Bingjun Guo*, 3210115445

the student making the submission

Q1.A solution

$$f(i) = d(s_{\leq i}) = f(j) = d(s_{\leq j})$$

Meanwhile, since $d(xy) = d(x) + d(y)$, we have:

$$d(s_{\leq j}) = d(s_{\leq i}s_{i+1}s_{i+2}\dots s_j) = d(s_{\leq i}) + d(s_{i+1}s_{i+2}\dots s_j)$$

Thus, as $d(s_{\leq j}) = d(s_{\leq i})$, we have:

$$d(s_{\leq i}) + d(s_{i+1}s_{i+2}\dots s_j) = d(s_{\leq i})$$
$$\Rightarrow d(s_{i+1}s_{i+2}\dots s_j) = 0$$

which means that $s_{i+1}s_{i+2}\dots s_j$ is a weakly balanced string.

Q1.B solution

For empty string $s = \epsilon$, that is, in case (i), since $d(s) = d(\epsilon) = 0$, s is indeed a balanced string.

Then we will prove that for any non-empty balanced string s such that $d(s) = 0$ and $\#_a(p) \geq \#_b(p)$ for any prefix p of s, s will satisfy one of case (ii) and (iii).

(credits to: CS 280 materials, UChicago)

For arbitrary non-empty string s, let $s = s_1s_2$, in which s_1 is the shortest prefix string of s such that $d(s_1) = 0$, which exists since s is non-empty.

Then, s_1 must start with a, or $d(s_{\leq 1}) = -1 < 0$, which conflicts the fact that s is balanced. Meanwhile, s_1 must end with b, or $d(s_{\leq |s_1|-1}) = -1$. Therefore, we can write s_1 as as'_1b, and thus s as as'_1bs_2 (both s'_1 and s_2 can be empty).

Claim: s'_1 is balanced.

Proof: Since for any prefix p of s we have $\#_a(p) \geq \#_b(p)$, for any prefix p_1 of s_1 we should also have $\#_a(p_1) \geq \#_b(p_1)$. Therefore, s_1 is balanced. Meanwhile, for any prefix p'_1 of as'_1 we should have $\#_a(p_1) \geq \#_b(p_1) + 1$, so for any prefix p' of s'_1 we should have $\#_a(p') \geq \#_b(p')$. Furthermore, $d(s'_1) = d(s_1) - d(a) - d(b) = d(s_1) = 0$. Thus, s'_1 is balanced.

Claim: s_2 is balanced.

Proof: Firstly, $d(s_2) = d(s) - d(s_1) = 0$, that is, s_2 is weakly balanced. Secondly, since $\#_a(s_1) = \#_b(s_1)$ and s is balanced, for arbitrary prefix p_2 of s_2, we should have $\#_a(s_2) \geq \#_b(s_2)$.

With the two claims above proved, balanced string s will fall into case (iii) if $s_2 = \epsilon$ and case (ii) if $s_2 \neq \epsilon$. In conclusion, all balanced string falls into one of the three cases.

Q1.C solution

Denote $d(s_{\leq i})$ as $f(i)$ and set $\{f(0), f(1), \dots f(n)\}$ as F_n. $d_{\max}(s) = \max(F_n)$. Since s is balanced, we have $0 \leq f(i) \leq d_{\max}(s)$ for arbitrary $0 \leq i \leq n$ and $f(n) = f(0) = 0$. In addition, if $f(j) = f(i)$, according to **Q1.A**, string $s_{i+1}s_{i+2}\dots s_j$ would be weakly balanced.

Assume $d_{\max}(s) < \sqrt{n}$. *Claim:* $\max m \geq \sqrt{n} - 1$.

Proof: As $0 \leq f(i) \leq d_{\max}(s)$ and $f(i)$ is an integer, there are $d_{\max}(s) + 1$ possible values for $f(i)$. Build the set $K(z) = \{k \mid f(k) = z, k \in [1, n] \cap \mathbb{Z}\}$, $z \in [0, d_{\max}(s)] \cap \mathbb{Z}$, and set $K' = \{K(z) \mid z \in [0, d_{\max}(s)] \cap \mathbb{Z}\}$. Then we have:

$$|K'| = d_{\max}(s) + 1$$

$$\sum_{K(z) \in K'} |K(z)| = n$$

and:

$$\begin{cases} \max m = \max_{K(z) \in K'} |K(z)| + 1 & \text{if } \operatorname{argmax}_z |K(z)| \neq 0 \\ \max m = \max_{K(z) \in K'} |K(z)| - 1 & \text{if } \operatorname{argmax}_z |K(z)| = 0 \end{cases}$$

Thus, proving $\max m \geq \sqrt{n} - 1$ for arbitrary F_n is equavalent to prove (with $K(z)$ shortened today K):

$$\max_{K \in K'} \min_{F_n} |K| - 1 \geq \sqrt{n} - 1$$

with constraint:

$$\sum_{K \in K'} |K| = n$$

Denote $\max |K(z)|$ as $C_{\max}$. In order to reach the minimum $C_{\max}$, referring to the constraint, we should split the cardinity sum n as even as possible into $|K(z)|$s, that is:

$$\min_{F_n} C_{\max} = \frac{n}{d_{\max}(s)}$$

and since $C_{\max} = \max_{K \in K'} |K|$ and $d_{\max}(s) < \sqrt{n}$:

$$\max_{K \in K'} \min_{F_n} |K| \geq \frac{n}{\sqrt{n}} = \sqrt{n}$$

$$\Rightarrow \max_{K \in K'} \min_{F_n} |K| - 1 \geq \sqrt{n} - 1$$

Therefore, if $d_{\max}(s) < \sqrt{n}$, $m \geq \sqrt{n} - 1$ can happen. Otherwise, $d_{\max} \geq \sqrt{n}$ happens.

Q2 solution

We will prove by induction.

When $n = 2$, there would be only one merge of two tribes each with one people, so only one lambs would be sacrificed. $1 \leq 2 \cdot \log_2 2 = 2$.

Inductive Assumption: at most $k \log_2 k$ *lambs got sacrificed for arbitrary* $1 < k < n$.

When $n > 2$, denote the number of sacrificed lamps for i people as $L(i)$ and maximum of such number as $L_{\max}(i)$. Then, we should have:

$$L(i + j) = L(i) + L(j) + \min(i, j)$$

$$L(1) = 0,\ L(2) = 1$$

as well as:

$$L_{\max}(i + j) \geq L_{\max}(i) + L_{\max}(j) + \min(i, j)$$

Let $n = k_1 + k_2$, and values of k_1, k_2 such that the equality above is reached as $k_{1\max}, k_{2\max}$:

$$L(k_1 + k_2) = L(k_1) + L(k_2) + \min(k_1, k_2)$$

$$L_{\max}(k_1 + k_2) = L(k_{1\max}) + L(k_{2\max}) + \min(k_{1\max}, k_{2\max})$$

and according to the inductive assumption:

$$L_{\max}(k_{1\max} + k_{2\max}) = k_{1\max} \log k_{1\max} + k_{2\max} \log k_{2\max} + \min(k_{1\max}, k_{2\max})$$

Claim: $k_{1\max} = \lceil n/2 \rceil$

Proof: Denote k_1 as k for simplicity in this proof and thus k_2 as $n - k$. Then:

$$L(k) = \begin{cases} k \log k + (n-k) \log(n-k) + k & \text{if } k < \lceil n/2 \rceil \\ k \log k + (n-k) \log(n-k) - k + n & \text{if } k \geq \lceil n/2 \rceil \end{cases}$$

Meanwhile we have:

$$(k \log k + (n-k) \log(n-k))' = \log \frac{k}{n-k}$$

$$k' = 1,\ (-k)' = -1$$

Denote $\log \frac{k}{n-k}$ as $f'(k)$. As $0 < k < n$, we have $f'(k) > 0$. Thus, $L(k)$ constantly increases on $(0,\ \lceil n/2 \rceil)$. Meanhwile, for $k \geq \lceil n/2 \rceil$, we have $k \geq \lfloor n/2 \rfloor = n - k$, and thus $f'(k) - 1 < 0$, so $L(k)$ constantly decreases on $(\lceil n/2 \rceil,\ 0)$. Therefore, $L(k)$ reaches its maximum at $k = \lceil n/2 \rceil$, that is, $k_{1\max} = \lceil n/2 \rceil$.

Thus, the way to split the mega tribe with n people into two tribes which leads to most lambs sacrificed in total is to split it into tribes with $\lceil n/2 \rceil$ and $n - \lceil n/2 \rceil = \lfloor n/2 \rfloor$ people. Before there is only one person in each tribe, such splits can happen for $\log_2 n$ rounds, while at most one lamb can be sacrificed for each of the n people and each of the $\log_2 n$ splits, which indicates $L_{\max}(n) \leq n \log_2 n$.

Therefore, we conludes that at most $n \log_2 n$ lambs get sacrificed for $n > 1$.

Q3.A solution

When $n < 10$, $T(n) = 1 = O(n \log n)$.

Inductive Assumption: for arbitrary $m < n$, $T(m) = O(m \log m)$.

When $n \geq 10$:

$$\begin{aligned} T(n) &= 2n + T(\lfloor n/4 \rfloor) + T(\lfloor (3/4)n \rfloor) \\ &= 2n + O(\lfloor n/4 \rfloor \log \lfloor n/4 \rfloor) + O(\lfloor (3/4)n \rfloor \log \lfloor (3/4)n \rfloor) \\ &= O(n) + O((n/4) \log(n/4)) + O((3/4)n \log(3n/4)) \\ &= O(n) + O((n/4) \log n - (n/4) \log 4) + O((3/4)n \log n + (3/4)n \log(3/4)) \\ &= O(n) + O(n \log n - n) + O(n \log n + n) \\ &= O(n) + O(n \log n) + O(n \log n) \\ &= O(n \log n) \end{aligned}$$

Therefore, it's concluded that $T(n) = O(n \log n)$.

Q3.B solution

When $n < 24$, $T(n) = 1 = O(n)$.

Inductive Assumption: for arbitrary $m < n$, $T(m) = O(m)$.

When $n \geq 24$:

$$\begin{aligned}
T(n) &= T(\lfloor n/2 \rfloor) + T(\lfloor n/6 \rfloor) + T(\lfloor n/7 \rfloor) + n \\
&= O(\lfloor n/2 \rfloor) + O(\lfloor n/6 \rfloor) + O(\lfloor n/7 \rfloor) + n \\
&= O(n/2) + O(n/6) + O(n/7) + n \\
&= O(n) + O(n) + O(n) + O(n) \\
&= O(n)
\end{aligned}$$

Therefore, it's concluded that $T(n) = O(n)$.