Jin Wei CS3230

$\sum^{n} i = \frac{n(n+1)}{2}$

AP $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$ $\sum i^2 = \frac{n(n+1)(2n+1)}{6}$

GP $\sum_{i=0}^{n-1} ar^i = \frac{ar^n - a}{r-1}$ $H_n = \sum_{k=1}^{n} \frac{1}{k} = \ln n + O(1)$

$\sum_{i=0}^{\infty} ar^i = \frac{a}{1-r}$ $\sum_{k=1}^{\infty} \frac{1}{k^2} = \frac{\pi^2}{6}$

$\binom{n}{r} = \frac{n!}{r!(n-r)!}$

$\binom{n}{2} = \frac{n(n-1)}{2}$

$nP_r = \frac{n!}{(n-r)!}$

height of $\sqrt{n}$ is $\lg\lg n$

→ lowest is $n^{1/2^k}$

$n^{1/2^k} = 2$

$\frac{1}{2^k} \lg n = \lg 2$

$\lg n = 2^k$

$k = \lg\lg n$

$(a^m)^n = a^{mn}$ ⋮ $a^m a^n = a^{m+n}$ ⋮ $e^x \geq 1 + x$ ⋮ $\lg(AB) = \lg A + \lg B$

$\lg_a 1 = 0$ ⋮ $\lg_a a = 1$ ⋮ $a^{\lg_a b} = b$ ⋮ $a^{\lg_b c} = c^{\lg_b a}$ ⋮ $\lg_b a = \frac{\lg_d a}{\lg_d b}$

$\lg(n!) = \Theta(n \lg n)$ $n! = \sqrt{2\pi n}\left(\frac{n}{e}\right)^n\left(1 + \Theta\left(\frac{1}{n}\right)\right)$ $(\lg n)! = 2^{\Theta(\lg n \lg\lg n)}$

$\lg\lg n << \lg n << \lg^2 n << \sqrt{n} << n^x << n^{\lg n} << (\lg n)! << x^n << (\lg n)^n << n! << n^n$

$O(g(n)) = \{ f(n) : \exists c > 0, n_0 > 0 \text{ st } 0 \leq f(n) \leq c g(n) \ \forall n \geq n_0 \}$

$\Omega(g(n)) = \{ f(n) : \exists c > 0, n_0 > 0 \text{ st } 0 \leq c g(n) \leq f(n) \ \forall n \geq n_0 \}$

$\Theta(g(n)) = \{ f(n) : \exists c_1 > 0, c_2 > 0, n_0 > 0 \text{ st } 0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n) \ \forall n \geq n_0 \}$

$o(g(n)) = \{ f(n) : \forall c > 0, \exists n_0 > 0 \text{ st } 0 \leq f(n) < c g(n) \ \forall n \geq n_0 \}$

$\omega(g(n)) = \{ f(n) : \forall c > 0, \exists n_0 > 0 \text{ st } 0 \leq c g(n) < f(n) \ \forall n \geq n_0 \}$

NOTE: $2^{5n} \neq O(2^n)$, trigo is good counter example

L'hopital if num = den = 0 or ±∞

$\lim_{n\to\infty} \frac{f(n)}{g(n)} \Rightarrow f(n) = \nabla(g(n))$

$\nabla = 0 \Rightarrow$ small o

$\nabla < \infty \Rightarrow$ big O

$0 < \nabla < \infty \Rightarrow \Theta$

$\nabla > 0 \Rightarrow \Omega$

$\nabla = \infty \Rightarrow \omega$

degree-k $= O(n^k) = o(n^{k+1}) = \omega(n^{k-1})$

✓ Transitive [all 5] $f(n) = \nabla(g(n))$ & $g(n) = \nabla(h(n))$ $\Rightarrow f(n) = \nabla(h(n))$

✓ Reflexive [$O \Omega \Theta$] $f(n) = \nabla(f(n))$

✓ Symmetry for Θ [Θ]

✓ Complementarity for (O, Ω), (o, ω)

$f(n) = O(g(n)) \iff g(n) = \Omega(f(n))$

Iterative

$i \Rightarrow i+1$

- Correctness : loop invariant
  - initialisation, maintainence, termination
- Runtime : obvious

Recursive

- Correctness : Base case + Strong induction
- Runtime : Recurrence relation

1. Recursion Tree (show ≤ & ≥ if bounding!)
2. Master Method (CANNOT differ by lg n)
3. Substition Method
   - → guess + induction
   - → use more terms
   - → use subtraction

$T(n) = aT\left(\frac{n}{b}\right) + f(n)$

$a \geq 1$, $b > 1$, $n^{\lg_b a}$

Case 1: $f(n) = O(n^{\lg_b a - \varepsilon})$, $\varepsilon > 0$

$f(n)$ grows polynomially slower than $n^{\lg_b a}$ by n^ε [leaf-heavy]

$\Rightarrow \Theta(n^{\lg_b a})$

i.e. $\lg n = O(n^{1-\varepsilon})$

$n \lg n = O(n^{2-\varepsilon})$

$\frac{n^2}{\lg n} \notin O(n^{2-\varepsilon})$

Case 2: $f(n) = \Theta(n^{\lg_b a} \lg^k n)$, $k \geq 0$

$f(n)$ grows at similar rates with $n^{\lg_b a}$

[similar work on each level]

$\Rightarrow \Theta(n^{\lg_b a} \lg^{k+1} n)$

Case 3: $f(n) = \Omega(n^{\lg_b a + \varepsilon})$, $\varepsilon > 0$

$f(n)$ grows polynomially faster than $n^{\lg_b a}$ by n^ε [root-heavy]

AND $f(n)$ satisfies regularity condition

that $af\left(\frac{n}{b}\right) \leq cf(n)$ for some $c < 1$ (state c or range of c)

$\Rightarrow \Theta(f(n))$

i.e. $n \lg n$ does NOT satisfy regularity condition for $a=2, b=2$

Random

Las Vegas: Always correct, random runtime

Monte Carlo: Sometimes incorrect, same runtime

Average running time: for non-randomized algorithms that depend on input

→ need to know distribution of input

Expected running time: for randomized algorithms

→ use $E[X]$

$P(A_k \mid B) = \frac{P(A_k) P(B \mid A_k)}{\sum_{i=1}^{n} P(A_i) P(B \mid A_i)}$

for n=2: $P(A \mid B) = \frac{P(A) P(B \mid A)}{P(A) P(B \mid A) + P(A') P(B \mid A')}$

Bernoulli Trials: p -- success, $q = 1 - p$ -- failure (geometric distribution)

$P_r[X = k] = (1-p)^{k-1} \cdot p$

$E[X] = \sum_{k=1}^{\infty} k \cdot P_r(X = k)$

$= \sum_{k=1}^{\infty} k \cdot (1-p)^{k-1} \cdot p = \frac{1}{p}$

Indicator Random Variable $X_i = \begin{cases} 1, & \text{if } A \text{ occurs} \\ 0, & \text{otherwise} \end{cases}$ $E[X_i] = P_r[A]$

$E[XY] = E[X]E[Y]$ if X and Y are independent

Hashing

→ CS3230 only cover chaining

(by PHP) for $h : [U] \to [M]$, if $U \geq (N-1)M + 1$, then $\exists$ set of N elements that collide

pairwise indep. ⇒ universal, ⇍

Universal H if $\forall$ distinct $x, y \in U$: $\Pr_{h \sim H}[h(x) = h(y)] \leq \frac{1}{M}$

Uniform H if $\forall x \in U, k \in [M]$: $\Pr_{h \sim H}[h(k) = i] \leq \frac{1}{M}$

let $H = \{h_1, h_2, \ldots, h_n\}$, $h_1(x) = 1$, $h_2(x) = 2$, … → NOT universal [all $x \in U$]

pairwise independent H if $\forall$ distinct $x, y \in U$, $i_1, i_2 \in [M]$: $\Pr_{h \sim H}[h(x) = i_1, h(y) = i_2] = \frac{1}{M^2}$

universal:

	a	b
h_1	0	0
h_2	0	1

	a	b
h_1	0	1
h_2	1	0

	a	b
h_1	0	0
h_2	1	0
h_3	0	1

NOT universal:

	a	b
h_1	0	0
h_2	1	1

	a	b	c
h_1	0	0	1
h_2	1	1	0
h_3	1	0	1

for universal:

- expected num of collisions for any N elements $< \frac{N-1}{M}$ ∵ each other elem $\leq \frac{1}{M}$
- expected cost of n operations is $O(n)$ if $M > N$
- expected num of pairs (i, j) of collisions after adding $x_1, \ldots, x_N$ and $M < N$ is $\leq N \cdot 1 + N(N-1) \cdot \frac{1}{M} < 2N$ (self-collision)
- expected max. load (max. elem in one slot) $\leq \sqrt{2N} \leq O(\sqrt{N})$

∵ total collisions $\geq (\text{max. load})^2$ ∵ $\binom{\text{max. load}}{2}$ collisions [everything collide in 1 slot]

$\{h_A : A \in \{0,1\}^{m \times u}\}$ is universal where $h(x) = Ax \pmod 2$ (x: col vector)

2-level hashing: $O(1)$ ops, $O(N)$ space

if N_1 = number of X_i and N_2 = number of Y_i:

$N_1 = \sum_{i=1}^{n} X_i$ $N_1^2 = \sum_{i=1}^{n} X_i^2 + 2\sum_{1 \leq i < j \leq n} X_i X_j$ $N_1 N_2 = \sum_{1 \leq i \leq n, 1 \leq j \leq n} X_i Y_j$

Amortized Analysis

→ most op cheap
→ rare op expensive

* NO probability

1. Aggregate Analysis
   - → average cost of worst sequence of operations
   - → simple to understand but tedious in practice
2. Accounting (Banker's) Method
   - → set $c(i)$ for each op st $\sum t(i) \le \sum c(i)$ (← true after any i) * only upper bound
   - → invariant: always enough credit to pay off any next op
   - → cheap $t(i)$ → overpay $c(i)$
   - → expensive $t(i)$ → lower $c(i)$

   } set $c(i)$ as low as possible
   1. tighter bound
   2. analysis easier (credit nearer to 0)
3. Potential Method
   - → cheap op → $\Delta\phi(i) > 0$
   - → expensive op → $\Delta\phi(i) < 0$
   - → $\phi(i)$ is "credit amount" after i-th op

   * state & check
     1. $\phi(0) = 0$
     2. $\phi(i) \geq 0 \;\; \forall i$ to be valid ϕ function

   $c(i) = t(i) + \phi(i) - \phi(i-1)$

   amortized cost = actual cost + $(\phi(n) - \phi(0))$

   * is an upper bound

Misc

Comparison-based Sorting
- ↳ $n!$ universe
- ↳ cut in half every query
- ↳ $\Omega(\lg(n!)) = \Omega(n \lg n)$

Common Recurrences

$T(\frac{n}{2}) + O(1)$	$O(\lg n)$
$2T(\frac{n}{2}) + O(1)$	$O(n)$
$T(\frac{n}{2}) + O(n)$	$O(n)$
$2T(\frac{n}{2}) + O(n)$	$O(n \lg n)$
$2T(\frac{n}{2}) + O(n \lg n)$	$O(n \lg^2 n)$
$T(\sqrt{n}) + O(\lg n)$	$O(\lg n)$
$T(n-1) + O(\sqrt{n})$	$O(n\sqrt{n})$
$2T(\frac{n}{2}) + \frac{n}{\lg n}$	$\Theta(n \lg\lg n)$ ← H_n
$2T(\sqrt{n}) + O(1)$	$\Theta(\lg n)$
$2T(\sqrt{n}) + O(\lg n)$	$\Theta(\lg n \lg\lg n)$

$$\int_{i=1}^{x} f(i)\,di \le \sum_{i=1}^{x} f(i) \le \int_{i=1}^{x+1} f(i)\,di$$

$$\left(1 - \frac{1}{n}\right)^n \approx \frac{1}{e}$$

$$\left(1 + \frac{a}{n}\right)^{bn+c} \approx e^{ab}$$