



Análisis de Algoritmos

Tarea 5: Búsquedas.

Profesora: María de Luz Gasca Soto

Ayudantes: Rodrigo Fernando Velázquez Cruz
Teresa Becerril Torres

Nombre: Alvaro Ramirez Lopez

Nº. de Cuenta.: 316276355

Correo: alvaro@ciencias.unam.mx



1. Problema 1

Consideremos el siguiente juego, entre dos personas: El jugador J_A piensa un número entero en un rango. El jugador J_B intenta encontrar tal número haciendo preguntas de la forma:

¿Es el número menor que x ? o ¿Es mayor que y ?

El objetivo es realizar el menor número de preguntas. Se supone que nadie hace trampa.

- Diseñar una buena estrategia para el juego... cuando el jugador J_A indica un rango específico, digamos de 1 a N . En este caso, ¿Cual resulta ser la complejidad del algoritmo? **Justificar.**
- Diseñar una buena estrategia para el juego... cuando el jugador J_A no indica el rango del número que pensó. ¿Cual es la complejidad del algoritmo propuesto? **Justificar.**

2. Problema 2

Dada un arreglo de n enteros $A[0, \dots, n-1]$, tal que $\forall i, 0 \leq i \leq n$, se tiene que $|A[i] - A[i+1]| \leq 1$; si $A[0] = x$ y $A[n-1] = y$, se tiene que $x < y$.

Diseña un algoritmo de búsqueda eficiente, de orden logarítmico, que localice al índice j tal que $A[j] = z$, para un valor dado de $z, x \leq z \leq y$. **Justificar.**

3. Problema 3

Se tiene un conjunto de N rocas, todas ellas de diferente tamaño, forma y consistencia. Rocas que se ven del mismo tamaño pueden tener peso muy diferente.

Un Experto desea convencer a un Jurado que una roca especial, de entre las N rocas dadas, es la de menor peso. El Experto si sabe los pesos de cada una de las rocas y el Jurado no. El Experto quiere mostrar que la roca especial es la de menor peso usando una balanza menos de $(N-1)$ veces. ¿Eso es posible? **Justifica.**

4. Problema 4

Consideremos el Algoritmo de Búsqueda por Interpolación.

- Presentar un ejemplo de al menos 400 datos para el cual el algoritmo termine la búsqueda (exitosa o no) en pocas iteraciones.
- Dar un ejemplo de, al menos, 400 datos para el cual el algoritmo termine la búsqueda (exitosa o no) en muchas iteraciones.

5. Problema Opcional

Supongamos que se tiene un programa que manipula textos muy grandes, como un procesador de palabras. El programa toma como entrada un texto, representado como una secuencia de caracteres y produce alguna salida.

Si en algún momento el programa encuentra un error del cual no puede recuperarse y además no puede indicar que error es o donde está, entonces la única acción que el programa toma es escribir ERROR y abortar el proceso.

Supongamos que el error es local, esto es, se tiene una cadena en particular del texto la cual, por alguna extraña razón, al programa no le gusta. El error es independiente del contexto en el cual aparece la cadena “ofensiva”. Diseñar una estrategia logarítmica para localizar la fuente del error.