

PARCIAL 1 – ANÁLISIS NUMÉRICO

Presentado por Adrián Esteban García Ardila.

PUNTO 1

Se realizaron las siguientes pruebas

Para $n = 2$

```
Digite n: 2
Posicion 0, 0 :3
Posicion 0, 1 :4
Posicion 1, 0 :0
Posicion 1, 1 :5

La suma es 12. Numero de operaciones realizadas fue 3. Tiempo de ejecucion = 0 milisegundos
El numero de operaciones realizadas (calculado) es igual a 3
```

Para $n = 3$

```
Digite n: 3
Posicion 0, 0 :2
Posicion 0, 1 :3
Posicion 0, 2 :4
Posicion 1, 0 :0
Posicion 1, 1 :5
Posicion 1, 2 :10
Posicion 2, 0 :0
Posicion 2, 1 :0
Posicion 2, 2 :30

La suma es 54. Numero de operaciones realizadas fue 6. Tiempo de ejecucion = 0 milisegundos
El numero de operaciones realizadas (calculado) es igual a 6
```

Para $n = 4$

```
Digite n: 4
Posicion 0, 0 :5
Posicion 0, 1 :7
Posicion 0, 2 :8
Posicion 0, 3 :1
Posicion 1, 0 :0
Posicion 1, 1 :4
Posicion 1, 2 :9
Posicion 1, 3 :3
Posicion 2, 0 :0
Posicion 2, 1 :0
Posicion 2, 2 :5
Posicion 2, 3 :8
Posicion 3, 0 :0
Posicion 3, 1 :0
Posicion 3, 2 :0
Posicion 3, 3 :32

La suma es 82. Numero de operaciones realizadas fue 10. Tiempo de ejecucion = 0 milisegundos
El numero de operaciones realizadas (calculado) es igual a 10
```

No se obtuvieron diferencias en el tiempo de ejecución medido en las pruebas realizadas debido a que el valor de n fue pequeño en todos los casos. Para ver diferencias en estas mediciones de tiempo es necesario usar valores de n muy grandes. Sin embargo, puede afirmarse que el tiempo de ejecución incrementa al incrementar el valor de n .

Se determinó que dada una matriz A_n se tiene que el número de operaciones requeridas mínimas para calcular la suma de los elementos de la submatriz triangular superior está dado por $f(n) = \frac{n(n+1)}{2}$. Dicha fórmula fue aplicada en las pruebas y se comprobó (como se muestra en los pantallazos) que coincide con el número de operaciones hechas durante la ejecución.

La complejidad del algoritmo implementado es $O(n^2)$.

PUNTO 2

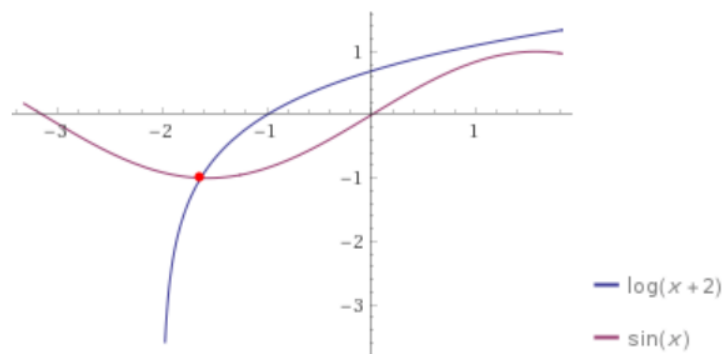
Se comprobó que la sucesión converge, con cada término calculado aproximándose cada vez más hacia el valor de e .

A continuación, se muestra el resultado obtenido.

Término	Iteración	Error relativo
3.0	1	
2.7499999999999996	2	0.09090909090909109
2.7222222222222222	3	0.010204081632653026
2.7187500000000004	4	0.0012771392081734002
2.7183333333333333	5	0.00015328019619883848
2.718287037037037	6	1.70314229753378e-05
2.7182823129251696	7	1.7379033240986566e-06
2.7182818700396827	8	1.6292846295018792e-07
2.718281831765628	9	1.4080237797666107e-08
2.7182818287037036	10	1.1264190708260698e-09
2.7182818284759573	11	8.378316917123542e-11
2.7182818284601415	12	5.818305134519292e-12
2.718281828459112	13	3.786946510310784e-13
2.718281828459049	14	2.319872322968696e-14
2.718281828459045	15	1.4703416131491758e-15
2.718281828459045	16	0.0

PUNTO 3

Las dos funciones tienen un punto de intersección, como se muestra a continuación.



Se planteó una función $h(x)$ como la resta de $f(x)$ y $g(x)$ para hallar la intersección de éstas últimas dos. Por tanto, $h(x) = f(x) - g(x) = \log(x+2) - \sin(x)$. Esta última función se usó para aplicar los algoritmos.

Parte a)

Dada la restricción en el dominio de la función $f(x)$, esto es, $x \neq -2$, y con el respaldo de la gráfica de las funciones $f(x)$ y $g(x)$, se tomó el intervalo $[-1.8, -1]$ para aplicar el algoritmo.

Se obtuvo el punto de intersección en $x = -1.6314435969774892$, como se muestra enseguida.

$x = -1.840215350016111$	$E = 0.13334201147774308$
$x = -1.573703771789245$	$E = 0.06274762265997307$
$x = -1.6123100245020496$	$E = 0.01968388196267236$
$x = -1.6331029050504058$	$E = 0.0016551578952348689$
$x = -1.6313958129917459$	$E = 4.778741787903624e-05$
$x = -1.6314434774712356$	$E = 1.194976707349975e-07$
$x = -1.6314435969774892$	$E = 8.604381612152636e-12$

Parte b)

Dada la restricción ya nombrada en el dominio de la función $f(x)$ con el respaldo de la gráfica de las funciones $f(x)$ y $g(x)$, se tomó -0.8 como valor inicial para aplicar el algoritmo.

Se obtuvo el punto de intersección en $x = -1.63144359696828$, como se muestra enseguida.

Iteración	Aproximación	Error relativo
1	-1.49279847723886	0.205789011340199
2	-1.61000342859342	0.0727979514037130
3	-1.63079569811286	0.0127497696636723
4	-1.63144296577157	0.000396745502164725
5	-1.63144359696828	3.86894597495152e-7
La raíz es -1.63144359696828		