



UNIVERSIDAD
DE GRANADA

Universidad de Granada

FACULTAD DE INGENIERÍA INFORMÁTICA Y
TELECOMUNICACIONES

PRÁCTICA 1: EFICIENCIA DE ALGORITMOS

Doble Grado Ingeniería Informática y Matemáticas

Autores:

Adolfo Martínez Olmedo, Pablo Delgado Galera,
Marcos Baena Solar

Marzo 2025

Índice

1. Introducción	2
1.1. Análisis de la eficiencia teórica	2
1.2. Análisis de la eficiencia empírica	2
1.3. Análisis de la eficiencia híbrida	2
2. Desarrollo	3
2.1. Ordenación de vectores	4
2.2. Los números de Catalan	4
2.2.1. Versión recursiva	4
2.2.2. Versión iterativa (programación dinámica)	4
2.2.3. Versión iterativa directa usando el coeficiente binomial	4
2.3. Las Torres de Hanoi	4
2.3.1. Versión recursiva	4
2.3.2. Versión iterativa usando una pila	4
2.3.3. Versión iterativa sin usar la pila	4
3. Conclusión	4

1. Introducción

Comenzemos estableciendo las características de cada uno de nuestros ordenadores, ya que tienen prestaciones diferentes

	CPU	RAM	Caché L1d	Caché L1i	Caché L2	Caché L3	SO
Adolfo Martínez	AMD Ryzen 7 4800HS	16GB	256 KiB	256 KiB	4 MiB	8 MiB	Ubuntu 22.04.4
Marcos Baena	AMD 3020e with Radeon Graphics	5.7Gi	64 KiB (2 instances)	128 KiB (2 instances)	1 MiB (2 instances)	4 MiB (1 instance)	Ubuntu 24.04.1 LTS
Pablo Delgado	11th Gen Intel(R) Core(TM) i7-11800H	15 Gi	384 KiB (8 instances)	256 KiB (8 instances)	10 MiB (8 instances)	24 MiB (1 instance)	Ubuntu 24.04.1

Cuadro 1: Características de los ordenadores

En esta práctica vamos a discutir la eficiencia de los algoritmos desde tres puntos de vista distintos:

- **Punto 1:** Descripción del primer punto.
- **Punto 2:** Descripción del segundo punto.
- **Punto 3:** Descripción del tercer punto.

1.1. Análisis de la eficiencia teórica

En el análisis de la eficiencia teórica estudiaremos el tiempo de ejecución del algoritmo mediante funciones en notación *Big-O*, que representarán el peor caso posible. En este análisis, no usaremos medidas reales de computación, sino que calcularemos funciones mediante técnicas vistas en Estructuras de Datos y Algorítmica.

1.2. Análisis de la eficiencia empírica

Para el análisis de la eficiencia empírica ejecutaremos los algoritmos implementados en C++ en cada una de nuestras máquinas y mediremos el tiempo de ejecución mediante la clase `<chrono>`. Cada miembro del equipo ejecutará cada algoritmo 10 veces con todos los tamaños especificados, para luego hacer una media y obtener resultados más fiables.

1.3. Análisis de la eficiencia híbrida

En el análisis de la eficiencia híbrida, tomamos los resultados de los integrantes del grupo y hallamos la constante κ . En la representación de los resultados usaremos la herramienta `gnuplot`.

Para poder completar esta parte del estudio de la eficiencia usaremos los resultados del análisis teórico, para poder conocer la forma de la función a

la que queremos ajustar los datos. Por ejemplo para representar en gnuplot $O(n^3)$:

```
1 gnuplot> f(x) = a0*x*x+a1*x*x+a2
```

Listing 1: Ejemplo de $O(n^2)$

Después de esto debemos hacer la regresión usando el método de mínimos, cuyo funcionamiento conoces gracias a la asignatura EDIP:

```
1 gnuplot> fit f(x) 'result.dat' via a0,a1,a2
```

Listing 2: Uso de gnuplot para l regresión

En este caso result.dat es nuestro fichero de datos. Nos centraremos en la parte que pone *Final set of Parameters*, que nos muestra los coeficientes de la fórmula de regresión junto con la bondad del ajuste realizado.

Finalmente, hacemos el plot de los puntos y la curva de ajuste para ver como de buena es el cálculo de la eficiencia híbrida. Usaremos el siguiente comando:

```
1 gnuplot> plot 'result.dat', f(x) title 'Curva de  
ajuste'
```

Listing 3: Representación de la regresión

2. Desarrollo

Una vez que hemos discutido las maneras de estudiar la eficiencia, veamos los problemas que vamos a analizar: La **Ordenación de vectores**, los **Números de Catalan**, y las **Torres de Hanoi**.

2.1. Ordenación de vectores

2.2. Los números de Catalan

2.2.1. Versión recursiva

2.2.2. Versión iterativa (programación dinámica)

2.2.3. Versión iterativa directa usando el coeficiente binomial

2.3. Las Torres de Hanoi

2.3.1. Versión recursiva

2.3.2. Versión iterativa usando una pila

2.3.3. Versión iterativa sin usar la pila

3. Conclusión

En conclusión, somos sudorosos.