INF221 – Algoritmos y Complejidad

Clase #26
Análisis Amortizado

Aldo Berrios Valenzuela

15 de noviembre de 2016

1. Análisis Amortizado

Idea: Analizar *secuencias* de operaciones.

Definición 1.1. El *costo amortizado* de una operación es el costo de una secuencia de operaciones dividida por su largo.

Ejemplo 1.1 (Arreglo dinámico)**.** Un stack se puede representar mediante un arreglo, operaciones son:

```
typedef item ...;
item A[...];
static int top = 0;

void push(item A[], item a) {
  A[top++] = x;
}
item pop(item A[]) {
  return A[--top];
}
```

Nótese que falta una serie de burocracias que consideran cuando la operación se sale de rango y otros, pero sólo nos interesa lo de arriba.

¿Qué hacer si el arreglo se llena?. Bienvenido a `realloc(3)` ...

Medida de costo: Nº de items copiados.

Si cada vez que llegamos al tamaño del arreglo lo agrandamos en 1, el costo de n `push` es $n(n+1)/2$, costo medio es $(n+1)/2$ `/* es por el realloc() */`. Inaceptable ...

El costo de n push es la suma:

$$\begin{aligned} 1+2+1+4+1+1+1+8+\cdots &= n + \sum_{0\le k\le \lfloor \log_2 n\rfloor} 2^k \\ &= n + 2^{\lfloor \log_2 n\rfloor+1} - 1 \\ &\le n+2n \\ &= 3n \end{aligned}$$

Por lo tanto, costo amortizado < 3. `/* nos interesa el costo de nuestro programa completo, no el costo de operación por operación */`

Análisis agregado: Se calcula el costo total (peor caso) de n operaciones se divide por n. ∎

1.1. Método contable

Idea: Cada operación paga por *su* costo y ahorra una cantidad para pagar sobre costos de operaciones futuras. Hay que asegurarse que el saldo no se haga negativo ...

1.2. Función Potencial

Idea: La estructura tiene un *potencial* $\phi(s)$ (s: estado de la estructura). Operaciones $\sigma_1, \sigma_2, \dots, \sigma_n$ llevan a la estructura de s_0 a s_1, a $s_2, \dots$, a s_n.

El costo de cada operación es c_i, su costo amortizado es:

$$a_i = c_i + \phi(s_i) - \phi(s_{i-1})$$

Costo amortizado de la secuencia (¡Viva la telescópica!) es:

$$\begin{aligned} \sum_i a_i &= \sum_i \big(c_i + \phi(s_i) - \phi(s_{i-1})\big) \\ &= \sum_i c_i + \phi(s_n) - \phi(s_0) \end{aligned}$$

Si (¡Caso común!) $\phi(s_n) \geq \phi(s_0)$:

$$\sum_i a_i \geq \sum_i c_i \rightsquigarrow \frac{\sum_i a_i}{n}$$

es costo amortizado de los c_i.