EECS 340: Assignment 3

Shaochen (Henry) ZHONG, `sxz517`
Yuhui ZHANG, `yxz2052`

Due and submitted on 02/20/2020
EECS 340, Dr. Koyutürk

Problem 1

(a) $T(n) = bT(n/a) + \Theta(n)$

For this recurrance, we have a comparsion between $n^{\log_a b}$ and n. Since it is given that $1 < a < b$, there must be $\log_a b > 1$. Therefore we may say that there must be a $n^{\epsilon} = \frac{n^{\log_a b}}{n}$ where $0 < \epsilon = \log_a b - 1$.

Now we have $f(n) = n = O(n^{(\log_a b) - \epsilon})$, where $0 < \epsilon = \log_a b - 1$, we can apply *case 1* of the master theorem and conclude that the solution is $T(n) = \Theta(n)$.

(b) $T(n) = a^2 T(n/a) + \Theta(n^2)$

For this recurrance, we have a comparsion between $n^{\log_a a^2}$ and n^2, thus n^2 and n^2. As now we have $f(n) = \Theta(n^2)$, we can apply *case 2* of the master theorem and conclude that the solution is $T(n) = \Theta(n \cdot \log n)$

(c) $T(n) = T(\lambda n) + n^{\lambda}$

We may rewrite it as $T(n) = T\left(\frac{n}{\frac{1}{\lambda}}\right) + n^{\lambda}$. Thus, we have a comparsion between $n^{\log_{\frac{1}{\lambda}} 1}$ and n^{λ}, which is equivalent as n^0 and n^{λ}, then we have $f(n) = \Omega(n^{\log_b a + \epsilon})$ for $\epsilon = \lambda$. We may also show $af(\frac{n}{b}) \leq cf(n)$ for $1 \cdot f\left(\frac{n}{\frac{1}{\lambda}}\right) = f(\lambda n)$. Combined, together, we can apply *case 3* of the master theorem and conclude that the solution is $T(n) = \Theta(n^{\lambda})$.

(d) $T(n) = aT(\frac{n}{a}) + \Theta(n^{\lambda}(\log n)^b)$

For this recurrance, we have a comparsion between $n^{\log_a a}$ and $n^{\lambda}(\log n)^b)$, which is equivalent to comparing n and $n^{\lambda}(\log n)^b)$. We may prove that n is polynomially larger than $n^{\lambda}(\log n)^b)$ by analyzing:

$$
\begin{aligned}
\text{W.T.S.} \quad \lim_{n\to\infty} \frac{n^{\epsilon} \cdot n^{\lambda} (\log n)^b}{n} &= 0 && (1) \\
\lim_{n\to\infty} \frac{n^{\lambda} (\log n)^b}{n^{1-\epsilon-\lambda}} &= 0 \\
\Longrightarrow 1 - \epsilon - \lambda > 0 \Rightarrow \epsilon < 1 - \lambda & && (2)
\end{aligned}
$$

Thus, we have $f(n) = O(n^{\log_b a - \epsilon})$ for $\epsilon < 1 - \lambda$. Then, we can apply *case 1* of the master theorem and conclude that the solution is $T(n) = \Theta(n)$.

Problem 2

Problem 3

Problem 4

Algorithm 1 QuickMiss(C, D, start, end)

procedure QuickMiss(C, D, start, end)
    if $p < r$ **then**
        $left, q, right \leftarrow$ Partition$(C, D, start, end)$
        if $left < right$ **then**
            Partition$(C, q+1, r)$
        else if $right > left$ **then**
            Partition$(C, p, q-1)$
        else
            return $D[q]$

Algorithm 2 Partition(C, D, p, r)

procedure Partition(C, D, p, r)
    $r \leftarrow r + 1$
    $C[r] \leftarrow D[\frac{p+r}{2}]$
    $q \leftarrow p$
    for $i \leftarrow p$ **to** $r - 1$ **do**
        if Compare-Strings$(C[i], C[r])$ **then**
            Swap$(C[i], C[q])$
            $q \leftarrow q + 1$
    Swap$(C[q], C[r])$
    return q