CC40A: Diseño y Análisis de Algoritmos

Profesor: Gonzalo Navarro

Examen - Julio de 2008

P1

Para las siguientes afirmaciones, responda V ó F, argumentando en a lo sumo 3 líneas. Respuestas sin la correcta argumentación no valen.

1. Los algoritmos probabilísticos pueden resolver algunos problemas (con cierta probabilidad de error) en un tiempo imposible para un algoritmo determinístico.
2. Todo problema de optimización admite un esquema de aproximación completamente polinomial.
3. Un algoritmo paralelo que usa p procesadores puede permitir un speedup de a lo sumo p.
4. Para mostrar que la complejidad de un problema es $\Omega(f(n))$ basta con mostrar un algoritmo que lo resuelva y que demore más de $O(f(n))$ en alguna entrada.
5. Si existe un algoritmo online k-competitivo para un cierto problema, entonces también existe una k-aproximación para el problema.
6. La programación dinámica puede permitir convertir un algoritmo de tiempo exponencial a uno polinomial.

P2

Cuando el Profesor Locovich era joven y sus neuronas funcionaban mejor, usted fue su alumno y aprendió sobre el problema P: el Profesor le mostró un algoritmo correcto que lo resolvía en tiempo $O(n^{4/3})$ en el peor caso, y también le demostró una cota inferior de $\Omega(n \log n)$ en el peor caso. El Profesor Locovich ha seguido trabajando intensamente en el problema P, y está considerando varias publicaciones para enviar a una prestigiosa revista. El problema es que el Profesor ya está un poco senil, por lo cual algunos de sus artículos son más que cuestionables. Para salvar lo que queda de su prestigio, indíquele al Profesor cuáles de los siguientes artículos necesariamente tienen algún error y cuáles no tienen ninguna contribución apreciable a partir del título (argumente). Las afirmaciones se refieren al peor caso a menos que se indique lo contrario. Cuando se habla de ε significa que la afirmación vale para cualquier constante $\varepsilon > 0$.

1. “El problema P es $\Omega(n^2/\log n)$”.
2. “Un algoritmo $O(n \log\log n)$ en promedio para P”.
3. “El problema P es $\Omega(n(\log\log n)^2)$”.
4. “Un algoritmo $O(\frac{1}{\varepsilon} n^{1+\varepsilon})$ para P”.
5. “Una $(1+\varepsilon)$-aproximación de costo $O(n^{1/\varepsilon})$ para P”.
6. “Un algoritmo para P que funciona con probabilidad $1-\varepsilon$, a costo $O(n^{2/(1+\varepsilon)})$”.

P3

El Profesor Locovich está empeñado en determinar cuál es el último piso de un edificio desde el que puede tirarse un huevo de avestruz sin romperse. (No es necesario insistir sobre la tremenda trascendencia del resultado para la Ciencia.) El edificio tiene n pisos, pero dado el alto costo de los huevos de avestruz, el Profesor sólo dispone de k huevos. Debe usted diseñar un algoritmo para resolver el problema. El costo de su algoritmo es la cantidad de veces que necesita tirar el huevo para responder.

1. Demuestre que si $k = 1$, la complejidad del problema es n.
2. Encuentre un algoritmo $O(\sqrt{n})$ para $k = 2$.
3. Generalice para obtener $O(kn^{1/k})$.
4. Obtenga la mejor complejidad posible si no hay limitaciones en k. Cuántos huevos necesita para obtener esta complejidad?
5. Suponga ahora que el Profesor no está en estado físico para realizar él mismo el experimento, pero dispone de p alumnos, de modo que puede paralelizar las tiradas. Ahora el tiempo se mide en *turnos* de tiradas de huevos: en cada turno puede tirar hasta p huevos a la vez. Muestre que puede conseguir tiempo $O(1)$ con $k = n = p$. Calcule speedup y eficiencia (siempre suponiendo $k = n$).
6. Proponga una variante que consiga tiempo $O(\log_p n)$ usando p alumnos. Calcule speedup y eficiencia, y cuántos huevos necesita. ¿Cuál es la mejor eficiencia que puede conseguir manteniendo tiempo $O(1)$?