

# CC4102/CC53A - Diseño y Análisis de Algoritmos

## Auxiliar 5

Prof. Gonzalo Navarro; Aux. Mauricio Quezada

2 de Octubre de 2012

1. Considere una tabla de tamaño  $size$  y almacenando  $num$  elementos, que se va llenando con inserciones y debemos realocarla cuando no queda espacio para insertar ( $size = num$ ). Al realocar la tabla debemos pagar un costo de  $O(num)$ .
  - (a) Muestre que el costo amortizado de la inserción es constante al duplicar la tabla cuando se llena.
  - (b) Considere que también ocurren borrados, y no queremos que  $size$  sea mucho mayor que  $num$ , para no desperdiciar espacio. Muestre que duplicar la tabla cuando se llena, y reducirla a la mitad cuando  $num = size/2$  no consigue un costo amortizado constante.
  - (c) Considere la estrategia de duplicar cuando la tabla se llena, y reducirla a  $size/2$  cuando  $num = size/4$ . Demuestre que esta estrategia obtiene un costo amortizado constante usando la siguiente función potencial:

$$\Phi = \begin{cases} 2 \cdot num - size & \text{si } 2 \cdot num \geq size \\ size/2 - num & \text{si } 2 \cdot num < size \end{cases}$$

2. Muestre que cualquier secuencia de  $m$  operaciones CrearConjunto, Find y Link, donde todos los Link aparecen antes de cualquier Find, toma tiempo  $O(m)$  si se usa Compresión de caminos y Unión por rank. ¿Qué pasa en la misma situación si sólo se usa Compresión de caminos?