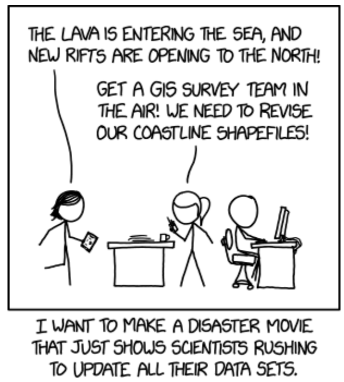
Certamen Recuperativo
Algoritmos y Complejidad

23 de agosto de 2018



1. Las funciones siguientes tienen $\sqrt{3}$ como punto fijo. ¿Cuál es la que converge más rápido cerca de $\sqrt{3}$?

$$g_a(x) = x(x+1) - 3 \qquad g_b(x) = \frac{3}{x}$$

(20 puntos)

2. Considere el algoritmo 1. ¿Cuántas veces come maní si se llama Beer(n)? ¿Cuánta cerveza bebe?

Algoritmo 1: El algoritmo Beer

```
procedure Beer(n)
  if n = 1 then
    Eat some peanuts
  else
    Pick m with 1 ≤ m ≤ n − 1
    Beer(m)
    Drink a pint of beer
    Beer(n − m)
  end
```

Pista: Las funciones $p(n)$ y $b(n)$ son lineales en n

(25 puntos)

3. Algoritmos voraces, backtracking y programación dinámica son generalmente aplicables a los mismos problemas. Explique cuándo aplicar cada uno de ellos, dando criterios claros.

(20 puntos)

4. La cadena *Empanadas to go* quiere instalarse a lo largo de Chile (que está en $\mathbb{R}^1$), posiciones posibles son $x_1, x_2, \dots, x_n$. La ganancia al instalarse en el punto i es g_i, nunca deben haber dos restaurantes a menos de d kilómetros de distancia. Sea $\mathrm{OPT}(k)$ la máxima ganancia considerando solo las posiciones $1, \dots, k$.

   a) Indique lo que nos interesa obtener, y un caso base.

   b) Escriba una recurrencia para OPT.

   c) Usando 4*a* y 4*b*, obtenga eficientemente lo buscado en pseudocódigo.

(30 puntos)

5. Dadas k listas ordenadas de n elementos, se pide intercalarlas para obtener una única lista ordenada. Analice las alternativas:

    a) Intercalar las primeras dos listas, intercalar el resultado con la tercera, y así sucesivamente hasta terminar el trabajo.

    b) Usar un esquema de dividir y conquistar.

    (20 puntos)

6. La empresa McWidget nstaló carteles que muestran el número de widget producidos. George Akeley gasta v en el viaje para actualizar todos los carteles y d por cada dígito que cambia (o sea, al cambiar de 21 a 22 gasta $c + d$, al cambiar 99 a 100 gasta $c + 3d$). Halle el costo amortizado por cada ítem producido.

    (30 puntos)

Teorema maestro

Al dividir un problema de tamaño n en a problemas de tamaño n/b que se resuelven recursivamente, el trabajo de dividir y combinar soluciones es $f(n) > 0$:

$$t(n) = at(n/b) + f(n) \qquad t(1) = t_1$$

cuya solución es:

$$t(n) = \begin{cases} \Theta(n^{\log_b a}) & f(n) = O(n^c) \text{ para } c < \log_b a \\ \Theta(n^{\log_b a}) & f(n) = \Theta(n^{\log_b a} \log^\alpha n) \text{ con } \alpha < -1 \\ \Theta(n^{\log_b a} \log\log n) & f(n) = \Theta(n^{\log_b a} \log^\alpha n) \text{ con } \alpha = -1 \\ \Theta(n^{\log_b a} \log^{\alpha+1} n) & f(n) = \Theta(n^{\log_b a} \log^\alpha n) \text{ con } \alpha > -1 \\ \Theta(f(n)) & f(n) = \Omega(n^c) \text{ con } c > \log_b a \text{ y } af(n/b) < kf(n) \text{ para } n \text{ grande con } k < 1 \end{cases}$$

Algunas series notables

En lo que sigue, $m, n \in \mathbb{N}$, $k \in \mathbb{N}_0$ y $\alpha, \beta \in \mathbb{C}$.

$$\frac{1}{1-az} = \sum_{n \geq 0} a^n z^n \qquad \frac{1-z^{m+1}}{1-z} = \sum_{0 \leq n \leq m} z^n$$

$$(1+z)^\alpha = \sum_{n \geq 0} \binom{\alpha}{n} z^n \qquad \binom{\alpha}{n} = \frac{\alpha^{\underline{n}}}{n!} = \frac{\alpha(\alpha-1)\cdots(\alpha-n+1)}{n!} \qquad \binom{n}{k} = \frac{n!}{k!(n-k)!}$$

$$\binom{1/2}{n} = \frac{(-1)^{n-1}}{n 2^{2n-1}} \binom{2n-2}{n-1} \quad (\text{si } n \geq 1) \qquad \binom{-1/2}{n} = \frac{(-1)^n}{2^{2n}} \binom{2n}{n} \qquad \binom{-n}{k} = (-1)^k \binom{n+k-1}{n-1}$$

$$\frac{1}{(1-z)^{n+1}} = \sum_{k \geq 0} \binom{n+k}{n} z^k = \frac{1}{n!} \sum_{k \geq 0} (k+1)(k+2)\cdots(k+n) z^k = \frac{1}{n!} \sum_{k \geq 0} (k+1)^{\overline{n}} z^k$$

$$\sum_{n \geq 0} \binom{n}{k} z^n = \frac{z^k}{(1-z)^{k+1}}$$