

Proyecto 0: Fibonacci Recursivo contra Fibonacci Iterativo

Ricardo Castro Jiménez

Juan Carlos Valverde Padilla

Escuela de Ingeniería en Computación, Tecnológico de Costa Rica, San José, Costa Rica

riccastro@estudiantec.cr

stickvalv@estudiantec.cr

En este proyecto, se despliega una tabla con la cantidad de trabajo que deben realizar las dos versiones estudiadas de algoritmos para calcular el n -ésimo término de una serie de Fibonacci. Los argumentos de la línea de comando afectarán el rango de valores desplegados. Toda la programación debe hacerse en Lenguaje C sobre Linux. La salida de este proyecto es a consola.

En clases estudiamos un algoritmo recursivo y un algoritmo iterativo para calcular el n -ésimo término de Fibonacci. Se calculan cantidades de trabajo que cada versión requiere, tanto con las fórmulas exactas, así como con la fórmula para la cota inferior. También se dará la cantidad de tiempo que tomaría cada algoritmo corriendo en nuestra máquina hipotética que realiza una instrucción por nanosegundo (10^{-9} segundos).

Se usó la función de C llamada `getopt()` de la librería “`unistd.h`” Para poder analizar los argumentos de la línea de comandos que se pasan a un programa, especialmente aquellos que siguen la convención de guiones (-). Permite a los programas interpretar opciones como `-k 10` o `-A 5`, haciendo que sean más flexibles y fáciles de usar sin necesidad de modificar el código y recompilar.[1]

Fuentes:

[1] GeeksforGeeks, “getopt() function in C to parse command line arguments,” GeeksforGeeks. Available: <https://www.geeksforgeeks.org/getopt-function-in-c-to-parse-command-line-arguments/>. [Accessed: Aug. 24, 2025].