Cheat Sheet

I - Solving Recurrence

① **Master theorem**: $T(n) = aT\left(\frac{n}{b}\right) + O(n^d)$

$$T(n) = \begin{cases} O(n^d) & \text{if } d > \log_b a \\ O(n^d \log n) & \text{if } d = \log_b a \\ O(n^{\log_b a}) & \text{if } d < \log_b a \end{cases}$$

② **Substitution**: ex: $T(n) = 2T(n/2) + n^2$

$T(n) \leq cn^2$

$T(n) \leq 2c\left(\frac{n}{2}\right)^2 + n^2 \leq cn^2$

$\frac{cn^2}{2} + n^2 \leq cn^2$

$\frac{c}{2} + 1 \leq c$

$\frac{c}{2} \geq 1 \quad c \geq 2$

So $T(n) = O(n^2)$

③ **Recurrence Tree Method / unrolling method:**

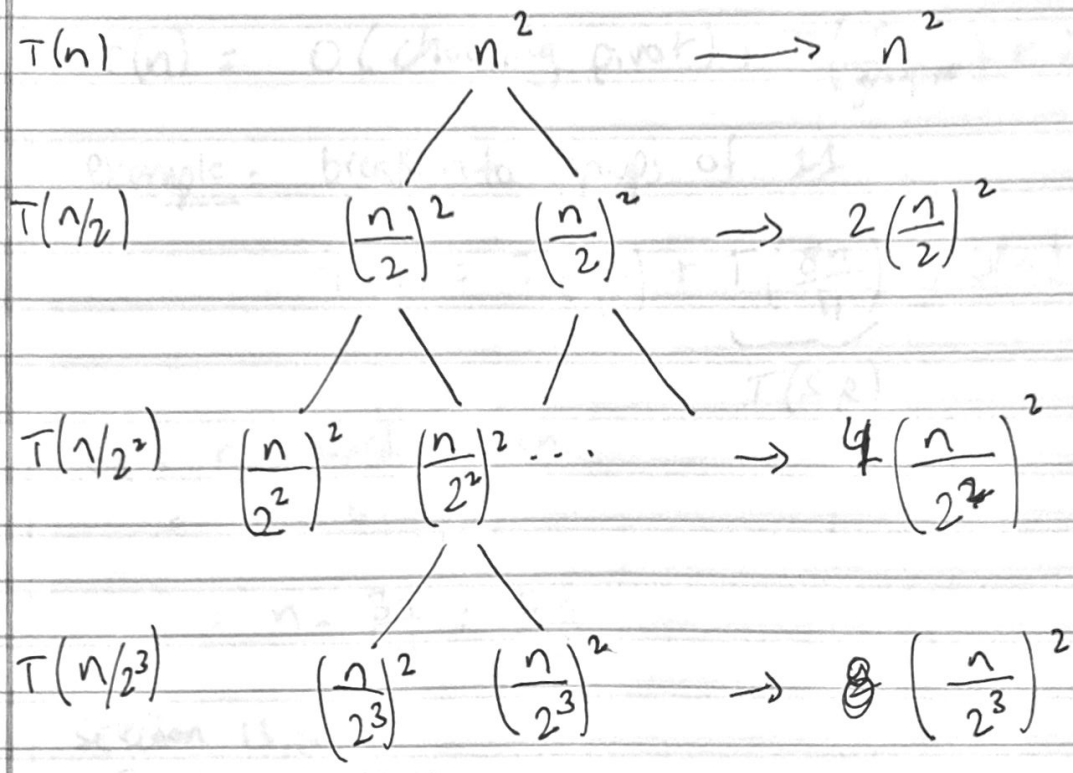
$T(n) = 2T(n/2) + n^2$

$T(n/2) = 2[2T(n/4) + (n/2)^2] + n^2$

$\vdots$

$T(n) = 2T(n/2) + n^2$ Tree

Row Sum



general equation: $2^i \left(\frac{n}{2^i}\right)^2$

Stop at base case

when $i = \log_2 n$ $\sum_{i=0}^{i=\log_2 n} 2^i \left(\frac{n}{2^i}\right)^2$

$$n^2 \sum_{i=0}^{i=\log_2 n} \frac{2^i}{2^i \cdot 2^i} = \frac{1}{2^i}$$

$$n^2 \cdot \frac{1}{2^{\log_2 n}} = n^2$$

$$\boxed{T(n) = O(n^2)}$$

Selection Algorithm

$$T(n) = O(\text{choosing pivot}) + T\left(\frac{n}{\text{group\#}}\right) + T(S_L, S_R)$$

example: break into groups of 11

$$T(n) = T\left(\frac{n}{11}\right) + \underbrace{T\left(\frac{8n}{11}\right)}_{T(S_R)} + O(n)$$

S_R = n - shaded region

$$= n - 6\left(\frac{n}{11}/2\right)$$

$$= n - \frac{3n}{11} = \frac{8n}{11}$$

selection (S, K)

① choose pivot

1-a) Break up S into groups of 5

b) find $\bar{S}_i$ = median of each group $\bar{S}_i$ after sorting

c) V = selection $(\{\bar{S}_1, \bar{S}_2, \bar{S}_3 \dots \bar{S}_{n/5}\}, n/5/2)$

↳ median of medians

② given V from step 1 divide groups into S_L, S_V, S_R

③ Recurse on either S_L, S_R or return V

Integer Multiplication

Pseudo-code: mult (x, y) :

1. Break up x and y in half
2. A = mult(x_L, y_L)
3. D = mult(x_R, y_R)
4. E = Mult(x_L+x_R, y_L+y_R) - A - D
5. Return shift (A, n) + shift (E, n/2) + D

Ex: $A = x_L y_L$ $D = x_R y_R$ $F = (x_L+x_R)(y_L+y_R)$

$E = F - A - D$ Shifted A = $A \times 10^n$

Shifted ~~E~~ E = $E \times 10^{n/2}$

Final Answer: Shifted E + Shifted A + D

$T(n) = 3T(n/2) + O(n)$

$\boxed{T(n) = O(n^{\log_2 3})}$

95 x 12

9×1 5×2 $(9+5)(1+2) = (14) \times 3$

42

① 1×0 ② 4×3 ③ $(1+4)(0+3) = 5 \times 3$

level 2: $A = 1 \times 0 = 0$ Shifted A = $0 \times 10^2 = 0$

$D = 4 \times 3 = 12$ Shifted E = $15 - 0 - 12 \times 10^1$

$F = 5 \times 3 = 15$ $= 3 \times 10 = 30$

Answer = $30 + 12 + 0 = 42$

level 1: $A = 9 \times 1 = 9$ Shifted A = $9 \times 10^2 = 900$

$D = 5 \times 2 = 10$ Shifted E = $42 - 10 - 9$

$F = 42$ $= 23 \times 10$

Answer = $230 + 900 \times 10 =$ ~~345~~ 1140

String Search Z-algorithm

Brute force method → $|P| \cdot |T|$ $\quad |P| = n/2$, $|T| = n$

$T(n) = O(n^2)$

Z-algorithm → $T(n) = |P| \$ |T| = |P| + |T|$

$T(n) = O(n)$

1 2 3 4 5 6 7 8 9 10 11

Ex: S = a a b x a a b x a a z

i	z_i	l_i	r_i
2	1	2	2
3	0	0	0
4	0	0	0
5	3	5	7
6	1	5	7
7	0	5	7
8	0	0	0
9	2	9	10
10	1	9	10
11	0	0	0

Case 1: $z_{k'} < \beta$

- $z_k = z_{k'}$

Case 2: $z_{k'} > \beta$

- $z_k = |\beta|$

Case 3: $z_{k'} = \beta$

$z_k = |\beta| +$ Bruteforce $(\beta+1, r+1)$