Ayudantía 3

Análisis de Algoritmos

Universidad Técnico Federico Santa Maria

Carlos Lagos *carlos.lagosc@usm.cl*
Nangel Coello *nangel.coello@usm.cl*

11 de abril de 2024

Contenidos

Insertar

Sea $A[i..d]$ un arreglo de $d - i + 1 = n$ elementos de algún tipo, almacenado en la memoria de un computador RAM, cuya memoria es infinita en ambos extremos. Suponga el siguiente algoritmo que inserta un elemento x en la posición p del arreglo A (asumiendo siempre que $i \leq p \leq d$):

Algoritmo 2: INSERTAR($A[i..d]$, x, p)

```
if p ≤ ⌊(i+d)/2⌋ then
    j ← i − 1
    while j < p do
        A[j] ← A[j + 1]
        j ← j + 1
    end
else
    j ← d + 1
    while j > p do
        A[j] ← A[j − 1]
        j ← j − 1
    end
end
A[p] ← x
```

Obtenga el tiempo promedio de ejecución de dicho algoritmo. Haga todas las suposiciones que sean necesarias para un correcto análisis.

Contenidos

Conjunto de listas enlazadas

Suponga que necesita almacenar un conjunto de ℓ listas enlazadas, $L_1, \ldots, L_\ell$, sobre las que realizan las siguientes operaciones:

- `append(i, j)`: agrega el número entero j al final de la lista L_i (en tiempo de peor caso $\Theta(1)$).
- `suma(i)`: elimina todos los elementos de la lista L_i, y los reemplaza por la suma de los elementos que estaban en la lista (es decir, luego de ejecutar la operación la lista almacena un único entero).

Suponga que eliminar un nodo de la lista tiene costo de peor caso $\Theta(1)$. Dada una secuencia de n operaciones `append` y `suma` sobre listas inicialmente vacías, ¿Cuál es el costo amortizado por operación? Muestre el análisis y argumentos para su respuesta.

Contenidos

Mejor Caso vs Cota Inferior

¿Cuál es la diferencia entre el análisis de mejor caso y la cota inferior de un problema?.

Mejor Caso vs Cota Inferior

¿Cuál es la diferencia entre el análisis de mejor caso y la cota inferior de un problema?.

El análisis de mejor caso habla de qué tan eficiente puede ser **un algoritmo en particular**, mientras que la cota inferior nos indica qué tan difícil es resolver un **problema** y no habla de un algoritmo en particular.

Contenidos

Intervalos

Sea $A[1..n]$ un arreglo ordenado de elementos de un universo U sobre el que se ha definido una relación de orden total $\trianglelefteq$. Sobre A se quiere resolver el siguiente tipo de búsquedas:

Dados como entrada dos elementos $l, r \in U$ tal que $l \preceq r$, se quiere encontrar el rango $[i..d]$ (para $1 \leq i \leq d \leq n$) tal que para todo elemento $x \in A[i..d]$ se cumpla $l \trianglelefteq x \trianglelefteq r$. Por ejemplo, para el arreglo de enteros $A[1.,5] = [1, 4, 5, 7, 9]$, la respuesta para $l = 3$ y $r = 8$ es el rango [2.,4], el cual contiene los elementos 4, 5, y 7.

Use árboles de decisión para mostrar que $2 \log n - 1$ comparaciones es una cota inferior para dicho problema en el modelo de comparaciones.

$((\text{deciduous-tree}/k)!)^e$

Suponga el problema de mezclar k arreglos ordenados de $\frac{n}{k}$ elementos cada uno, para producir un único arreglo ordenado que contiene los n elementos de dichos arreglos. Use árboles de decisión para mostrar que $\Omega\left(nlg\frac{k}{e}\right)$ comparaciones es una cota inferior para dicho problema, siendo e el número de Euler.

- *Hint 1:* Hay $\frac{n!}{((n/k)!)^k}$ maneras distintas de mezclar k arreglos de tamaño n.
- *Hint 2:* $(n/e)^n \leq n! \leq n^n$.