Topic: CSC263 Monday Lecture — Feb 26, 2018

Amortized Complexity: CLRS 17

- Given a data structure with operations,
  - $T(n)$ = worst case (max) cost of doing seq of n operations
  - $A(n) = T(n)/n$ = average cost of doing an op in worst case
    = "amortized" cost of an operation

- Example: incrementing a binary counter.
  A : | k-1 | | | ... | 1 | 0 | // initially set to zero.
  INC(A): $A \rightarrow A+1$
  Cost of INC(A) is # of bits that must flip.

- Brute Force Method - "Aggregate Analysis"
  Let's explore what happens when $K = 4$...

0000
0001
0010
0011
0100
0101
0110
0111
1000
1001
1010
1011
1100
1101
1110
1111
0000

	flips for n incs:
Bit 0 flips every $2^0 = 1$ increments.	n
Bit 1 flips every $2^1 = 2$ increments	$\lfloor n/2 \rfloor$
Bit i flips every 2^i icrements.	$\lfloor n/2^i \rfloor$
Bit $k-1$ flips every 2^{k-1} increments.	$\lfloor n/2^{k-1} \rfloor$

Total bit flips $= \sum_{i=0}^{k-1} \lfloor n/2^i \rfloor < \sum_{i=0}^{\infty} n/2^i$

Note $\sum_{i=0}^{\infty} n/2^i = n \sum_{i=0}^{\infty} (1/2)^i$.

Note $\sum_{i=0}^{\infty} x^i = \frac{1}{1-x}$ $\therefore$ $\sum_{i=0}^{\infty} (\frac{1}{2})^i = \frac{1}{1-1/2} = 2$.

Then $\sum_{i=0}^{\infty} n/2^i = 2n$. So $T(n) < 2n$.

Then $A(n) < T(n)/n = 2$

- **Accounting Method**

  // explain concept of charging fees for all incrs, overcharging for smaller incrs to cover all expensive ones

Prev Ex:
- Charging 2 per bit flip means you can save 1 for when that bit needs to be reset back to zero.
- Store +1 credit with each bit set at 1.
- Then any operation can be executed with initial 2 and all stored credits.

INVARIANT:

Total charges (so far) on any sequence of OPS $\geq$ Total actual cost of doing these OPS

Ex: Dynamic Tables: Contiguous space in main memory.

Given; T: table; OPS: INSERT, DELETE.

- [a | b | c |] ; Load factor $\alpha(T) = 3/4$

  $\hookrightarrow$ [a | b | c | d] ; insert(d) $\therefore$ $\alpha(T) = 4/4$.
- When $\alpha(T) = 1$ and an insert occurs, allocate a new table in memory that is twice the size of the old one.
- Cost of that insert is cost of copying all from old table and cost of single element insert.
- 2^ith insert for $\forall i \in \mathbb{N}$ costs 2^i