

ALGORITMOS Y ESTRUCTURAS DE DATOS III  
Trabajo Práctico Nº1

**Primera entrega:** LUNES 13-Marzo-2008, hasta las 19:30 horas.

**Segunda entrega:** Viernes 24-Marzo-2008, hasta las 19:30 horas.

Ver información general sobre los Trabajos Prácticos en la página de la materia en Internet.

---

- Desarrollar e implementar algoritmos para los problemas enumerados.
- Utilizando el pseudocódigo, calcular el orden de la complejidad de cada algoritmo, y decir si es constante, logarítmica, lineal, polinomial, exponencial o mayor **en función del tamaño de la entrada**.
- Aplicar los algoritmos a un conjunto de instancias de entrada, midiendo para cada instancia el tiempo de ejecución o la cantidad de operaciones. Elegir las instancias de manera tal que se pueda apreciar el comportamiento de los algoritmos e incluirlas en el soporte digital a entregar.
- Graficar para cada algoritmo el tiempo de ejecución o la cantidad de operaciones en función del *tamaño de la entrada*. **Comparar con la complejidad teórica calculada.**
- Leer los datos de entrada desde un archivo con el nombre `Tp1EjX.in`, donde `X` es el número del problema. Escribir los resultados en un archivo con el nombre `Tp1EjX.out`.
- Respetar los formatos de archivos que se indican en cada caso.
- Ignorar los costos de lectura y escritura de los archivos tanto al medir como al calcular complejidad.

1. **Enunciado**

El 7 de junio de 1742, Christian Goldbach le escribió una carta a Leonhard Euler (uno de los más grandes matemáticos de todos los tiempos), sugiriéndole que pensara una demostración para la siguiente afirmación porque a él no se le ocurría:

“Todo número par positivo, mayor que dos, se puede escribir como la suma de dos números primos.”

Esta se conoce como la conjetura Goldbach y si bien no ha sido probada, ningún matemático ha hallado jamás número par alguno mayor que 2, que no pudiera expresarse mediante la suma de dos números primos.

Diseñar un algoritmo para probar la conjetura, es decir, dado  $n$ , encontrar dos números primos cuya suma de  $n$  (Obs: esta descomposición puede no ser única).

**Entrada** `Tp1Ej1.in`

Cada instancia de entrada es definida en una línea que contiene el entero positivo  $n$ . El archivo conteniendo varias instancias termina con una línea con solamente un 0, la cual no debe ser considerada como una instancia.

**Salida** `Tp1Ej1.out`

Para cada instancia de la entrada, se debe escribir, en caso de hallar la descomposición,

una línea en la salida conteniendo ambos primos encontrados, separados por un espacio en blanco y en caso de no hallar la descomposición, se debe escribir una línea de salida que diga “Goldbach la pifió!”. No debe haber espacios al comienzo o al final de la línea.

## 2. Enunciado

Se dispone de  $n^2$  fichas cuadradas de  $1\text{ cm}^2$  de lado, cada una con un número en cada arista del cuadrado. El juego consiste en colocar las fichas en un tablero de  $n \times n\text{ cm}^2$  tal que dos fichas pueden ser consecutivas (en el sentido vertical u horizontal, es decir, que comparten un lado), si tienen el mismo número en ese lado del cuadrado. Las fichas no se pueden rotar, es decir los números siempre tienen que estar derechos. Además los números que quedan en los bordes del tablero pueden ser cualquier número.

Diseñar un algoritmo que, dado un conjunto de fichas, decida si se puede llenar el tablero o no, y en ese caso, dar la solución.

### Entrada Tp1Ej2.in

Cada instancia de entrada es definida de la siguiente manera: La primera línea contiene el entero positivo  $n$ . Las siguientes  $n^2$  líneas contienen las  $n^2$  fichas. Cada ficha está dada por 4 enteros positivos separados por un espacio en blanco, siendo cada uno, el número de un lado de la arista del cuadrado en el orden siguiente: ARRIBA, ABAJO, IZQUIERDA, DERECHA. El archivo conteniendo varias instancias termina con una línea con solamente un 0, la cual no debe ser considerada como una instancia.

### Salida Tp1Ej2.out

Para cada instancia de la entrada, se debe escribir  $n^2$  líneas con las fichas en el orden que deben ser puestas en el tablero, de izquierda a derecha, de arriba hacia abajo. No debe haber espacios al comienzo o al final cada línea.

## 3. Enunciado

Dado un vector  $v$  de enteros, se define la mediana de  $v$  como el elemento que, una vez ordenado el vector, deja a ambos lados la misma cantidad de elementos. Sean dos vectores  $X, Y$  de tamaño  $n$ , cada uno ordenado en forma no decreciente. Hallar la mediana de los dos vectores, es decir, si ordenáramos la unión de los dos vectores, sería el elemento de la posición  $n$ .

### Entrada Tp1Ej3.in

Cada instancia primero define el tamaño  $n$  en la primera línea y luego en las dos siguientes cada uno de los vectores  $X$  e  $Y$ , con espacios en blanco para separar cada entero. El archivo conteniendo varias instancias termina con una línea con solamente un 0, la cual no debe ser considerada como una instancia.

### Salida Tp1Ej3.out

Para cada instancia de entrada, se debe escribir una línea con la mediana correspondiente. No debe haber espacios al comienzo o al final cada línea.