esta parte de la práctica nos enfocamos en graficar la información que nos arrojó el archivo “tiro.dat”, para eso utilizamos gnuplot y el siguiente código.

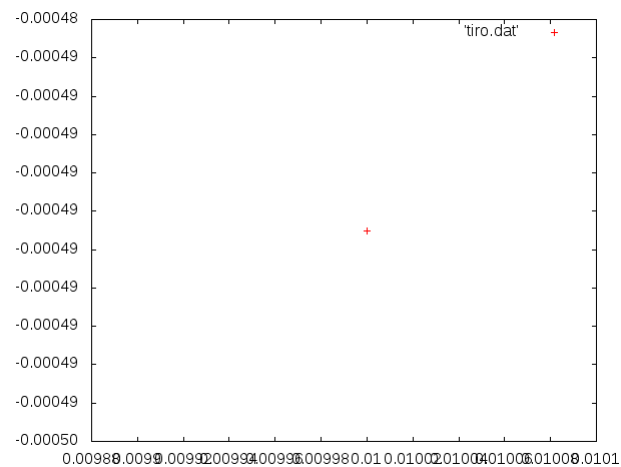
```
gnuplot>set term png
gnuplot>set output 'tirogrados.png'
gnuplot>plot 'tiro.dat'
```

Obteniendo el siguiente resultado para cada ángulo de prueba;

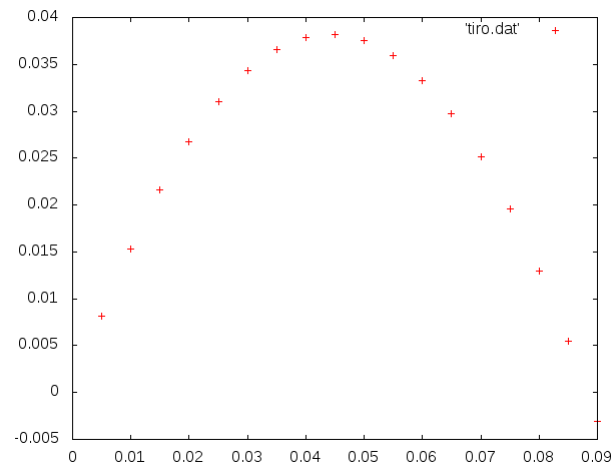
### 2.1. 90 grados



### 2.2. 0 grados



### 2.3. 60 grados



### 2.4. 30 grados

