第 2 讲课后作业

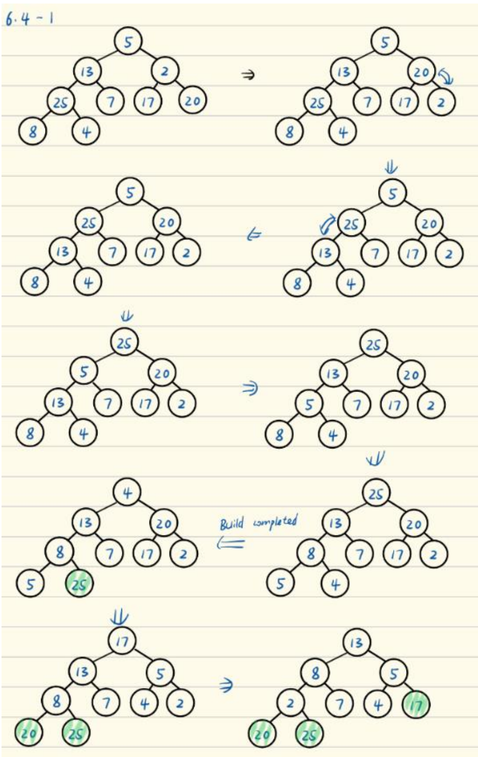
Exercise 10.1-1

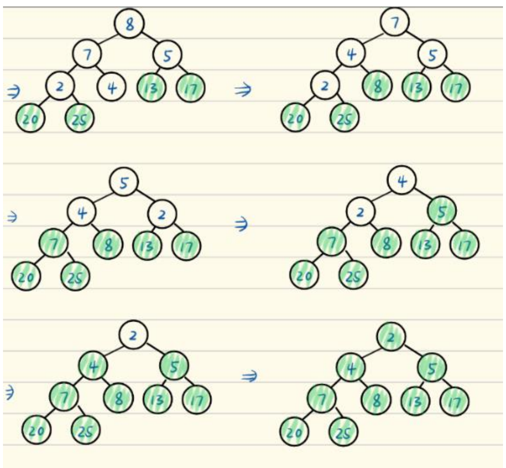
4		
4	1	
4	1	3
4	1	
4	1	8
4	1	

Exercise 10.1-3

4			
4	1		
4	1	3	
	1	3	
	1	3	8
		3	8

第 6 讲课后作业



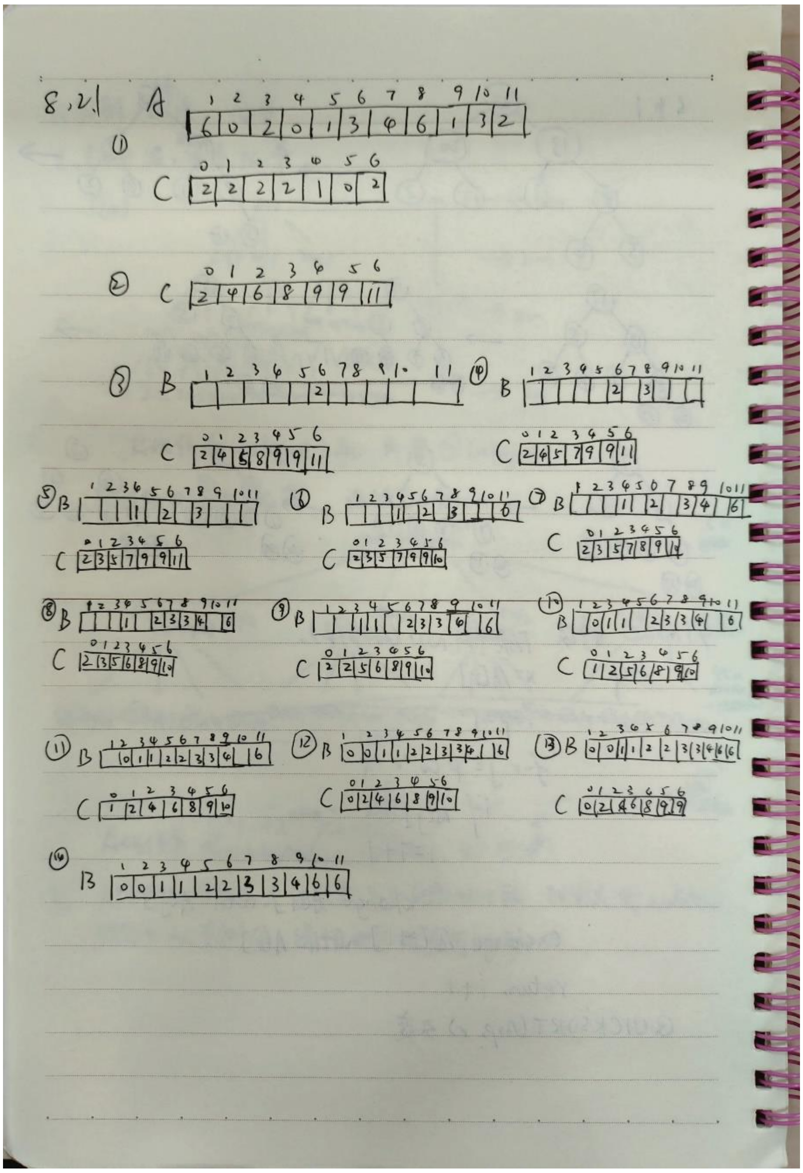


Exercise 7.1-4

To modify QUICKSORT to run in non-increasing order we need only modify line 4 of PARTITION, changing $\leq$ to $\geq$.

Exercise 8.2-1

We have that $C = \langle 2, 4, 6, 8, 9, 9, 11 \rangle$. Then, after successive iterations of the loop on lines 10-12, we have $B = \langle , , , , , 2, , , , , \rangle$,$B = \langle , , , , , 2, , 3, , , \rangle$,$B = \langle , , , 1, , 2, , 3, , , \rangle$, and at the end, $B = \langle 0, 0, 1, 1, 2, 2, 3, 3, 4, 6, 6 \rangle$



第 7 讲课后作业

Exercise 3.1-1

$0 \le \frac{1}{2}(f(n)+g(n)) \le \max\{f(n), g(n)\} \le f(n)+g(n)$

$\max\{f(n), g(n)\} = \Theta(f(n)+g(n))$.

Exercise 3.1-2

Let $c = 2^b$ and $n_0 \ge 2a$. Then for all $n \ge n_0$ we have $(n+a)^b \le (2n)^b = cn^b$ so $(n+a)^b = O(n^b)$. Now let $n_0 \ge \frac{-a}{1-1/2^{1/b}}$ and $c = \frac{1}{2}$. Then $n \ge n_0 \ge \frac{-a}{1-1/2^{1/b}}$ if and only if $n - \frac{n}{2^{1/b}} \ge -a$ if and only if $n + a \ge (1/2)^{a/b} n$ if and only if $(n+a)^b \ge cn^b$. Therefore $(n+a)^b = \Omega(n^b)$. By Theorem 3.1, $(n+a)^b = \Theta(n^b)$.

3.1-4

(1) if $O(2^n) = 2^{n+1}$，则 $\exists\, c, n_0 > 0$, s.t. $\forall n > n_0$ 有

$0 \le 2^{n+1} \le c \cdot 2^n$ 显然取 $c \ge 2$ 即可 ∴成立

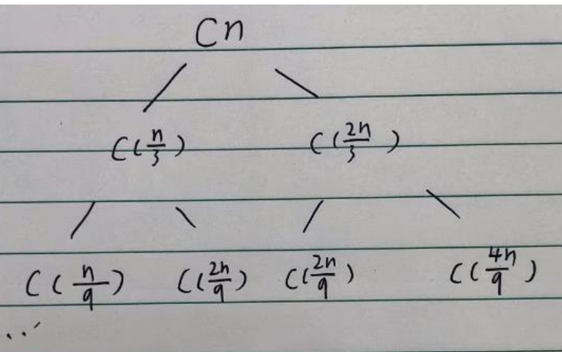
(2) if $O(2^n) = 2^{2n}$，则 $\exists\, c, n_0 > 0$, s.t. $\forall n > n_0$ 有

$0 \le 2^{2n} \le c \cdot 2^{n+1}$ 即 $0 \le 2^{n-1} \le c$

而 2^{n-1} 是单调递增的无界序列，∴不存在这样的 c 与 n_0 ∴不成立

第 8 讲作业

4.4-6:



则最小深度为 $\log_3 n$，每层代价为 cn

∴ $T(n) \ge cn\log_3 n$，即 $T(n) = \Omega(n\log n)$

4.5-3，由递推条件，$a=1$，$b=2$，$f(n) = \Theta(1) = \Theta(n^{\log_2 1})$

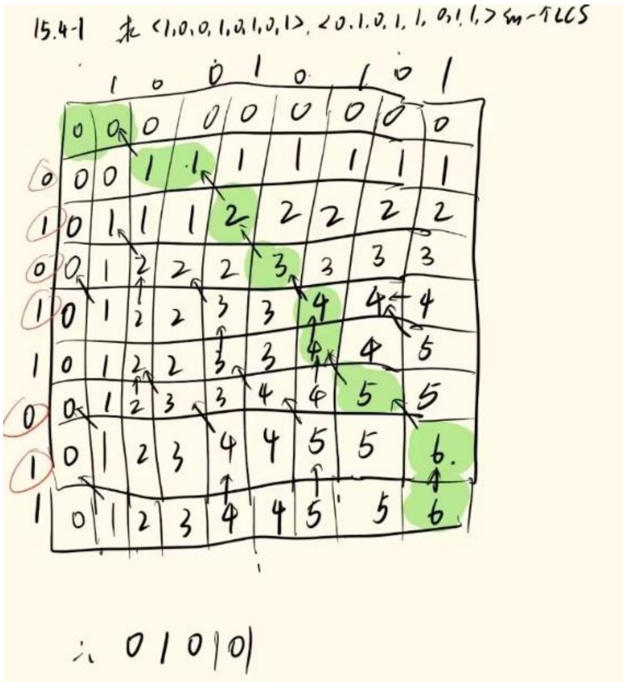
∴ $T(n) = \Theta(n^{\log_2 1} \cdot \lg n) = \Theta(\lg n)$

第 9 讲作业

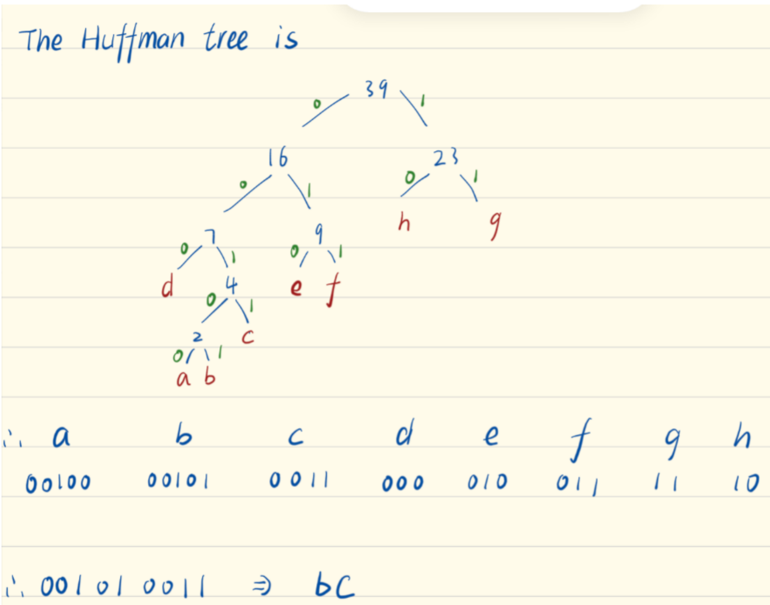
Exercise 15.2-1

An optimal parenthesization of that sequence would be $(A_1A_2)((A_3A_4)(A_5A_6))$ which will require $5*50*6+3*12*5+5*10*3+3*5*6+5*3*6 = 1500+180+150+90+90 = 2010$.

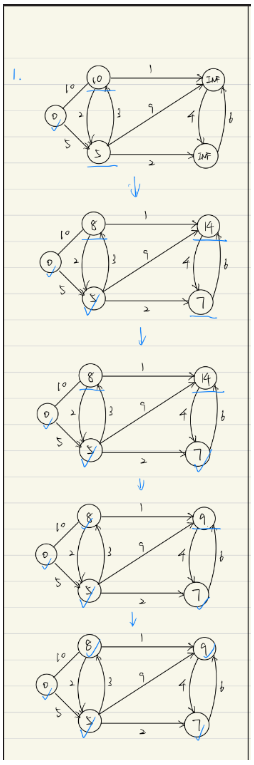
Exercise 15.4-1

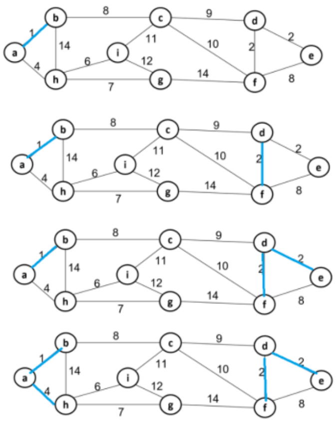


第 10 讲作业

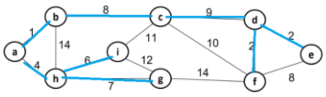


第 14 讲作业

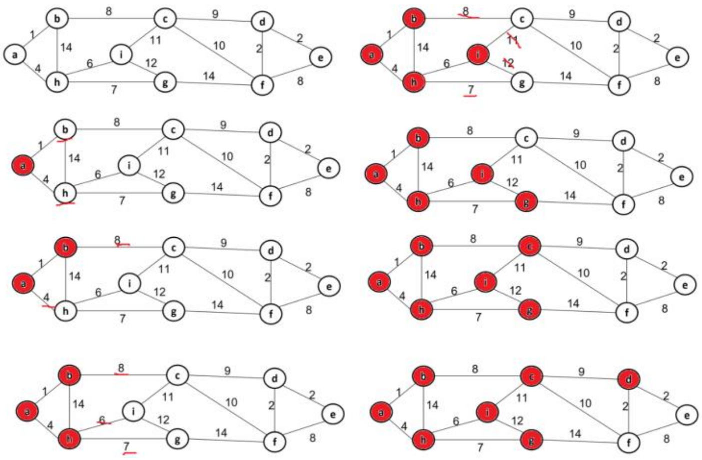


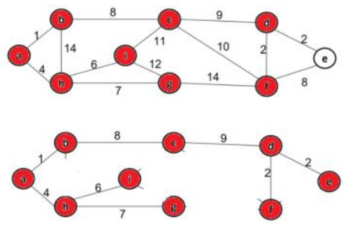


以此类推...

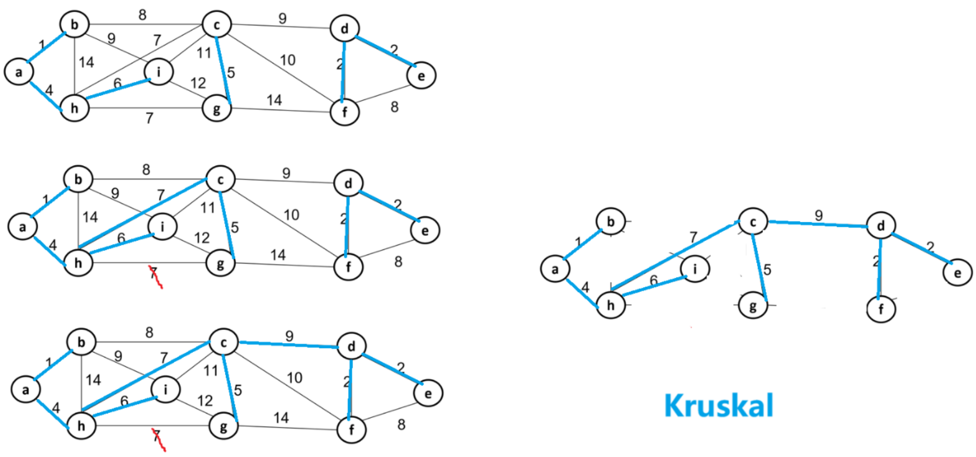


Kruskal

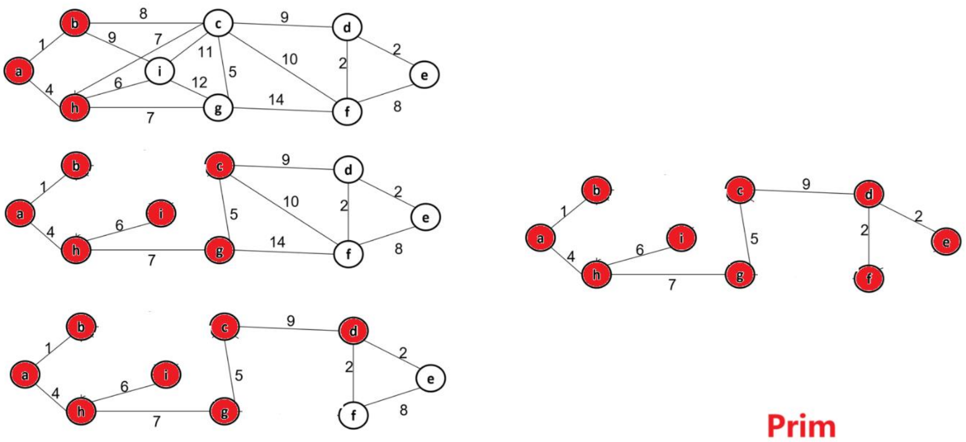




Prim



Kruskal



Prim

稀疏图：在稀疏图中，边的数量相对较少。Kruskal 算法通过排序和选择边来构造最小生成树，由于只需要遍历一遍所有的边，因此在稀疏图中通常会更快。而 Prim 算法则需要遍历所有的点和边，可能会稍慢。因此，在稀疏图中，Kruskal 算法可能更加适用。

稠密图：在稠密图中，边的数量很多。Kruskal 算法因为需要遍历所有的边而变得较慢。而 Prim 算法则可以通过选择与当前点相连且权重最小的边来避免遍历所有的边，因此在稠密图中可能会更快。因此，在稠密图中，Prim 算法可能更加适用。