Auxiliar 8 - “Análisis Amortizado y Universos Discretos y Finitos”

Profesores: Pablo Barceló
Gonzalo Navarro
Auxiliar: Dustin Cobas

P1. Next Smaller Value

Dado un arreglo $A[1..n]$, se quiere construir otro arreglo $NSV[1..n]$, llamado “next smaller value”. Concretamente, $NSV[i] = \min\{j > i,\ A[j] < A[i]\}$, suponiendo $A[n+1] = -\infty$.

Diseñe un algoritmo de tiempo $O(n)$ para construir NSV y use técnicas de análisis amortizado para demostrar que su complejidad es $O(n)$.

P2. PLCP

El arreglo $LCP[2\dots n]$ (“longest common prefix”) para un texto $T[1\dots n]$ se define en función del arreglo de sufijos $A[1\dots n]$ de T: $LCP[i] = lcp(T[A[i]\dots n], T[A[i-1]\dots n])$, donde $lcp(X,Y)$ es el largo del mayor prefijo común a X e Y. Por ejemplo, si $T[1\dots 12] = abracadabra\$$, entonces $A[1\dots 12] = \langle 12, 11, 8, 1, 4, 6, 9, 2, 5, 7, 10, 3\rangle$ y $LCP[2\dots 12] = \langle 0, 1, 4, 1, 1, 0, 3, 0, 0, 0, 2\rangle$.

Definimos el arreglo permutado $PLCP[1\dots n-1]$ como $PLCP[j] = LCP[A^{-1}[j]]$, es decir, los sufijos se recorren en orden de texto. En nuestro ejemplo, $PLCP[1\dots 11] = \langle 4, 3, 2, 1, 0, 1, 0, 1, 0, 0, 0\rangle$.

a) Demuestre que $PLCP[j] \geq PLCP[j-1] - 1$.

b) Para calcular $PLCP$ se propone repetir para $j = 1$ hasta n: calcular directamente $PLCP[j] = lcp(T[j\dots n], T[A[A^{-1}[j]-1]\dots n])$ carácter a carácter, y luego pasar a $j+1$. La única gracia es que, por la propiedad del punto anterior, sabemos que los sufijos coincidirán en los primeros $l = PLCP[j-1] - 1$ símbolos, por lo que podemos calcular $PLCP[j] = l + lcp(T[j+l\dots n], T[A[A^{-1}[j]-1]+l\dots n])$.

Demuestre que el costo total de este procedimiento es $\mathcal{O}(n)$.

Soluciones

P1. Next Smaller Value

Se inicializa un stack S con elementos previos aún no resueltos y se recorre A de izquierda a derecha. Para cada $A[i]$, mientras $A[i] < A[S.top]$, $NSV[S.top] \leftarrow i$ y $pop(S)$. Finalmente, $push(i)$. Esto es lineal por el mismo argumento del multipop visto en clase.

P2. PLCP

a) Solo nos preocuparemos del caso $P[j-1] > 1$ pues sabemos que $PLCP \geq 0$.

Notemos que $PLCP[j-1]$ es el largo del prefijo más largo entre el $j-1$-ésimo sufijo y su anterior lexicográfico, es decir, si w es el $j-1$-ésimo sufijo, este coincide con su anterior lexicográfico, digamos w_a en los primeros $PLCP[j-1] > 1$ caracteres. Si a w y w_a le sacamos sus primeras letras obtendremos v y v_a ambos sufijos del texto que comparten las primeras $PLCP[j-1]-1$ letras y siendo v el j-ésimo sufijo. Finalmente, el anterior lexicográfico al j-ésimo sufijo v debe ser $\geq v_a$ en términos lexicográficos y por lo tanto el número de caracteres que coinciden entre v y su anterior lexicográfico debe ser $\geq PLCP[j-1]-1$, que es lo que se pedía demostrar.

b) De la primera ecuación se puede ver que el primer $PLCP$ se puede calcular en $\mathcal{O}(n)$. De la última ecuación se deduce que para el resto de $PLCPs$, calcular $PLCP[j]$ cuesta $\mathcal{O}(1) + PLCP[j] - PLCP[j-1]$, siendo lo último el costo del lcp. Finalmente, sumando los costos de todos los $PLCP$ entre 2 y $n-1$ la suma "telescopea" y queda $PLCP[n-1] - PLCP[1]$ lo que es $\mathcal{O}(n)$.