

Ejercicio



Serie Fibonacci (Comparación)

Integrantes del equipo

César Mauricio Arellano Velásquez

Profesor

César Arturo Ángeles Ruiz

Materia

Taller de Desarrollo de Aplicaciones

Introducción

La sucesión de Fibonacci es una sucesión definida por **recurrencia**. Esto significa que para calcular un término de la sucesión se necesitan los términos que le preceden.

Se proporcionan los dos primeros términos: $a_0=0$ y $a_1=1$.

Los siguientes se calculan con la siguiente fórmula:

$$a_{n+1} = a_{n-1} + a_n, \quad n \geq 1$$

Definición del problema

El objetivo del programa era realizar dos programas que construyeran la serie fibonacci, uno de forma recursiva y otro de manera iterativa; esto para comparar cuál método era el más eficiente respecto a su tiempo de ejecución, ciclo por ciclo o llamada por llamada.

Pseudocódigo Iterativo

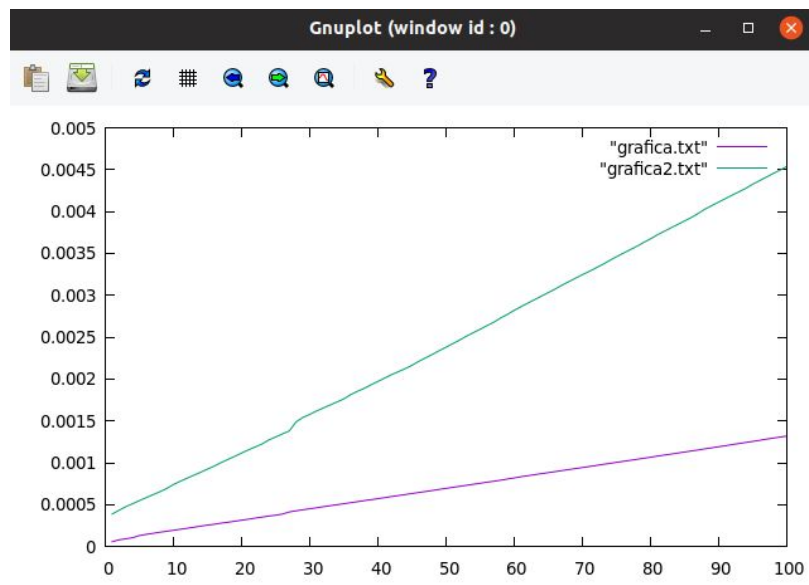
```
#define clock() -> función que  
retorna tiempo actual.  
#define CLOCKS_PER_SEC ->  
función que retorna tiempo en  
segundos.  
principal(){  
    Imprimir("Serie Fibonacci");  
    Imprimir("Escribe el rango de  
iteraciones:");  
    Leer(rango);  
    Fibonacci(rango);  
}  
Fibonacci(rango){  
    Archivo =  
AbrirArchivo("grafica.txt",Modo  
Escritura);  
    Start = clock();  
    num=0;  
    num2=1;  
    res=1;
```

Pseudocódigo Recursivo

```
#define clock() -> función que  
retorna tiempo actual.  
#define CLOCKS_PER_SEC ->  
función que retorna tiempo en  
segundos.  
principal(){  
    Start = clock();  
    Imprimir( "Introduce rango de  
números fibonacci a imprimir: " );  
    Leer(numero);  
    Fibonacci(0,1, numero,Start,1);  
}  
Fibonacci(penultimo, ultimo, n,  
Inicio, i)  
{  
    Archivo =  
AbrirArchivo("grafica2.txt","Modo  
adjuntar")  
    Imprimir(ultimo);
```

<pre> desde i=0; hasta i<rango; i++){ res = num+num2; num = num2; num2 = res; Si (i>=9){ ImprimirEnArchivo(Archivo,i+1); End = clock(); Time = (End-Start) /CLOCKS_PER_SEC; ImprimirEnArchivo(Archivo,Time); } } CerrarArchivo(Archivo); } </pre>	<pre> Si(i>=10){ ImprimirEnArchivo(Archivo,i); Fin = clock(); Time = (Fin-Inicio) /CLOCKS_PER_SEC; ImprimirEnArchivo(Archivo,Time); } Si (n>1){ actual = penultimo + ultimo; n=n-1; Fibonacci(ultimo,actual,n,Inicio,i+1); } } </pre>
---	--

Gráfica Iterativo Vs Recursivo



Conclusión

Opino que el método recursivo consume más tiempo de ejecución en comparación con el iterativo debido a que las llamadas a las función consumen un cierto espacio en la memoria por la generación de variables nuevas.