Universidad Simón Bolívar
Departamento de Computación y Tecnología de la Información
CI5651 – Diseño de Algoritmos I
Enero–Marzo 2024

Tarea 8

A continuación encontrará 3 preguntas, cada una dirá cuántos puntos vale en su preámbulo. Sea lo más detallado y preciso posible en sus razonamientos, algoritmos y demostraciones.

Además del informe expresando su solución, debe dar una implementación de su solución en el lenguaje de su elección (solamente como una función; el formato de entrada/salida no es relevante), para las preguntas 2 y 3.

La entrega se realizará únicamente por correo electrónico a rmonascal@gmail.com.

Fecha de entrega: Hasta las 11:59pm. VET del **Miércoles, 04 de Abril** (*Semana 11*).

1. (2 puntos) – Considere un polinomio formado por los números de su carné, donde el i–ésimo número corresponde al coeficiente para x^i.

   Por ejemplo, si su carné es `12-02412`, entonces el polinomio será:

$$\begin{aligned} P(x) &= 1x^0 + 2x^2 + 0x^3 + 2x^4 + 4x^5 + 1x^6 + 2x^7 \\ &= 1 + 2x^2 + 2x^4 + 4x^5 + x^6 + 2x^7 \end{aligned}$$

   Calcule y muestre el resultado de aplicar la DFT (Transformada Discreta de Fourier) al polinomio obtenido, usando las **raíces ocatavas** de la unidad.

2. (4 puntos) – Considere un número entero positivo X. Definimos la función $decomp(X)$ como la cantidad de enteros positivos, a, b, c y d de tal forma que $ab + cd = X$.

$$decomp(X) = |\{(a, b, c, d) : a > 0 \land b > 0 \land c > 0 \land d > 0 \land ab + cd = X\}|$$

   Dado un número N, queremos hallar el máximo valor para $decomp(X)$ donde $1 \leq X \leq N$.

   Diseñe un algoritmo que permita encontrar la respuesta en $O(N \log N)$.

   Nota: Puede suponer que todas las operaciones aritméticas, incluyendo multiplicaciones, divisiones y módulos se hacen en $O(1)$.

   Pistas:

   - ¿De cuántas formas se puede descomponer N en dos sumandos a y b, tal que $a+b = N$?
   - ¿De cuántas formas se puede descomponer N en dos factores a y b, tal que $a \times b = N$?
   - ¿Que relación existe entre la cantidad de divisores de un número y su descomposición en factores primos?
   - La Criba de Eratóstenes se puede usar para ver si un número es primo. ¿Se podrá modificar para calcular algo más?
   - Un cambio de perspectiva pudiera ser de utilidad.

3. (3 puntos) – Sea $P = \{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\}$ un conjunto de n puntos.

   Para cualquier subconjunto $C \subseteq P$, definimos la lejanía de C como la multiplicación de la distancia horizontal hasta el origen más grande en C por la distancia vertical hasta el origen más grande en C (nótese que no necesariamente es el mismo punto quien tiene estos máximos).

$$lejania(C) = \left(\max_{(x,y)\in C} |x|\right) \times \left(\max_{(x,y)\in C} |y|\right)$$

   Queremos realizar una partición de P en subconjuntos $C_1, C_2, \cdots, C_m$ de tal forma que

   - $C_1 \cup C_2 \cup \cdots \cup C_m = P$
   - $C_1 \cap C_2 \cap \cdots \cap C_m = \emptyset$

   La cantidad m de subconjuntos que forma la partición es libre, entre 1 y n. La lejanía de la partición es la suma de las lejanías de los conjuntos que lo conforman.

$$lejania(C_1 \cup C_2 \cup \cdots \cup C_m) = lejania(C_1) + lejania(C_2) + \cdots + lejania(C_m)$$

   Diseñe un algoritmo que permita hallar una partición con mínima lejanía en $O(n \log n)$, usando memoria adicional $O(n)$.

   Pistas:

   - ¿Existen puntos en la entrada que son redundantes?
   - ¿Dar un orden a los puntos nos permitiría considerar **subsecuencias** en lugar de **subconjuntos**?
   - La geometría es un área muy útil de las matemáticas.

R. Monascal / Marzo 2024