

## TRABAJO PRÁCTICO 1

### PROBLEMA 1

Tenemos una serie de  $n$  partículas cargadas muy pequeñas, ubicadas a intervalos regulares a lo largo de una línea recta en las ubicaciones  $\{1, 2, \dots, n\}$ . En cada uno de estos puntos, tenemos una partícula cargada  $q_j$ , que puede ser positiva o negativa. Queremos estudiar la fuerza total sobre cada partícula mediante la Ley de Coulomb:

$$F_j = \sum_{i < j} C \frac{q_i q_j}{(j - i)^2} - \sum_{i > j} C \frac{q_i q_j}{(j - i)^2}$$

Desarrollar un algoritmo que calcule  $F_j$  mediante División y Conquista.

### PROBLEMA 2

Juan Curuchet tiene planeando un rally por el Camino de las Altas Cumbres. Puede llevar dos litros de agua y rodar 7 kilómetros antes de que se le agote. Tiene un mapa con lugares donde puede repostar agua, y conoce la distancia entre cada uno. El objetivo de Juan es detenerse la menor cantidad de veces que sea posible. Desarrollar un algoritmo Greedy que determine en qué lugares detenerse a cargar agua, y mostrar si siempre encuentra el óptimo o no.

### PROBLEMA 3

Una *suma encadenada* es una secuencia de números  $a_0, a_1, \dots, a_l$  tal que  $a_0=1$  y para cada  $k>0$ :  $a_k=a_i+a_j$  para algún  $i, j, k < l$ . Por ejemplo, 1, 2, 3, 6 es una suma encadenada para  $n=6$  de longitud  $l=3$  ya que:

- $2=1+1$
- $3=2+1$
- $6=3+3$

Desarrollar un algoritmo de Backtracking para calcular una suma encadenada de longitud mínima para un entero positivo  $n$

### ***Requisitos para cada problema***

1. Supuestos: identificar supuestos, condiciones, limitaciones y/o premisas bajo los cuales funcionará el algoritmo desarrollado
2. Diseño: Pseudocódigo y estructuras de datos utilizadas
3. Seguimiento: Ejemplo de seguimiento con un set de datos reducido
4. Complejidad: Análisis de la complejidad temporal a partir del pseudocódigo
5. Sets de datos: diseñar sets de datos aleatorios. Se pueden generar utilizando una función random con una semilla fija, para permitir la reproducibilidad de los resultados. Cada set de datos debe ser incluido en la entrega, junto con el resultado obtenido en cada caso. El programa entregado debe generar los sets de datos o leerlos desde archivos.
6. Tiempos de Ejecución: medir los tiempos de ejecución de cada set de datos y presentarlos en un gráfico.
7. Informe de Resultados: redactar un informe de resultados comparando los tiempos de ejecución. ¿Se corresponde con la complejidad determinada inicialmente?

### ***Condiciones Generales de Entrega***

- El trabajo debe ser entregado en un archivo zip conteniendo:
  - Documento con carátula, índice y numeración de páginas. La carátula debe incluir nombre y padrón de los integrantes del grupo. Debe presentarse en formato PDF.
  - Archivos con los fuentes de los algoritmos desarrollados.
  - Archivo README indicando el lenguaje de programación utilizado, versión mínima y bibliotecas requeridas, e instrucciones para ejecutar.
  - Archivos con los sets de datos utilizados
  - Archivos con resultados obtenidos para cada set de datos
- IMPORTANTE: el tamaño de los sets de datos debe ser tal que permita obtener suficiente información como para graficar los tiempos de ejecución y verificar la complejidad temporal.
- Si se incluyeran referencias bibliográficas, utilizar normas APA 7ma edición