



UNIVERSIDAD NACIONAL DE ROSARIO
FACULTAD DE CIENCIAS EXACTAS, INGENIERÍA Y AGRIMENSURA
Licenciatura en Ciencias de la Computación
Estructuras de datos y algoritmos II

Especificación de costos

Alumnos:

CRESPO, Lisandro (C-6165/4)
MISTA, Agustín (M-6105/1)

Docentes:

JASKELIOFF, Mauro
RABASEDAS, Juan Manuel
SIMICH, Eugenia

1 de Junio de 2015

Implementación con listas

filterS

Para implementar la función `filterS`, consideramos la función `filter` presente en el preludio, y paralelizamos el llamado recursivo para mejorar el rendimiento si los predicados que `filterS` evalúa son costosos de calcular. Luego podemos considerar a `filterS` como la siguiente recurrencia:

$$T(n) = T(n-1) + f(n)$$

Donde $f(n)$ es el costo de evaluar cada predicado, además del costo de las comparaciones, que consideramos constantes. Puede verse que en esta implementación, el paralelizar las operaciones no mejora el problema de tener que recorrer todo el arreglo de forma secuencial. Resolviendo la recurrencia tenemos entonces:

$$W(\text{filterS} \oplus s) \in O\left(|s| + \sum_{i=0}^{|s|-1} W(f(i))\right)$$

$$S(\text{filterS} \oplus s) \in O\left(|s| + \max_{i=0}^{|s|-1} (S(f(i)))\right)$$

Finalmente, si consideramos que $f(n) \in O(1)$ resulta:

$$W(\text{filterS} \oplus s) \in O(n)$$

$$S(\text{filterS} \oplus s) \in O(n)$$

showtS

Para el caso de `showtS`, la implementación mediante listas es poco eficiente dado que para poder partir la lista en dos mitades en el caso de que existan dos o más elementos, se necesita conocer el tamaño de la misma, lo cual resulta en un coste lineal tanto para el trabajo como para la profundidad. Por lo tanto:

$$W(\text{showtS} \oplus s) \in O(n)$$

$$S(\text{showtS} \oplus s) \in O(n)$$

redusceS

scanS

Implementación con arreglos persistentes

filterS

showtS

redusceS

scanS

Bla Bla Bla...