

Tarea Examen 2
Análisis de Algoritmos

Fecha de entrega: Noviembre 05 de 2020.
No se reciben tareas después de esta fecha.

Todas sus respuestas deben estar plenamente justificadas.

1. Sea $M[1 \dots n][1 \dots n]$ una matrix de $n \times n$, en el que cada renglón y cada columna están ordenados en orden creciente. Suponga que no hay dos elementos iguales.
 - a) Diseña un algoritmo que encuentre la posición de un valor k en M o que determine si que no está. ¿Cuántas comparaciones usa tu algoritmo en el peor caso?
 - b) Describe y analiza un algoritmo para resolver el siguiente problema en tiempo lineal. Dados 4 índices i, j, i', j' como entrada, calcule el número de elementos de M que son más pequeños que $M[i][j]$ y más grandes que $M[i'][j']$.
2. **Permutaciones de Josephus:** Supongamos que n personas están sentadas alrededor de una mesa circular con n sillas, y que tenemos un entero positivo $m \leq n$. Comenzando con la persona con etiqueta 1, (moviendonos siempre en la dirección de las manecillas del reloj) comenzamos a remover los ocupantes de las sillas como sigue: Primero eliminamos la persona con etiqueta m . Recursivamente, eliminamos al m -ésimo elemento de los elementos restantes. Este proceso continua hasta que las n personas han sido eliminadas. El orden en que las personas han sido eliminadas, se le conoce como la (n, m) -permutación de Josephus. Por ejemplo si $n = 7$ y $m = 3$, la $(7, 3)$ -permutación de Josephus es: $\{3, 6, 2, 7, 5, 1, 4\}$.
 - a) Supongamos que m es constante. De un algoritmo lineal para generar la (n, m) -permutación de Josephus.
 - b) Supongamos que m no es constante. Describa un algoritmo con complejidad $O(n \log n)$ para encontrar la (n, m) -permutación de Josephus.
3. Queremos ordenar una lista S de n enteros que contiene muchos elementos duplicados. Supongamos que los elementos de S sólo toman $O(\log n)$ valores distintos.
 - Encuentre un algoritmo que toma a lo más $O(n \log \log n)$ tiempo para ordenar S .
 - ¿Por qué esto no viola la cota inferior de $O(n \log n)$ para el problema de ordenación?
4. Dado un arreglo A de n números, queremos contestar la pregunta ¿Hay algún elemento de A que aparezca al menos $\frac{n}{3}$ veces? Encuentre un algoritmo lineal para resolver este problema.
5. Sea $A[1, \dots, n]$ un arreglo de números reales. Diseña un algoritmo que realice cualquier secuencia de las siguientes operaciones:

- $Add(i, y)$, suma el valor y al i -ésimo número.
- $Partial - sum(i)$, regresa la suma de los primeros i números, es decir,
 $Partial - sum(i) = A[1] + \dots + A[i]$

Considera que no hay ni inserciones ni borrado de elementos, sólo se cambia el valor de los números. Cada operación debe tomar $O(\log n)$ pasos. Puedes usar un arreglo de espacio extra de tamaño n .

6. Sea A un arreglo de n números enteros distintos. Suponga que A tiene la siguiente propiedad: existe un índice $1 \leq k \leq n$ tal que $A[1], \dots, A[k]$ es una secuencia incremental y $A[k+1], \dots, A[n]$ es una secuencia decremental.

- a) Diseña y analiza un algoritmo eficiente para encontrar k .
- b) Si no conoces el valor de n , cómo resuelves el problema.

7. You are a young scientist who just got a new job in a large team of 100 people (you the 101- st). A friend of yours who you believe told you that you have more honest colleagues than liars, and that that's all what he can tell you, where a liar is a person who can either lie or tell the truth, while an honest person is one who always tells the truth. Of course, you'd like to know exactly your honest colleagues and the liars, so that you decide to start an investigation, consisting of a series of questions you are going to ask your colleagues. Since you don't wish to look suspicious, you decide to ask only questions of the form "Is Mary an honest person?" and of course, to ask as few questions as possible. Can you sort out all your honest colleagues? What's the minimum number of questions you'd ask in the worst case? You can assume that your colleagues know each other well enough to say if another person is a liar or not. (Hint: Group people in pairs (X,Y) and ask X the question "Is Y honest?" and Y the question "Is X honest?". Analyze all the four possible answers. Once you find an honest person, you can easily find all the others. Challenge: can you solve this enigma asking less than 280 questions in total?)

Generalize the strategy above and show that given n people such that less than half are liars, you can sort them out in honest persons and liars by asking $\theta(n)$ questions.

8. Suponga que tenemos dos arreglos ordenados $A[1 \dots n]$ y $B[1 \dots n]$ y un entero k . Describe un algoritmo para encontrar el k -ésimo elemento en la unión de A y B . Por ejemplo, si $k = 1$, tu algoritmo debe regresar al elemento más pequeño de $A \cup B$; si $k = n$, tu algoritmo debe regresar la mediana de $A \cup B$. Puedes suponer que los arreglos no contienen duplicados. Tu algoritmo debe tener complejidad de tiempo $\Theta(\log n)$. Hint: Primero resuelve el caso especial $k = n$.
9. Considera que un río fluye de norte a sur con caudal constante. Suponga que hay n ciudades en ambos lados del río, es decir n ciudades a la izquierda del río y n ciudades a la derecha. Suponga también que dichas ciudades fueron numeradas de 1 a n , pero se desconoce el orden. Construye el mayor número de puentes entre ciudades con el mismo número, tal que dos puentes no se intersecten.