Problem Set 4

Ruijian An & Make Zhang & Haoran Fang

Q1. Little-Oh

(a) Want to prove: $\forall a, b \in \mathbb{R}^+, a < b \Rightarrow n^a \in O(n^b)$
Expanding the definition of little-oh, want to prove:
$\forall a, b \in \mathbb{R}^+, a < b \Rightarrow \forall c \in \mathbb{R}^+, \exists n_0 \in \mathbb{R}^+, \forall n \in \mathbb{N}, n \geq n_0 \Rightarrow n^a \leq cn^b.$

Proof:
$Let\ a, b \in \mathbb{R}^+.\ Assume\ a < b.$
$Let\ c \in \mathbb{R}^+, n \in \mathbb{N}, n_0 = c^{\frac{1}{a-b}}.\ (n_0 \in \mathbb{R}^+\ since\ c \in \mathbb{R}^+)$
$Assume\ n \geq n_0.$
$cn^b = cn^a n^{b-a} \geq cn^a {n_0}^{b-a}\ \ (since\ n \geq n_0)$
$= cn^a (c^{\frac{1}{a-b}})^{b-a}\ \ (since\ n_0 = c^{\frac{1}{a-b}})$
$= cn^a c^{-1} = n^a$

(b) Want to prove: $\forall f, g: \mathbb{N} \to \mathbb{R}^+, g \in o(f) \Rightarrow f \notin O(g)$
Expanding the definition of little-oh and big-oh:
$\forall f, g: \mathbb{N} \to \mathbb{R}^+, \forall c \in \mathbb{R}^+, \exists n_0 \in \mathbb{R}^+, \forall n \in \mathbb{N}, n \geq n_0 \Rightarrow g(n) \leq cf(n)$
$\Rightarrow \forall c_0, n_1 \in \mathbb{R}^+, \exists n \in \mathbb{N}, n \geq n_1 \wedge f(n) > c_0 g(n).$

Proof:
$Let\ c_0, n_1 \in \mathbb{R}^+.$
$Since\ g \in o(f), by\ definiton, for\ c = \dfrac{1}{1 + c_0}, \exists n_0 \in \mathbb{R}^+, \forall n \in \mathbb{N}, n \geq n_0 \Rightarrow g(n) \leq cf(n)$
$Let\ c = \dfrac{1}{1 + c_0}, n_0\ be\ as\ above, n = \lceil n_1 + n_0 \rceil.$
$n = \lceil n_1 + n_0 \rceil \geq (n_1 + n_0), so\ n \geq n_1\ and\ n \geq n_0.$
$so\ g(n) \leq cf(n), since\ g \in o(f(n))\ and\ n \geq n_0.$
$= \frac{1}{1 + c_0} f(n)$
so $f(n) \geq (1 + c_0) g(n)$
$> c_0 g(n),\ since\ g(n) > 0, \forall n \in \mathbb{N}.$
$so\ we\ proved\ \forall c_0, n_1 \in \mathbb{R}^+,\ \exists n \in \mathbb{N}, n \geq n_1 \wedge f(n) > c_0 g(n).$

Q2. A tricky nested loop

(a) (i) $g(n) \in O(f(n))$
(ii) $f(n)\ and\ g(n)\ are\ eventually\ \geq b$
(iii) $b > 1$
Want to prove $log_b(g(n)) \in O(log_b f(n))$

Proof:

From (i), we know $\exists\, c, n_0 \in \mathbb{R}^+, \forall\, n \in \mathbb{N}, n \geq n_0 \Rightarrow g(n) \leq cf(n)$.

From (ii), we know $\exists n_1 \in \mathbb{R}^+, \forall n \in \mathbb{N}, n \geq n_1 \Rightarrow f(n) \geq b \wedge g(n) \geq b$.

Want to prove $\exists\, c_0, n_2 \in \mathbb{R}^+, \forall\, n \in \mathbb{N}, n \geq n_2 \Rightarrow log_b(g(n)) \leq c_0 log_b(f(n))$.

$Let\ c, n_0, n_1, b\ be\ as\ above.$

$Let\ c_0 = |log_b c| + 1, n_2 = n_0 + n_1, n \in \mathbb{N}. Assume\ n \geq n_2.$

$By\ fact\ (ii), we\ know\ log_b(g(n)) \geq 1 \wedge log_b(f(n)) \geq 1, since\ n \geq n_2 \geq n_1.$

$$
\begin{aligned}
c_0 log_b(f(n)) &= (|log_b c| + 1) log_b(f(n)) \\
&= |log_b c| log_b(f(n)) + log_b(f(n)) \\
&\geq |log_b c| + log_b(f(n)), \quad since\ log_b(f(n)) \geq 1. \\
&\geq log_b c + log_b(f(n)) \\
&\geq log_b(g(n)),\ since\ from\ (i), g(n) \leq cf(n), so\ log_b(g(n)) \leq \\
&\qquad\qquad\qquad\qquad\qquad\qquad log_b c + log_b(f(n)).
\end{aligned}
$$

(b) After k iterations, $i_k = 3^k\ and\ loop\ will\ end\ when\ i_k \geq b.$

$so\ when\ k \geq log_3 b, the\ loop\ ends.$

$Therefore, the\ loop\ would\ still\ work\ at\ k-1,$

$so\ 3^{k-1} < b,$

$so\ k < log_3 b + 1.$

$Then\ we\ want\ the\ smallest\ integer\ k\ such\ that\ loop\ ends, so\ k = \lceil log_3 b \rceil$

$$so\ there\ will\ be\ \sum_{b=1}^{n} \lceil log_3 b \rceil\ iterations.$$

(c) After k iterations, $b_k = k$, and the outer loop ends when $b_k > n$.

As a result, there will be n iterations for outer loop.

The total cost would be $\sum_{b=1}^{n} \lceil log_3 b \rceil$.

By fact 1, $log_3 b \leq \lceil log_3 b \rceil < log_3 b + 1$.

Then $\sum_{b=1}^{n} \lceil log_3 b \rceil < \sum_{b=1}^{n} (log_3 b + 1), since\ Fact\ 1$

$$
\begin{aligned}
&= \textstyle\sum_{b=1}^{n} (log_3 b) + n \\
&= log_3(n!) + n \\
&< \ln(n!) + n, since\ e < 3.
\end{aligned}
$$

$$By\ fact\ 2\ and\ part\ (a),$$

$$n \in O(n \ln n)$$

$$\ln(n!) \in O(\ln e^{\text{n}\ln n - n + \frac{1}{2} lnn})$$

$$nln - n + \frac{1}{2} \ln n \in O(n \ln n)$$

$$\ln(n!) \in O(n \ln n)$$

$$
\begin{aligned}
\ln(n!) + n &\in O(n + \ln e^{\text{n}\ln n - n + \frac{1}{2} lnn}) \\
&= O\left(n + nlnn - n + \tfrac{1}{2} \ln n\right) = O(nln)
\end{aligned}
$$

Q3. Algorithm analysis

(a) $Let\ n \in \mathbb{N}, let\ L\ be\ an\ aribitrary\ list\ of\ length\ n.$
$The\ loop\ iterates\ at\ most\ (n-1) times, since\ x +$
$y\ increases\ at\ least\ 1\ for\ each\ loop.$
$Each\ loop\ iteration\ counts\ as\ one\ basic\ operation.$
$and\ the\ assignment\ could\ be\ considered\ as\ one\ basic\ operation.$
$so\ the\ function\ takes\ at\ most\ n-1\ steps$
so WC(n)$\in O(n)$

(b) $Let\ n \in \mathbb{N}, let\ L\ be\ an\ \ list\ of\ length\ n\ from\ a\ worst\ case\ input\ family.$
$that\ is\ L\ is\ a\ list\ such\ that\ L[0] = 0, and\ all\ other\ elements\ are\ 1.$
The first loop increases x +y by 1 since L[0] = 0
The following loops increases x + y by 2 since all other elements are 1.
After k iterations, $(x+y)_k = 1 + \ 1 + 2(k-1) = \ 2k$
The loop will end when $(x+y)_k = 2k \ \geq n$, so the loop iterates $\left\lceil \frac{n}{2} \right\rceil$ times.
Then each loop iteration counts as one basic operation.
As a result, the total cost equals $\left\lceil \frac{n}{2} \right\rceil$.
so $WC(n) \in \Omega(n)$

(c) Want to show $BC(n) \notin \Theta(n)$
By definition, only need to show $BC(n) \notin \Omega(n)$

Proof:
$Let\ n \in \mathbb{N}, L\ be\ an\ list\ of\ length\ n\ such\ that\ L[2k] = 0\ and\ L[2k+1] = 1\ for\ k \in \mathbb{N}.$
$By\ observation, (2k+1)th\ iteration\ increses\ x+y\ by\ 1.$
$(2k)th\ iteration\ incrses\ x+y\ by\ (k+1)$

$$After\ (2k+1) iterations, (x+y)_{2k-1} = k + \sum_{i=1}^{k-1}(i+1) \ = k + \frac{(2+k)(k-1)}{2}$$

$$= \frac{1}{2}k^2 + \frac{3k}{2} - 1$$

$$After\ 2k\ iterations, (x+y)_{2k} = k + \sum_{i=1}^{k}(i+1) = \ k + \ \frac{(2+k+1)(k)}{2} = \frac{1}{2}k^2 + \frac{5k}{2}$$

The loop will end when $(x+y)_k \geq n\ and\ (x+y)_{k-1} \ < n$, *then there will be at most* $\lceil(\sqrt{2n})\rceil\ iterations$, *because* $(x+y)_{2k-1} \ \geq \frac{1}{2}k^2\ and\ (x+y)_{2k} \ \geq \frac{1}{2}k^2$,
$then\ solve\ for\ smallest\ k,$
$such\ that \frac{1}{2}k^2 \geq \ n\ and\ this\ k\ is\ greater$ than $the\ actual\ number\ of\ iterations.$
Each iteration counts as one basic operation.

As a result, the total cost is $\lceil(\sqrt{2n})\rceil \in O(\sqrt{n})$

So $O\left(\sqrt{n}\right)$ *is an upper bound for* $BC(n), BC(n) \in O\left(\sqrt{n}\right)$, *then* $BC(n) \notin \Omega(n)$.

Then BC(n)$\notin \Theta(n)$