Tarea 5. Error y su programación.

Gustavo Hernández Angeles[1]

[1]Facultad de Ciencias Físico Matemáticas – Universidad Autónoma de Nuevo León.

gustavoha40@gmail.com

I Primer Problema

Redacción

En el programa para calcular integrales por el método de Trapecios. ¿qué pasaría si las x's (subintervalos) se calcularan de la siguiente manera? Estableciendo antes del DO que $x2 = xi$.

$$x2 = x2 + h$$

a Modifique el programa de manera que se implemente la forma mencionada arriba para calcular las x2, considere el caso cuando $h = 0.1$.

b Evalúe la siguiente integral:

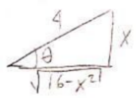
$$\int_{-4}^{4} \sqrt{16 - x^2}$$

Si el número de particiones es 8000, ¿cuál es el error absoluto en el área calculada?

Desarrollo

En este programa realizamos dos integrales por el método de trapecio, la única diferencia es cómo se definen las constantes h y n. En el primer caso consideraremos $h = 0.1$ y en el segundo consideraremos $n = 8000$. Sin embargo, al final calculamos el error absoluto de la integral calculada por cada método. Para esto precisamos del valor verdadero de la integral, podríamos considerar obtener una integral numérica con un número exagerado de intervalos por un método que converja rápidamente (como Romberg), sin embargo, para este caso al tratarse de una integral resoluble analíticamente decidí hacerlo por este mismo método. El resultado nos devuelve $I = 8\pi$.

$$\int_{-4}^{4} \sqrt{16-x^2}\, dx$$



$x = 4\sin\theta \rightarrow dx = 4\cos\theta\, d\theta$

$$= \int 4\cos\theta \sqrt{16 - 16\sin^2\theta}\, d\theta \quad \rightarrow \theta = \arcsin\left(\frac{x}{4}\right)$$

$$= \int 16\cos^2\theta\, d\theta$$

$$= \int 16\, \frac{1+\cos(2\theta)}{2}\, d\theta = \int (8 + 8\cos(2\theta))\, d\theta$$

$$= 8\theta + 4\sin(2\theta)\Big| = 8\arcsin\left(\frac{x}{4}\right)\Big|_{-4}^{4} + 4\sin\left(2\arcsin\left(\frac{x}{4}\right)\right)\Big|_{-4}^{4}$$

$$= 8\arcsin\left(\frac{x}{4}\right)\Big|_{-4}^{4} + 8\sin\left(\arcsin\left(\frac{x}{4}\right)\right)\cos\left(\arcsin\left(\frac{x}{4}\right)\right)\Big|_{-4}^{4}$$

$$= 8\left(\frac{\pi}{2} - \left(-\frac{\pi}{2}\right)\right) + 8\left[\left(\frac{\pi}{2}\right)(0) - \left(-\frac{\pi}{2}\right)(0)\right] = 8\pi$$

$$\therefore \int_{-4}^{4} \sqrt{16-x^2}\, dx = 8\pi$$

De esta forma podemos empezar con nuestro código. Definiremos ahora las variables iniciales que ya hemos visto varias veces en los métodos de integración, la sumatoria, separación h, intervalos n, los extremos del intervalo, la función, dos variables que guardarán los valores de integral calculados por cada método y el valor analítico (verdadero) de la integral. Obviamente, en caso de cambiar la función el valor verdadero de la integral se tendrá que cambiar por un método numérico de alta precisión (que nunca será verdadero), es por esto que decidí utilizar el valor analítico.

```
program problema1
    implicit none
    integer :: n,j
    real*8 :: h,sum,a1,integral,f,xf,xi,x2,integral2, intverd = 8*4.0*atan(1.0)
    write(*,'(a)',advance='no')'Intervalo de la integral: ';read*,xi,xf
```

Después haremos el método de trapecio usual, eligiendo de manera arbitraria un valor para h, haciendo x2 de la forma propuesta en el problema. Observese que el valor de esta integral se guarda en la variable Integral.

```
print*,'---------------------- h arbitraria -----------------------'
write(*,'(a)',advance='no')'Seleccione un valor para h: ';read*,h
n = int((xf-xi)/h)
a1 = (0.5*h)*(f(xf)+f(xi))
sum=0.0
x2 = xi
do j=1,n-1
    x2 = x2+h
    sum = sum+f(x2)
enddo
Integral = h*sum+a1
print*,'La integral numerica trapezoidal evaluada: ',Integral
```

Realizamos el mismo procedimiento con un n arbitrario, cuidando que la suma se inicialice en 0 y x2 vuelva a su estado inicial x2 = xi. El valor para esta integral calculada se guardará en Integral2.

```
print*,'--------------------- n arbitraria ----------------------'
write(*,'(a)',advance='no')'Seleccione un valor para n: ';read*,n
h = (xf-xi)/n
a1 = (0.5*h)*(f(xf)+f(xi))
sum=0.0
x2 = xi
do j=1,n-1
    x2 = x2+h
    sum = sum+f(x2)
enddo
Integral2 = h*sum+a1
print*,'La integral numerica trapezoidal evaluada: ',Integral2
```

Una vez obtenidos los valores de las integrales, procederemos a evaluar el error absoluto e imprimirlo en la terminal, este error absoluto se define como:

$$E = |V_{real} - V_{calculado}|$$

```
print*,'-------------------- Error absoluto ----------------------'
write(*,'(a,f18.13)')'Error absoluto con h arbitraria: ',abs(Integral-Intverd)
write(*,'(a,f18.13)')'Error absoluto con n arbitraria: ',abs(Integral2-Intverd)
```

Al ejecutarlo con los valores que nos propone el problema de h y n el resultado es el siguiente:

```
Intervalo de la integral: -4,4
 ---------------------- h arbitraria ----------------------
Seleccione un valor para h: 0.1
 La integral numerica trapezoidal evaluada:    25.006699929562927
 --------------------- n arbitraria ----------------------
Seleccione un valor para n: 8000
 La integral numerica trapezoidal evaluada:    25.132704041184816
 -------------------- Error absoluto ----------------------
Error absoluto con h arbitraria:    0.1260419985377
Error absoluto con n arbitraria:    0.0000378869158
```

Pero hacer las comparaciones de estos valores y atribuirle el crédito al error por suma de números pequeños a la gran incertidumbre que posee la integral cuando $h = 0.1$ a comparación de utilizar $n = 8000$, ya que cuando $h = 0.1$ el número de intervalos a utilizar en el programa es de 80. Para poder observar este tipo de error de forma más honesta, podemos intentar el caso donde $h = 0.001$ produciendo así los mismos 8000. El resultado se muestra a continuación.

```
Intervalo de la integral: -4,4
 ---------------------- h arbitraria ----------------------
Seleccione un valor para h: 0.001
 La integral numerica trapezoidal evaluada:    25.132614604056059
 --------------------- n arbitraria ----------------------
Seleccione un valor para n: 8000
 La integral numerica trapezoidal evaluada:    25.132704041184816
 -------------------- Error absoluto ----------------------
Error absoluto con h arbitraria:    0.0001273240445
Error absoluto con n arbitraria:    0.0000378869158
```

De esta forma, podemos observar el error producido al definir primero h e ir sumándolo iterativamente. Este error es poco perceptible para nuestro ejemplo pero es también un error propagado, por lo que el error irá aumentando según cuantas veces sumemos este valor.

II Segundo Problema

Redacción

La forma de calcular la magnitud de un vector $< x, y >$ es la siguiente:

$$\sqrt{x^2 + y^2}$$

1. Halle las expresiones equivalentes que permitan calcular la magnitud de un vector cuyas componentes sean grandes y escriba el programa en Fortran.

Desarrollo

Honestamente, no supe cómo cambiar la expresión para la magnitud más que haciendo:

$$\sqrt{(x + y)^2 - 2xy}$$

Sin embargo, los resultados se mantienen siendo los mismos. Seguiré buscando información al respecto.

III Tercer Problema

Redacción

Haga un programa que calcule la siguiente expresión:

$$\frac{k}{10} - \sum_{i=1}^{k} 0.1$$

1. El programa debe calcular el error relativo. Grafique sus resultados y concluya.

Desarrollo

Analíticamente, el resultado de esta operación es 0. Sin embargo, con ayuda de este programa podemos analizar el error de sumar o restar números pequeños de forma iterativa como en el problema número 1. Sin embargo, para calcular un error relativo debemos tener un valor verdadero. Como no podemos dividir entre 0, utilizaremos el número menor al que la máquina ya no redondea (y que es diferente de 0), epsilon, cuyo valor se calcula en un programa dado por el docente. En mi máquina personal este valor es de 1.084×10^{-19}.
Podemos programar esta expresión para distintos valores positivos de k e ir calculando el error relativo en cada uno de ellos. De esta forma, podemos ir escribiendo los resultados en un archivo de datos que podremos graficar. El código para la expresión y el almacenado de datos es el siguiente.

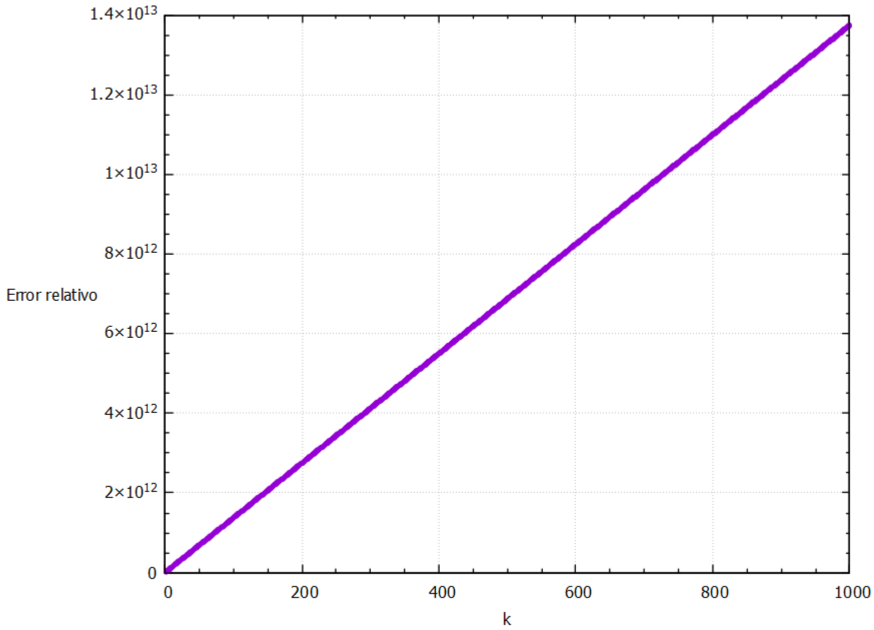
```fortran
real*8 :: epsilon = 1.084e-19, resultado,sum, error_rel
integer :: i, k
open(10, file='p3data.dat')
do k = 1, 1000
    sum=0
    do i = 1, k
        sum = sum + 0.1
    end do
    resultado = k/10.0 - sum
    error_rel = abs(resultado-epsilon)/epsilon
    write(10,*)k,error_rel,abs(resultado)
end do
close(10)
```

Como podemos ver en el código, se calculó la expresión para todos los enteros que van desde 0 a 1000. Además, el código escribe en nuestro archivo de datos mediante 3 columnas; número, el error relativo y el resultado de la expresión. Podemos graficar estos datos fácilmente en GNUPLOT, el siguiente script grafica el error relativo vs k.

```gnuplot
set zeroaxis lt -1 lc -1
unset key
set xlabel 'k'
set ylabel 'Error relativo' rotate by 0
set grid xtics
set grid ytics
set mytics 4
set mxtics 4

plot [0:1000] 'p3data.dat' u 1:2 w p pt 7 ps 0.5 pc 7
```

Los resultados se presentan en la siguiente gráfica:



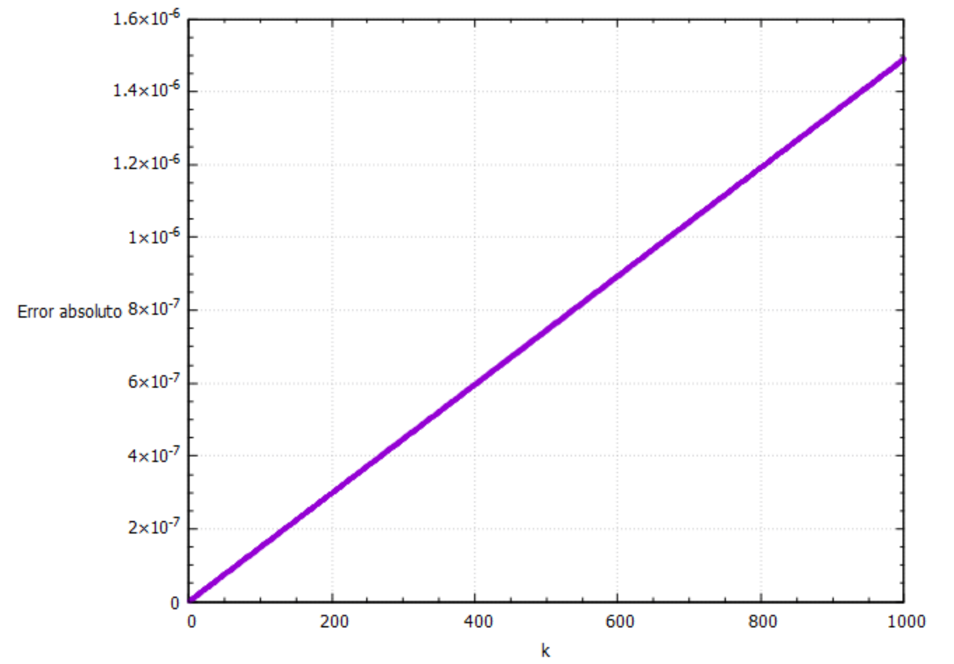
De esta forma, podemos ver que el error crece de forma lineal. Podemos calcular la rápidez con la que crece este error respecto al valor de k utilizando la hoja de datos generada por nuestro programa. Utilizando el primer y último valor obtenemos los siguientes datos:

$$k_1 = 1, E(1) = 13746458283.629713$$

$$k_{1000} = 1000, E(1000) = 13746458282643.215$$

$$\implies Rapidez = \frac{E(1000) - E(1)}{k_{1000} - k_1} = 3.74 \times 10^9$$

Por lo que, con doble precisión, el error relativo aumenta linealmente de forma bastante rápida. Un mejor tratamiento quedaría utilizando el error absoluto, de esta forma únicamente utilizaríamos el valor absoluto del resultado de la expresión. Este error absoluto ya fue guardado en los datos en la tercera columna como se vió en el código. La gráfica resultante es la siguiente:



Calculando la rapidez con la que crece el error absoluto de la misma forma en que lo calculamos para el relativo, nos resulta:

$$Rapidez = 1.49 \times 10^{-9}$$

IV Cuarto problema

Redacción

Del programa de la evaluación 1. Hágalo general de manera que pueda aceptar límites en términos de x y en términos de y. Pruebe el programa con al menos 3 ejemplos de cálculo de longitud de arco.

Desarrollo

En este problema ya conseguimos la mayoría del código, logramos calcular la integral dada la selección del intervalo en términos de y. Para hacer el caso general donde seleccionemos límites en términos de x o de y simplemente le preguntamos al usuario cómo le gustaría definirlos con un condicional.

```
real :: xi, xf, yi, yf
integer :: decision, istep, lstep

 write(*,*)'Intervalos con X o con Y? Seleccione uno.'
write(*,*)'0 - X';write(*,*)'1 - Y'
write(*,'(a)',advance='no')'Eleccion: ';read*,decision
```

Al ingresar la opción 0, el programa simplemente leerá el intervalo en términos de x y procederá directamente a realizar la integral sin preguntar más. En el caso de seleccionar la opción 1, el programa pedirá los límites del intervalo en términos de y y buscará los límites en x mediante el método de la secante junto a un programa para encontrar mini intervalos de las raíces, después se procede a integrar. En el caso en donde la entrada del usuario no coincida con ninguna de las opciones se imprime un mensaje de error y con el comando GOTO vuelve a realizar la pregunta.

```
if (decision .eq. 0) then
    write(*,'(a)',advance='no')'Intervalo en x: ';read*,xi,xf
else if (decision .eq. 1) then
    write(*,'(a)',advance='no')'Intervalo en y: ';read*,yi,yf
    CALL roots(xi,yi)
    call roots(xf,yf)
    call secant(istep,xi,yi)
    call secant(lstep,xf,yf)
else
    write(*,'(a)')'Error, elija una opcion de las disponibles:'
    goto 1
end if
call romberg(xi,xf,10)
```

Hacemos la ejecución del programa con la función que se propuso en la evaluación 1 $f(x) = 0.2\sqrt{3x}$ y da los siguientes resultados:

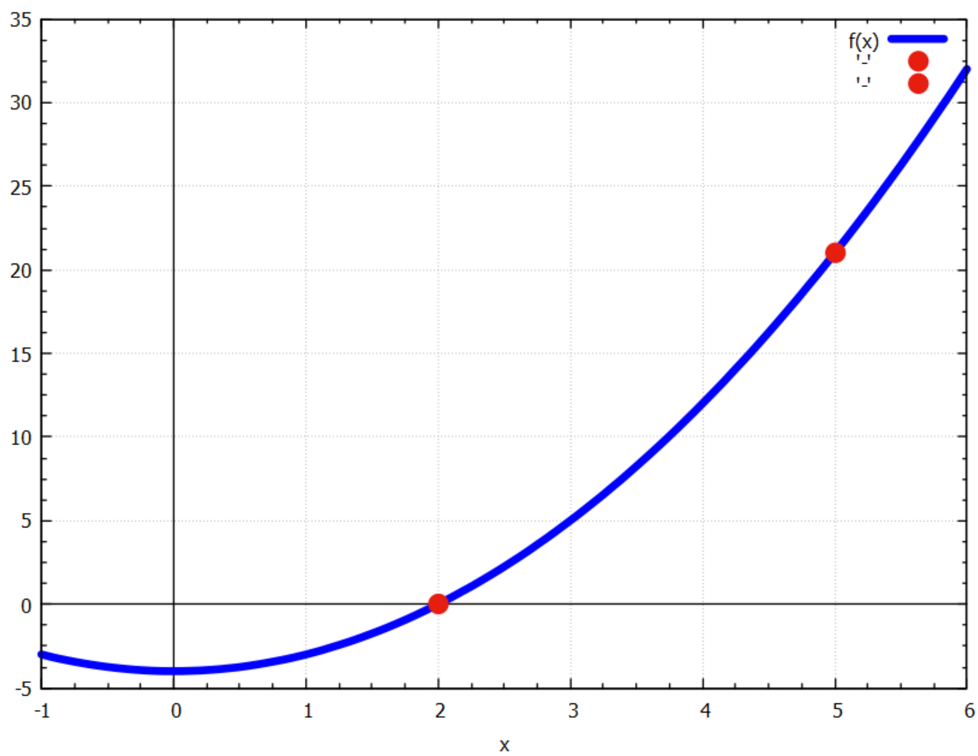
```
 Intervalos con X o con Y? Seleccione uno.
 0 - X
 1 - Y
Eleccion: 1
Intervalo en y: 25,80
 Raiz se encuentra en el intervalo : [   8.23273659     ,    8.43273735     ].
 Raiz se encuentra en el intervalo : [   17.9956837     ,    18.1956844     ].
 Raiz por secante
   8.33333302
 Raiz por secante
   18.0961170
  Integracion por Romberg
     1 |     55.090626
     2 |     55.716785
     3 |     55.849697
     4 |     55.917328
     5 |     55.819717
     6 |     55.831146
     7 |     55.842628
     8 |     55.817554
     9 |     55.822208
    10 |     55.819519
```

```
 Intervalos con X o con Y? Seleccione uno.
 0 - X
 1 - Y
Eleccion: 0
Intervalo en x: 8.333,18.096
  Integracion por Romberg
     1 |     55.182747
     2 |     55.855175
     3 |     55.965717
     4 |     55.964561
     5 |     55.969765
     6 |     55.881123
     7 |     55.852489
     8 |     55.825916
     9 |     55.807659
    10 |     55.812534
```

Como observamos en este caso, definiendo los límites tanto por x como por y arrojan el mismo resultado. Podemos probar para distintas funciones:

1) $f(x) = x^2 - 4$

La gráfica de la función a integrar está delimitada como la siguiente gráfica, siendo los intervalos los puntos [2,0] y [5,21].



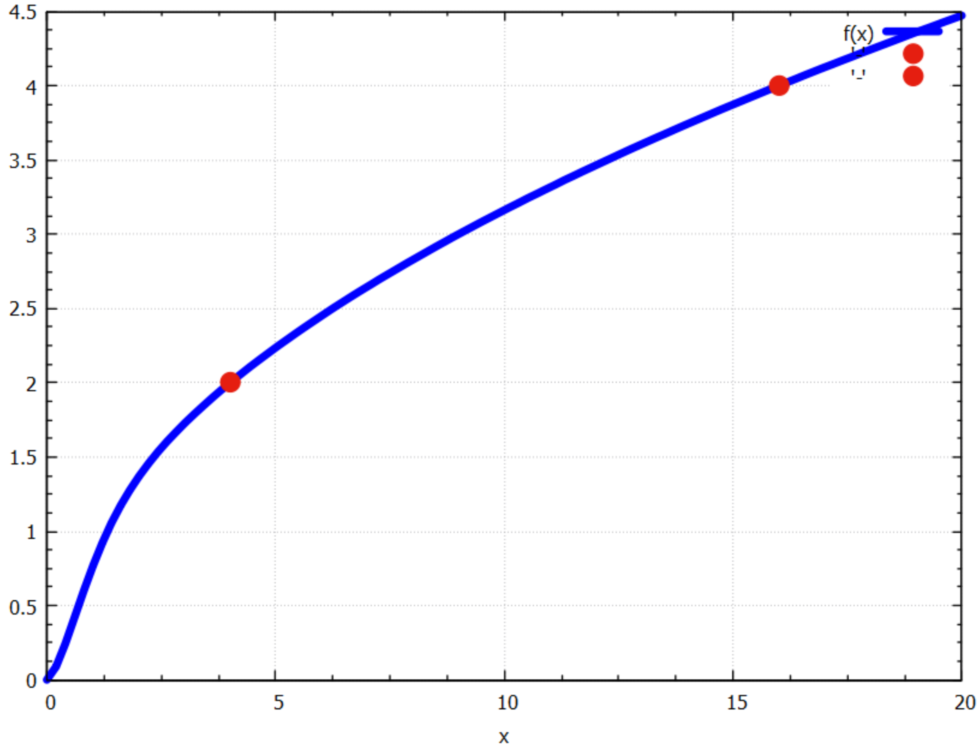
Ingresando estos datos en el programa, podemos encontrar los siguientes resultados:

```
 Intervalos con X o con Y? Seleccione uno.
 0 - X
 1 - Y
Eleccion: 0
Intervalo en x: 2,5
  Integracion por Romberg
    1 |    21.277239
    2 |    21.231014
    3 |    21.231535
    4 |    21.229160
    5 |    21.229107
    6 |    21.226709
    7 |    21.227526
    8 |    21.227551
    9 |    21.227665
   10 |    21.227413
```

```
 Intervalos con X o con Y? Seleccione uno.
 0 - X
 1 - Y
Eleccion: 1
Intervalo en y: 0,21
 Raiz se encuentra en el intervalo : [   1.89903736     ,    2.09903741     ].
 Raiz se encuentra en el intervalo : [   4.89982033     ,    5.09982014     ].
 Raiz por secante
   2.00000000
 Raiz por secante
   5.00000000
  Integracion por Romberg
    1 |    21.277239
    2 |    21.231014
    3 |    21.231535
    4 |    21.229160
    5 |    21.229107
    6 |    21.226709
    7 |    21.227526
    8 |    21.227551
    9 |    21.227665
   10 |    21.227413
```

2) $f(x) = \sqrt{x}\tanh(x)$

La gráfica de la función a integrar está delimitada como la siguiente gráfica, siendo los intervalos los puntos [4.005,2] y [16,4].



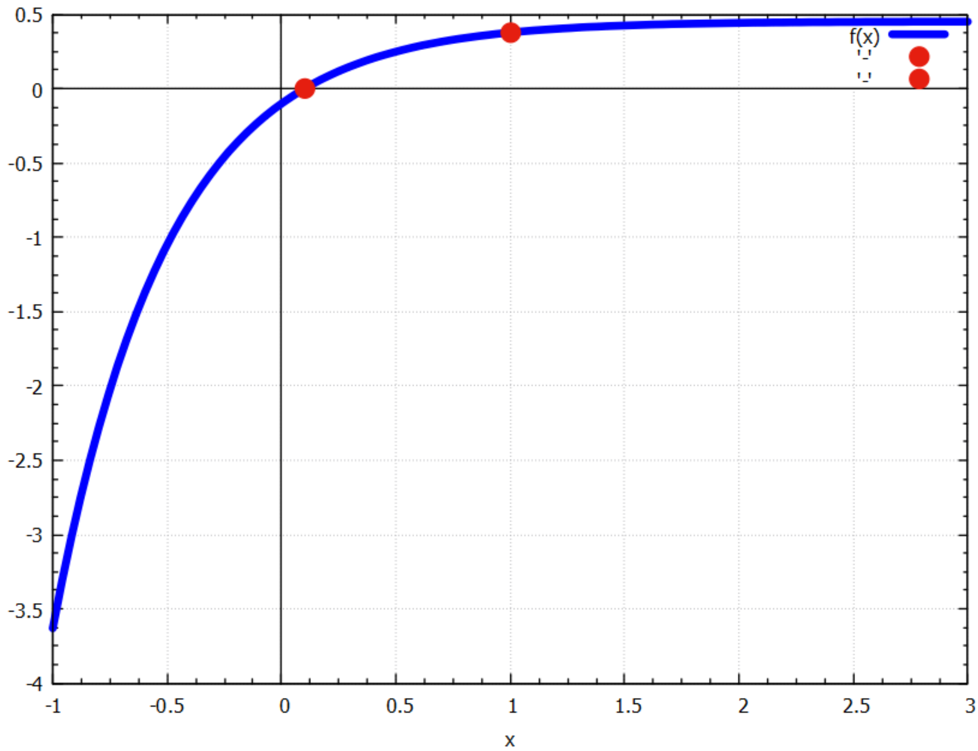
Ingresando estos datos en el programa, podemos encontrar los siguientes resultados:

```
 Intervalos con X o con Y? Seleccione uno.
 0 - X
 1 - Y
Eleccion: 0
Intervalo en x: 4.005,16
  Integracion por Romberg
     1 |     12.231241
     2 |     12.186918
     3 |     12.173378
     4 |     12.169004
     5 |     12.167755
     6 |     12.167385
     7 |     12.167313
     8 |     12.167429
     9 |     12.167505
    10 |     12.167467
```

```
Intervalos con X o con Y? Seleccione uno.
0 - X
1 - Y
Eleccion: 1
Intervalo en y: 2,4
 Raiz se encuentra en el intervalo : [   3.90489244     ,   4.10489225     ].
 Raiz se encuentra en el intervalo : [   15.8998356     ,   16.0998363     ].
 Raiz por secante
   4.00531435
 Raiz por secante
   16.0000000
  Integracion por Romberg
     1 |     12.230922
     2 |     12.187716
     3 |     12.173617
     4 |     12.169008
     5 |     12.167919
     6 |     12.167442
     7 |     12.167125
     8 |     12.167147
     9 |     12.167193
    10 |     12.167167
```

3) $f(x) = e^{-x}\sinh(x-0.1)$

La gráfica de la función a integrar está delimitada como la siguiente gráfica, siendo los intervalos los puntos [0.1,0] y [1.0,0.378].



Ingresando estos datos en el programa, podemos encontrar los siguientes resultados:

```
 Intervalos con X o con Y? Seleccione uno.
 0 - X
 1 - Y
Eleccion: 0
Intervalo en x: 0.1,1.0
  Integracion por Romberg
     1 |      1.061894
     2 |      1.010413
     3 |      0.996069
     4 |      0.992398
     5 |      0.991476
     6 |      0.991251
     7 |      0.991193
     8 |      0.991179
     9 |      0.991175
    10 |      0.991174
```

```
 Intervalos con X o con Y? Seleccione uno.
 0 - X
 1 - Y
Eleccion: 1
Intervalo en y: 0,0.378
 Raiz se encuentra en el intervalo : [  -9.99979675E-04 ,  0.199000031    ].
 Raiz se encuentra en el intervalo : [  0.901990771     ,   1.10199082    ].
 Raiz por secante
  0.100000001
 Raiz por secante
   1.00245023
  Integracion por Romberg
     1 |      1.064726
     2 |      1.013018
     3 |      0.998586
     4 |      0.994898
     5 |      0.993968
     6 |      0.993741
     7 |      0.993675
     8 |      0.993659
     9 |      0.993656
    10 |      0.993654
```

V Quinto problema

Redacción

Calcule el volumen generado por una función $f(x)$ al girarlo con respecto al eje x. Elija un intervalo en x de manera que se tenga un sólido huevo y un sólido "macizo". Haga el programa que calcule el volumen usando el método de arandelas y el de cilindros. Use al menos 5 ejemplos donde compruebe que su programa funciona para ambos tipos de sólidos de revolución.

$$f(x) = 1/x$$

Desarrollo

En este problema, primero definiremos las funciones que utilizaremos para los distintos métodos de integración de volumenes por revolución. La función $w(x)$ indica la función que dará la forma característica al volumen, la función $g(x)$ servirá como el radio menor. Definimos los integrandos tanto del método de cilindros como el de arándelas como la literatura marca.

```
function w(x)
real*8 w,x
w=(1/x)
return
end

function g(x)
real*8 x,g
g=0.5
return
end

function f(x) !Arandelas
real*8 w,x,g,f
f=(w(x))**2-(g(x))**2

return
end


function cil(x)!Cilindros
real*8 cil,x,w,g

cil=w(x)*g(x)
return
end
```

Se utilizará el método de Simpson para la integración numérica de estas funciones. Definimos únicamente el número de intervalos y los extremos del intervalo:

```
program solidos
implicit none
Real*8 a,b,h,i2,sum,ar,xk,f,aran,cilindro,cil
Integer n,k
Print*, '                Metodo de Simpson                  '
print*,'-----------------------------------------------------'
Print*, 'Numero de particiones (par)'
Read*, n
Print*, 'Ingrese el intervalo de integracion'
Read*, a,b
```

Para el método de arandelas se integra la función $f(x)$ que se define como el integrando del método de arandelas. El resultado de la integral se multiplica por π al final para obtener el volumen.

```
print*,'Arandelas******************'
H= abs(b-a)/real(n)
I2= (h/3.0)*(f(a)+f(b))
Sum= 0
Do k=1,n-1
 Xk= a + k*h
 If (mod(k,2).eq.0) then
  Sum= sum + 2.0*f(xk) !terminos pares
 Else
  Sum= sum + 4.0*f(xk) !terminos impares
 End if
End do
Ar = 0
Ar = I2 + sum*(h/3.0)
Print*, 'I=',Ar
aran=4*atan(1.d0)*Ar
print*, 'El volumen del solido de revolucion es:', aran
```

De forma similar, para el método de capas cilindricas se integra la función $cil(x)$ que se define como el integrando del método de capas cilindricas. El resultado de la integral se multiplica por 2π al final para obtener el volumen.

```
print*,'Cilindros*****************'
H= abs(b-a)/real(n)
I2= (h/3.0)*(cil(a)+cil(b))
Sum= 0
Do k=1,n-1
 Xk= a + k*h
 If (mod(k,2).eq.0) then
  Sum= sum + 2.0*cil(xk) !terminos pares
 Else
  Sum= sum + 4.0*cil(xk) !terminos impares
 End if
End do
Ar = 0
Ar = I2 + sum*(h/3.0)
Print*, 'I=',Ar
cilindro=8*atan(1.d0)*Ar
print*, 'El volumen del splido de revolucion es:', cilindro
stop
end program
```

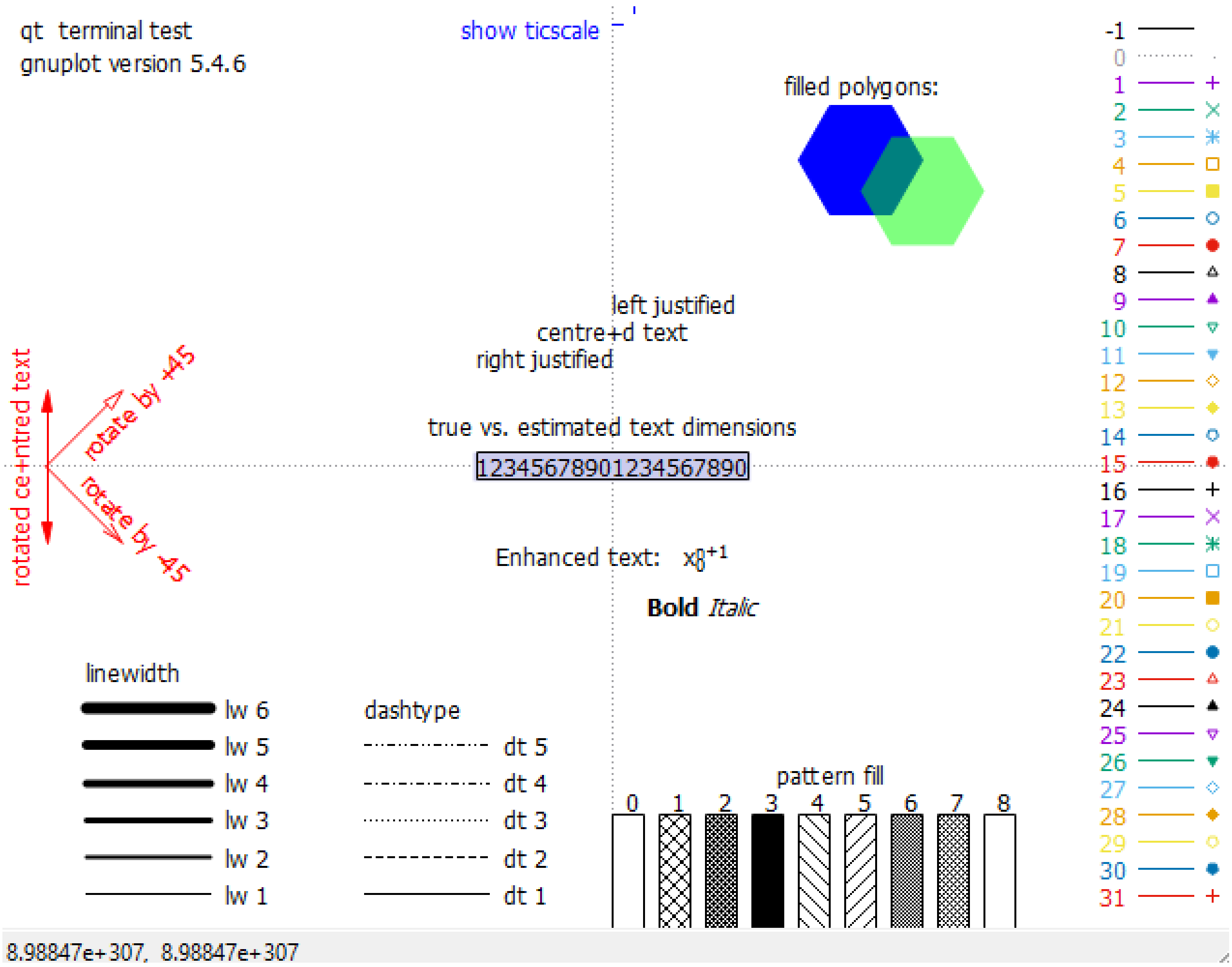
Al ejecutar el programa nos arroja el siguiente resultado:

```
                Metodo de Simpson
 --------------------------------------------------
 Numero de particiones (par)
1000
 Ingrese el intervalo de integracion
1,2
 Arandelas*****************
 I=  0.25000000000012917
 El volumen del solido de revolucion es:  0.78539816339785407
 Cilindros*****************
 I=  0.34657359027998769
 El volumen del solido de revolucion es:   2.1775860903036963
```

Por falta de tiempo no pude completar otros ejemplos, pero pienso correjirlo. :(

VI Conclusión

En esta tarea pude observar el error que existe intrínseco en las computadoras y las maneras de mejorarlo. En esta tarea pude observar los distintos tipos de errores que existen en la manipulación de números intrínsecos a las computadoras y cómo interpretan a los mismos, y varias maneras de neutralizarlos (o mejorarlos). Obtuve resultados curiosos (aunque pendientes a revisar jaja) sobre cómo se propaga el error al ir sumando iterativamente valores pequeños a una variable. También supe graficar puntos en GNUPLOT mediante un código que nos proporcionó (gracias). Otra cosa muy importante que encontre de GNUPLOT para no estar perdiendo tiempo buscando el formato que podemos darle a las gráficas, se trata del comando TEST que nos recibe directamente esta información.


qt terminal test
gnuplot version 5.4.6
show ticscale
filled polygons:
left justified
centre+d text
right justified
true vs. estimated text dimensions
12345678901234567890
rotated ce+ntred text
rotate by +45
rotate by -45
Enhanced text: x
Bold Italic
linewidth
lw 6
lw 5
lw 4
lw 3
lw 2
lw 1
dashtype
dt 5
dt 4
dt 3
dt 2
dt 1
pattern fill
0 1 2 3 4 5 6 7 8
-1 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
8.98847e+307, 8.98847e+307