Homework 1

吴昊 2016K8009929015 （本科生）

邮件：wuhao164@mails.ucas.ac.cn

1 Quick Sort time complexity analysis

So far we know that

$$E(T(n)) = \frac{1}{n} \sum_{1 \le k \le n} [E(T(k-1)) + E(T(n-k)) + n - 1]$$

just for simplicity,we can abuse the notation T(n) to represent E(T(n)),so the recursive equation can be transformed into formula (1)

$$nT(n) = n(n-1) + 2 \sum_{1 \le i \le n-1} T(i)$$

replace n by $n-1$ in the equation above,we have fomula (2)

$$(n-1)T(n-1) = (n-1)(n-2) + 2 \sum_{1 \le i \le n-2} T(i)$$

subtract (2) from (1),we get

$$nT(n) = (n+1)T(n-1) + 2n - 2$$

both side multipled by factor $\frac{1}{n(n+1)}$

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

by using change of variable technique,let $G(n) = \frac{T(n)}{n+1}$

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

notice that $G(1) = \frac{T(1)}{2} = 0$

$$G(n) = G(n) - G(1) = \sum_{2 \leq i \leq n} [G(i) - G(i-1)]$$

$$= \sum_{2 \leq i \leq n} \frac{2}{i+1} - \sum_{2 \leq i \leq n} \frac{2}{i(i+1)}$$

$$= 2 \sum_{3 \leq i \leq n} \frac{1}{i} - 1 + \frac{2}{n+1}$$

for the reason that

$$\int_3^{n+1} \frac{1}{x} dx \leq \sum_{3 \leq i \leq n} \leq \int_2^n \frac{1}{x} dx$$

so we have

$$\frac{T(n)}{n+1} = O(\log n)$$

which implies

$$T(n) = O(n \log n)$$

2 modified Min-Cut

Exercise 1.2: Suppose that at each step of our min-cut algorithm, instead of choosing a random edge for contraction we choose two vertices at random and coalesce them into a single vertex. Show that there are inputs on which the probability that this modified algorithm finds a min-cut is exponentially small

consider a graph with n vertices,among which k vertices form a complete graph, and the other n-k vertices form another complete graph, there is only one edge between the two complete graph. Obviously,the min cut is 1. Only if whenever we coalesce two vertices, these two vertices come from the same complete subgraph,can we get the right answer.

The order of vertices to coalesce forms a sequence,for a certain sequence, the successful probability is

$$Pr(correct) = \frac{C_k^2 C_{k-1}^2 C_{k-2}^2 \cdots C_2^2 C_{n-k}^2 C_{n-k-1}^2 \cdots C_2^2}{C_n^2 C_{n-1}^2 C_{n-2}^2 \cdots C_3^2}$$

$$= \frac{k(n-k) \prod_{1 \le i \le k-1} i^2 \prod_{1 \le i \le n-k-1} i^2}{n \prod_{1 \le i \le n-1} i^2}$$

the number of sequences that can produce right answers is

$\frac{(n-2)!}{(n-k-1)!(k-1)!}$

when $k = \frac{n}{2}$, the successful probability of the algorithm should be

$$Pr = \frac{\prod_{1 \le i \le \frac{n}{2}-1} i^2}{2 \prod_{\frac{n}{2} \le i \le n-1} i^2} \frac{(n-2)!}{(\frac{n}{2}-1)!(\frac{n}{2}-1)!} = \frac{\prod_{1 \le i \le \frac{n}{2}-1} i}{2(n-1)^2 \prod_{\frac{n}{2} \le i \le n-2} i} \le \frac{1}{2^{\frac{n}{2}-1}}$$

which is exponentially small.