Discrete Structures Midterm Project

Jerry Ngo

September 2020

1. For this exam, I choose a function named *decToBin* that takes a non-negative integer s as input and returns a binary string *ans* of s in the second base as output.

The Program:

```
1 def decToBin(s):
2     if not s: return "0"
3     ans = ""
4     power = 1
5     for i in range(int(log(s, 2)) + 1):
6         ans = ["1", "0"][((1 << i) & s) == 0] + ans
7         power *= 2
8     return ans
```

2. The domain and range of the functions is:

$$decToBin\text{: } \mathbb{Z}^{nonneg} \to \bigcup_{a \in \mathbb{Z}^+} \{0,1\}^a$$

3. I will directly prove that the *decToBin* is injective.
Suppose x, $y \in \mathbb{Z}^{nonneg}$ and $decToBin(x) = decToBin(y)$. Let $\{b_1, b_2, ..., b_m\}$ be the set of index where $decToBin(x)_{b_i} = decToBin(y)_{b_i} = 1$; $0 \leq m, b_i \leq log_2(x)$. We have $x = 2^{b_1} + 2^{b_2} + ... + 2^{b_m}$ and $y = 2^{b_1} + 2^{b_2} + ... + 2^{b_m}$. Thus, we have $x = y$. $\square$

4. I will directly prove that the *decToBin* is surjective.
Suppose $s \in \{0,1\}^a$. Let $\{b_1, b_2, ..., b_m\}$ be the set of index where $s_{b_i} = 1$; $0 \leq m, b_i \leq log_2(x)$. Consider $x = 2^{b_1} + 2^{b_2} + ... + 2^{b_m}$. Since $2, b_i \in Z^{nonneg}$, $x \in Z^{nonneg}$. $\square$

5. Analyze the best-case and worst-case complexity of your function.
(d). The best case and worst case have the same complexity because, regardless of how large or how small s is, the program has the same step formula.
(a).
(b). The number of step in all the case is 4 (from line 2, 3, 4, 8) + $log_2(s) * 2$ (line 5, 6, 7) $= 4 + \lfloor log_2(s) * 2 \rfloor$.
(c). Suppose $g(s) = log_2(s)$, we will try to prove $4 + log_2(s) * 2 \in O(g(s))$ and

$4 + log_2(s) * 2 \in \Omega(g(n))$.
To prove $4 + log_2(s) * 2 \in O(g(s))$, we find an positive constants c and s_0 such that for any $s \geq s_0$, $4 + log_2(s) * 2 \leq cg(s)$. Let $c = 3$, $s_0 = 16$, we have:

$$log_2(16) = 4 \leq log_2(s) \tag{1}$$

Add $2log_2(s)$ to both size of (1):

$$2 * log_2(s) + 4 \leq log_2(s) + 2log_2(s) = 3log_2(s) = 3g(s)$$

Therefore, $4 + log_2(s) * 2 \leq 3log_2(s)$ or $4 + log_2(s) * 2 \in O(log_2(s))$. (*)

To prove $4 + log_2(s) * 2 \in \Omega(g(s))$, we find an positive constants c and s_0 such that for any $s \geq s_0$, $4 + log_2(s) * 2 \geq cg(s)$. Let $c = \frac{2}{3}$, $s_0 = \frac{1}{8}$ and since $\frac{-4}{3}log_2(s)$ is a decreasing function, we have:

$$\frac{-4}{3}log_2(\frac{1}{8}) = 4 \geq \frac{-4}{3}log_2(s) \tag{2}$$

Add $2log_2(s)$ to both size of (2):

$$2 * log_2(s) + 4 \leq \frac{-4}{3}log_2(s) + 2log_2(s) = \frac{2}{3}log_2(s) = \frac{2}{3}g(s)$$

Therefore, $4 + log_2(s) * 2 \geq \frac{2}{3}log_2(s)$ or $4 + log_2(s) * 2 \in \Omega(log_2(s))$. (**)

From (*) and (**), we have $4 + log_2(s) * 2 \in \Theta(log_2(s))$. $\square$