UNIVERSIDAD DE CHILE
ESCUELA DE INGENIERIA
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACION

**Matemáticas Discretas para la Computación - CC3101**
**Examen - Semestre Primavera 2012**

## P1 (1.5 pt)

Considere el siguiente conjunto de strings $\mathcal{B}$ sobre el alfabeto $\{a, b\}$, definido recursivamente.

- $\varepsilon \in \mathcal{B}$ (la cadena vacía).
- Si $S \in \mathcal{B}$ y $S' \in \mathcal{B}$, entonces $aSbS' \in \mathcal{B}$ (la concatenación de caracteres y cadenas).

Demuestre que $\mathcal{B}$ es el conjunto de cadenas *balanceadas*, es decir en las que hay tantas $a$'s como $b$'s y en ningún prefijo tienen más $b$'s que $a$'s.

**Solución:** Primero demostraremos que toda cadena generada según esas reglas cumple con ser balanceada, y luego que toda cadena balanceada se puede generar con esas reglas.

Para la primera parte, usamos inducción estructural. $\varepsilon$ es balanceada. Supongamos ahora que $S$ y $S'$ lo son. Entonces claramente $aSbS'$ tiene tantas $a$'s como $b$'s. Más aún, como $S$ es balanceada, ningún prefijo de $aSb$ puede tener más $b$'s que $a$'s. Como $aSb$ tiene tantas $a$'s como $b$'s, y $S'$ es balanceada, tampoco un prefijo de $aSbS'$ puede tener más $a$'s que $b$'s.

Para la segunda parte, tomemos el menor prefijo no vacío $P$ de una cadena balanceada $S$ tal que $P$ tenga tantas $a$'s como $b$'s. Existe $P$ porque $S$ tiene tantas $a$'s como $b$'s. Como $P$ no puede partir con una $b$, pues sino $S$ no sería balanceada, parte con una $a$, y entonces hay más $a$'s que $b$'s antes de $P$. Entonces $P$ debe terminar con una $b$ para llegar a diferencia cero. Por ello, $P$ es de la forma $aP'b$, donde $P'$ debe ser balanceada. El resto de $S$ parte con tantas $a$'s como $b$'s y debe cumplir con ser balanceada, por lo que $S$ se puede escribir como $aP'bS'$.

## P2 (1.5 pt)

Considere el siguiente algoritmo para encontrar el $k$-ésimo elemento de un arreglo desordenado $A[1, n]$:

1. Cortar el arreglo en grupitos de 5 elementos y calcular la mediana de cada grupito.
2. Calcular la mediana $m$ de las $n/5$ medianas, usando recursión.
3. Usar esa mediana para separar el arreglo entre los menores y mayores que $m$.
4. Sea $t$ la cantidad de elementos menores que $m$. Si $k = t + 1$, entonces responder $m$ y terminar. Si $k \leq t$ continuar recursivamente en el conjunto de los menores. Sino, continuar recursivamente en el grupo de los mayores, ahora con $k - t - 1$ en vez de $k$.

Se pide:

1. Demostrar que $m$ debe ser mayor que $3/10$ de los elementos de $A$, y menor que otros $3/10$.
2. Establecer una recurrencia para el costo $T(n)$ del algoritmo, y probar que $T(n) = O(n)$.

**Solución:** Para 1., deben ver que la mediana de las medianas $m$ es mayor que la de la mitad de los grupitos, y cada mediana es mayor que 2 de los 5 elementos de su grupito. Hay $n/5$ grupitos, $n/10$ de ellos son la mitad, y $m$ es mayor que 3 elementos de cada grupito (su mediana y los otros 2), con lo que $m$ es mayor que $3n/10$ de los elementos. Simétricamente con ser menor.

Para 2., tenemos un procesamiento $O(n)$ para todo lo que hay que hacer, una recursión sobre grupitos, $T(n/5)$, y una llamada recursiva sobre a lo más $7n/10$ elementos, por lo que $T(n) = cn + T(n/5) + T(7n/10)$. No da directo para el teorema maestro, pero pueden por ejemplo usar inducción con $T(n) \leq dn$, entonces da $T(n) \leq cn + (9/10)dn \leq dn$ sii $d \geq 10c$.

## P3 (1.5 pt)

## P4 (1.5 pt)