COP3502H 1/22/26

Big Oh

$O(f(n)) + O(g(n)) = O(\max(f(n), g(n)))$

$3n^3 + 2n = O(n^3)$

functions in n

$n^k < n^m$ iff $k < m$

lg (poly) grows slower than any poly

$\lg n^{100000}$ smaller than $n^{0.001}$

$= 100{,}000 \lg n = O(\lg n)$

Hierarchy

$\lg n \times \lg n$

$O(1), O(\lg n), O(\lg^2 n), O(\text{poly}), O(2^n), O(3^n), O(n!), O(n^n)$

$O(\sqrt{n}), O(n), O(n\lg n), O(n\sqrt{n}), O(n^2),$

$O(\frac{n}{\lg n})$

$\log_8 n = \frac{\log_2 n}{\log_2 8} = \frac{1}{3}\log_2 n$

If run time based on both n, m

$f(n,m) = cn^2m$

Code Seg #1

```
for (int i = 0; i < n; i++)
  for (int j = 0; j < i; j++)
    sum++;
```

$$\sum_{i=0}^{n-1} i = \frac{(n-1)n}{2} = O(n^2)$$

$$= \sum_{i=1}^{n-1} i$$

$0 + 1 + 2 + 3 + \cdots + (n-1)$

$$\sum_{i=0}^{n-1} i = 0 + 1 + 2 + \cdots (n-1)$$

$\sum_{i=a\,(int)}^{b\,(int)} f(i)$ short hand notation for $\boxed{a \le b}$

$f(a) + f(a+1) + f(a+2) + \cdots f(b)$

$\boxed{\text{if } b < a, \text{ sum is } 0}$

```
double sum = 0;
for (int i = a; i <= b; i++)
    sum += f(i)
→ // value of sum here.
```

$$\sum_{i=a}^{b} c = (b-a+1)c, \; a \le b$$

$$\underbrace{c + c + c + \cdots}_{b-a+1 \text{ times}}$$

Gauss story

$S = 1 + 2 + 3 + \cdots + n = \sum_{i=1}^{n} i$

$S = n + (n-1) + (n-2) + \cdots + 1 = \sum_{i=1}^{n} (n+1-i)$

$2S = \underline{(n+1)} + (n+1) + (n+1) \quad \underline{+(n+1)}$

$\frac{2S}{2} = \frac{n(n+1)}{2}$

$\boxed{S = \frac{n(n+1)}{2}}$

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

$$2S = \sum_{i=1}^{n} i + \sum_{i=1}^{n} (n+1-i)$$

$$2S = \sum_{i=1}^{n} (i + n + 1 - i)$$

$$2S = \sum_{i=1}^{n} \underline{(n+1)}$$

$$2S = (n-1+1)(n+1)$$

$$2S = n(n+1)$$

$$\boxed{\sum_{i=a}^{b} f(i) + \sum_{i=a}^{b} g(i) = \sum_{i=a}^{b} (f(i) + g(i))}$$

$$\sum_{i=a}^{b} cf(i) = c\sum_{i=a}^{b} f(i), \quad \sum_{i=a}^{b} f(i) = \sum_{i=1}^{b} f(i) - \sum_{i=1}^{a-1} f(i)$$

$a > 1$

$$f(a)+f(a+1)+f(a+2)+\cdots f(b)$$

$$\boxed{\cancel{f(1)+f(2)+\cdots f(a-1)}}+f(a)+f(a+1)+\cdots f(b)$$

$$\sum_{i=n+1}^{n^2}(4i+3) = \sum_{i=n+1}^{n^2}(4i) + \sum_{i=n+1}^{n^2} 3$$

$$= 4\sum_{i=n+1}^{n^2} i + 3\left(n^2-(n+1)+1\right)$$

$$= 4\left(\sum_{i=1}^{n^2} i - \sum_{i=1}^{n} i\right) + 3n(n-1)$$

$$= 4\left(\frac{n^2(n^2+1)}{2} - \frac{n(n+1)}{2}\right) + 3n(n-1)$$

$$= 2(n^4 + \cancel{n^2} - \cancel{n^2} - n) + 3n^2 - 3n$$

$$= 2n^4 - 2n + 3n^2 - 3n$$

$$= \boxed{2n^4 + 3n^2 - 5n}$$

$n=3$

$$\sum_{i=4}^{9}(4i+3)$$

$19+23+27$

$39+35+31+$

$\cancel{27 \times 5 = 135}$

$135 + 39 = 174$ ✓

$$= 2(3)^4 + 3(3^2) - 15$$

$162 + 27 - 15$

$162 + 12$ $\quad \underline{174}$ ✓

Code Seg #2

```
int lo = 0
int hi = n
while (lo < hi) {
    int mid = (lo+hi)/2;
    if (ans < a[mid])
        hi = mid - 1
    else if (ans > a[mid])
        lo = mid + 1
    else
        break;
}
```

$O(\lg n)$

look at

0. $hi - lo \quad n$
1. $hi - lo \leq \frac{n}{2}$
2. $hi - lo \leq \frac{n}{4}$

k. $hi - lo \leq \frac{n}{2^k}$

times loop runs is k where

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

$$n = 2^k$$

$$k = \log_2 n$$

$$a^b = c \longleftrightarrow \log_a c = b$$

```
lo = 0, hi = n
while (lo < hi) {
    int mid = (lo+hi)/2;
    for (int i = 0; i < n; i++)
        { sum += a[i]
        }
    if (sum <
        hi = mid - 1
    else
        lo = mid + 1
}
```

loop runs $O(\lg n)$ time

$O(n)$

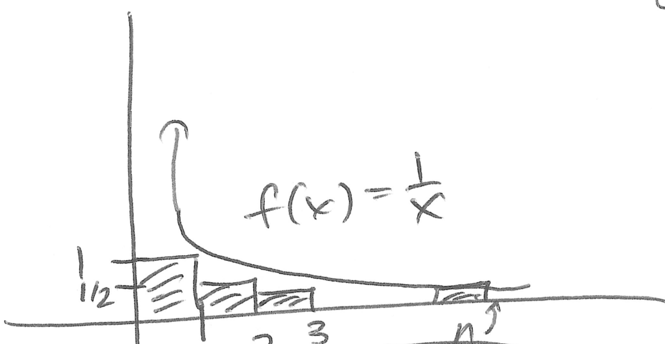
$\rightarrow O(n \lg n)$

Code Seg # 3

```
for (int i = 1; i <= n; i++)
    for (int j = i; j <= n; j += i)
        Sum++;
```

i	# time
1	n
2	$\frac{n}{2}$
3	$\frac{n}{3}$
4	$\frac{n}{4}$
$\vdots$	
n	$\frac{n}{n}$

$$\text{runtime} \le \sum_{i=1}^{n} \frac{n}{i} = n \boxed{\sum_{i=1}^{n} \frac{1}{i}}$$



$$H_n = \sum_{i=1}^{n} \frac{1}{i}$$

$$\sum_{i=1}^{n} \frac{1}{i} \le 1 + \int_1^n \frac{1}{x}\,dx$$

$$= 1 + \ln x \Big|_1^n$$

$$= 1 + \ln n$$

$$= O(\lg n)$$

Code Segment #4

```
while (n > 0) {
    for (int i = 0; i < n; i++)
        x++;
    n = n/2;
}
```

$n, \frac{n}{2}, \frac{n}{4}, \ldots, 1$

$$\leq \sum_{i=0}^{\infty} \frac{n}{2^i}$$

$$= n\sum_{i=0}^{\infty} \left(\frac{1}{2}\right)^i = n \cdot \frac{1}{\left(1-\frac{1}{2}\right)}$$

$$= 2n = O(n)$$

$$S = 1 + \cancel{x} + \cancel{x^2} + \cancel{\cdots} \qquad |x|<1$$

$$-xS = \quad \cancel{x} + \cancel{x^2} + \cancel{x^3} \cancel{\cdots}$$

$$S - xS = 1$$

$$S(1-x) = 1$$

$$S = \frac{1}{1-x}$$

$$S = \sum_{i=0}^{\infty} x^i = \frac{1}{1-x}, \quad |x|<1$$

$$S = \sum_{i=0}^{n-1} a_1 \cdot r^i$$

$$a_1 + \cancel{a_1 r} + \cancel{a_1 r^2} + \cdots \cancel{a_1 r^{n-1}}$$

$$- \quad \cancel{a_1 r} + \cancel{a_1 r^2} + \cdots \cancel{a_1 r^{n-1}} + a_1 r^n$$

$$a_1 \qquad\qquad -a_1 r^n$$

$$rS = \sum_{i=0}^{n-1} a_1 \cdot r^{i+1} = \sum_{i=1}^{n} a_1 \cdot r^i$$

$$S(1-r) = a_1 - a_1 r^n$$

$$S = \frac{a_1(1-r^n)}{1-r}$$

$$S - rS = \sum_{i=0}^{n-1} a_1 r^i - \sum_{i=1}^{n} a_1 r^i = \sum_{i=0}^{0} a_1 r^i + \cancel{\sum_{i=1}^{n-1} a_1 r^i} - \cancel{\sum_{i=1}^{n-1} a_1 r^i} - \sum_{i=n}^{n} a_1 r^n$$

$$S(1-r) = a_1 - a_1 r^n$$

$$S = \frac{a_1(1-r^n)}{1-r}$$

$$\sum_{i=0}^{n-1} a_1 r^i = \frac{a_1(1-r^n)}{1-r}, \quad r \neq 1$$

$$= \frac{a_1(r^n-1)}{r-1}$$

$$S = \sum_{i=1}^{n} i \cdot 2^i = 1\cdot 2^1 + 2\cdot 2^2 + 3\cdot 2^3 + \dots \quad n\cdot 2^n$$

$$-2S = \sum_{i=1}^{n} i \cdot 2^{i+1} = - \quad 1\cdot 2^2 + 2\cdot 2^3 \quad (n-1)2^n + n\cdot 2^{n+1}$$

$$= \boxed{\sum_{i=2}^{n+1} (i-1) 2^i}$$

$$= 1\cdot 2^1 + 1\cdot 2^2 + 1\cdot 2^3 + \dots + 2^n - n2^{n+1}$$

$$S - 2S = \sum_{i=1}^{n} 2^i - n(2^{n+1})$$

$$-S = \sum_{i=0}^{n} 2^i - 1 - n(2^{n+1})$$

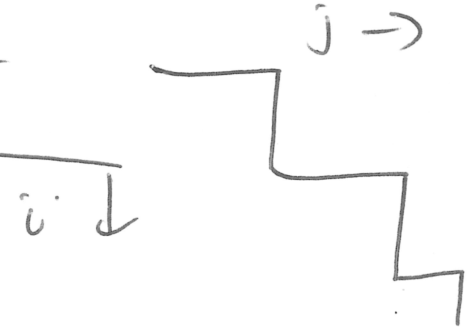
$$-S = \frac{2^{n+1}-1}{2-1} - 1 - n2^{n+1}$$

$$-S = 2^{n+1} - 1 - 1 - n2^{n+1}$$

$$S = n\cdot 2^{n+1} + 2^{n+1} - 2$$

$$\boxed{S = 2^{n+1}(n+1) - 2}$$

~~Cose~~ Code Seg #5




```
int i=0, j=0;
while (i<n) {
    while (j<n && a[j]==1)
        j++;
    i++;
}
```

$O(n)$

Value of j only changes in here so over the entire course of code j++ runs no more than n times

If you insert j=0 $O(n^2)$