

PROGRAMACION II - GUIA DE EJERCICIOS DE PROGRAMACIÓN DINÁMICA

1.- Escriba un programa que calcule el valor de Fibonacci de manera recursiva para el número entero 30 e imprima la cantidad de llamadas que se ejecutan para obtener el mismo. Luego, optimice dicho algoritmo utilizando algunas de las estrategias de Programación Dinámica Top-Down o Bottom-Up. Vuelva a imprimir el valor de Fibonacci para el mismo entero 30 y la cantidad de llamadas que se utilizó para el mismo.

Por último verifique lo siguiente:

- ¿Fueron los resultados de cálculo de Fibo(30) similares?
- ¿Obtuvo optimización en las llamadas recursivas utilizando alguna de las estrategias de Programación Dinámica?
- ¿Cual es la relación de optimización?

2.- El combinatorio $\binom{n}{k}$ se puede calcular como $\frac{n!}{k!(n-k)!}$. Sabiendo que el cálculo combinatorio se obtiene a partir de la fórmula:

$$n! = n \cdot (n - 1) \cdot (n - 2) \cdot \dots \cdot 2 \cdot 1$$

Realice un algoritmo que permita obtener el combinatorio de 2 números enteros. Intente optimizar dicho algoritmo utilizando alguna de las técnicas de Programación Dinámica para evitar realizar cálculos repetitivos.

3.- Maximum Increase (Ejercicio de CodeForces)

<http://codeforces.com/problemset/problem/702/A>