Programming Assignment 1 Report

Date

1. pseudo code

```
Select(N, K, G):
    if N.length <= G
        sort(N)   // constant time becasuse #item is very small
        return N[K-1]

    // last group may not contain enough items
    Let AllMedians be a new array
    Divide N into G-items groups
    Collect median of each group into AllMedians (By calling Select recursively)

    m = Select(AllMedians, AllMedians.length/2, G)

    X = {items in N that are smaller than m}
    Y = {items in N that are larger than m}
    duplicate = #num in N that equals to m

    if K >= X.length + 1 && K <= X.length + duplicate
        return m
    else if K <= X.length
        return Select(X, K, G)
    else
        return Select(Y, K - X.length - duplicate, G
```

- 如果 array 長度 $\leq G$，那麼我們就直接 sort 然後回傳 $N[K-1]$，因為 G 很小，所以不會花太多時間在 sorting
- 接著把整個 array 每 G 個數字分成一組，再遞迴呼叫 `Select` 找出 median，全部加進 `AllMedians` 之後一樣遞迴呼叫 Select 找出 median of median m
- 找到 median of median 之後，我們用一個迴圈即可把 $< m$ 的數字放在 X array 裡面，$> m$ 的數字放在 Y array 裡面，如果是 $== m$，我們就增加 duplicate 的值，來計算有多少跟 m 一樣的數字
- 以 `[1, 3, 5, 5, 5, 5, 9, 8]` 為例，假設我們要找的是第 4 大的數字，X 裡面會有 `[1, 3]`，Y 裡面會有 `[9, 8]`，duplicate 會是 4，m 是 5
    - 由此可知，如果 $K \geq X.length + 1 = 3$ 而且 $K \leq X.length + duplicate = 6$，我們就應該 return m

- 如果 $K \leq X.length$，代表我們要找的數字在 X 裡面，所以 return `Select(X, K, G)`
- 否則代表我們要找的數字在 Y 裡面，而且我們要找的是在 Y 裡面，第 $K - X.length - duplicate$ 大的數字，所以 return `Select(Y, K - X.length - duplicate, G)`

- 基本上跟一般的 Select kth smallest item 演算法一樣，只是因為會有重複的數字，所以在比較的時候要有一個區間，而不是 `K == X.length + 1` 就 return m

---

2. Running Time Comparison

- Group 3
  1. Execution time: **4012.71 ms**
  2. Time Complexity

$$
\begin{aligned}
T(n) &= T(\lceil\frac{n}{3}\rceil) + T(\frac{2n}{3} + 4) + \theta(n) \\
&\leq c\lceil\frac{n}{3}\rceil + c(\frac{2n}{3} + 4) + an \\
&\leq \frac{cn}{3} + c + \frac{2cn}{3} + 4c + an \\
&= cn + 5c + an \\
&\nleq cn
\end{aligned} \tag{1}
$$

因為 $5c + an$ 不可能 ≤ 0，所以不成立，成長速度會 $> O(n)$

- Group 5
  1. Execution time: **1510.68 ms**
  2. Time Complexity

     利用 recursion tree，可以猜測 $T(n) = O(n)$

     利用 substitution method，假設 $T(m) \leq cm \ \forall m < n$

$$
\begin{aligned}
T(n) &= T(\lceil \frac{n}{5} \rceil) + T(\frac{7n}{10} + 6) + \theta(n) \\
&\le c\lceil \frac{n}{5} \rceil + c(\frac{7n}{10} + 6) + an \\
&\le \frac{cn}{5} + c + \frac{7cn}{10} + 6c + an \\
&= \frac{9cn}{10} + 7c + an \\
&= cn + (-\frac{cn}{10} + 7c + an) \\
&\le cn \quad if \quad -\frac{cn}{10} + 7n + an \le 0
\end{aligned} \tag{2}
$$

$c \ge \frac{10an}{(n-70)}$ when $n > 70$，如果 $n \ge 140$ 我們就有 $\frac{n}{n-70} \le 2$，因此我們可以選 $c \ge 20a$

- Group 7
  1. Execution time: **1021.97 ms**
  2. Time Complexity

     利用 recursion tree，可以猜測 $T(n) = O(n)$

     利用 substitution method，假設 $T(m) \le cm \ \ \forall m < n$

$$
\begin{aligned}
T(n) &= T(\lceil \frac{n}{7} \rceil) + T(\frac{10n}{14} + 8) + \theta(n) \\
&\le c(\lceil \frac{n}{7} \rceil) + c(\frac{10n}{14} + 8) + an \\
&\le \frac{cn}{7} + c + \frac{10cn}{14} + 8c + an \\
&= \frac{6cn}{7} + 9c + an \\
&= cn + (-\frac{cn}{7} + 9c + an) \\
&\le cn \quad if \quad -\frac{cn}{7} + 9c + an \le 0
\end{aligned} \tag{3}
$$

$c \ge \frac{7an}{(n-63)}$ when $n > 63$，如果 $n \ge 126$ 我們就有 $\frac{n}{n-63} \le 2$，因此我們可以選 $c \ge 14a$

- Group 9
  1. Execution time: **973.821 ms**

2. Time Complexity

$$
\begin{aligned}
T(n) &= T(\lceil \frac{n}{9} \rceil) + T(\frac{13n}{18} + 10) + \theta(n) \\
&\le c(\lceil \frac{n}{9} \rceil) + c(\frac{13n}{18} + 10) + an \\
&\le \frac{cn}{9} + c + \frac{13cn}{18} + 10c + an \\
&= \frac{5cn}{6} + 10c + an \\
&= cn + (-\frac{cn}{6} + 10c + an) \\
&\le cn \quad if \quad -\frac{cn}{6} + 10c + an \le 0
\end{aligned} \tag{4}
$$

$c \ge \frac{6an}{(n-60)}$ when $n > 60$，如果 $n \ge 120$ 我們就有 $\frac{n}{n-60} \le 2$，因此我們可以選 $c \ge 12a$

- Randomized-Select

1. Execution time: **130.539 ms**
2. Time Complexity

- Average case

$$
\begin{aligned}
E[T(n)] &= O(n) + \frac{1}{n} \sum_{1 \le k \le n} E[T(max(k-1, n-k))] \\
&= O(n) + \frac{2}{n} \sum_{\lfloor \frac{n}{2} \rfloor \le k \le n-1} E[T(k)]
\end{aligned} \tag{5}
$$

Assume $E[T(n)] \le cn$ hold for $n \le k$

$$
\begin{aligned}
E[T(n)] &\leq \frac{2}{n}\sum_{k=\lfloor\frac{n}{2}\rfloor}^{n-1} ck + an \\
&= \frac{2c}{n}(\sum_{k=1}^{n-1} k - \sum_{k=1}^{\lfloor\frac{n}{2}\rfloor-1} k) + an \\
&= \frac{2c}{n}(\frac{(n-1)n}{2} - \frac{(\lfloor\frac{n}{2}\rfloor-1)\lfloor\frac{n}{2}\rfloor}{2}) + an \\
&\leq \frac{2c}{n}(\frac{(n-1)n}{2} - \frac{(\frac{n}{2}-2)(\frac{n}{2}-1)}{2}) + an \\
&= \frac{2c}{n}(\frac{n^2-n}{2} - \frac{\frac{n^2}{4}-\frac{3n}{2}+2}{2}) + an \\
&= \frac{c}{n}(\frac{3n^2}{4} + \frac{n}{2} - 2) + an \\
&= c(\frac{3n}{4} + \frac{1}{2} - \frac{2}{n}) + an \\
&\leq \frac{3cn}{4} + \frac{c}{2} + an \\
&= cn - (\frac{cn}{4} - \frac{c}{2} - an)
\end{aligned} \tag{6}
$$

因為要證明 $\leq cn$ 所以我們需要 $\frac{cn}{4} - \frac{c}{2} - an \geq 0$

$$
=> \ \frac{cn}{4} - an \geq \frac{c}{2}
$$

$$
=> \ n(\frac{c}{4} - a) \geq \frac{c}{2} \ \ => n \geq \frac{\frac{c}{2}}{\frac{c}{4} - a} = \frac{2c}{c-4a}
$$

所以，如果我們假設 $T(n) = O(1)$ for $n < \frac{2c}{c-4a}$ 那麼 $E[T(n)] = O(n)$