Daniel Ty

TCSS 343

Assignment 2

Due 10/30/2020

2.1 UNDERSTAND

1.

Pseudocode:

OS(A[a…b])

If $(a \geq b)$

return A

k = minx(A[a…b])

If $(a \ < \ b \ and \ A[a] \ is \ even \ or \ a \ == \ k)$

return A[a] || OS([$a + 1 \ldots b$])

If $(a \ < \ b \ and \ A[a] \ is \ odd \ and \ a \ < \ k)$

return A[k] || OS(A[$a + 1 \ldots k - 1$]) || A[a] || A([$k + 1 \ldots b$])

End OS

2.

(I am assuming array concatenation takes dn operations.)

$$T(n) = \begin{cases} c \ if \ n \leq 1 \\ T(n-2) + dn \ if \text{ n > 1} \end{cases}$$

$$T(n) = \ T(n-2) + dn$$

$$= T(n-4) + d(n-2) + dn$$

$$= T(n-6) + d(n-4) + d(n-2) + dn$$

$$= \cdots$$

$$= T(n-k) + \sum_{i=0}^{k-2} d(n-2i)$$

$$T(n-(n-1)) = T(1) = c$$

Substitute k = $n-1$

$$c + d\sum_{i=0}^{n-3}(n-2i)$$

$$c + d\sum_{i=3}^{n} 2i = c + 2d\sum_{i=3}^{n} i = c + 2d\frac{n(n+1)}{2} - d = c + dn^2 + dn - d = \Theta(n^2)$$

Upper Bound

Lemma: The recurrence $T(n) \leq bn^2 \; for \; all \; n > n_0$.

Proof:

Base Case ($n = 1$):

$$T(1) = c \leq b * 1^2 = b$$

The base case is true if $b \geq c$.

Inductive Hypothesis: Let n > 1, and assume $T(k) \leq bk^2 \; for \; all \; 1 \leq k < n$

Inductive Step ($n > 1$):

$$T(n) = T(n-2) + dn$$

$$\leq b(n-2)^2 + dn$$

$= bn^2 - 4bn + 4b + dn$

$\leq bn^2$

The last step is true if $4bn \geq 4b + dn$. Since $n > 1$, we know $4bn > 4b$, so this condition holds if $4bn > dn$. We select $b = max(c, d)$.

Lower Bound

Lemma: The recurrence $T(n) \geq an^2 \; for \; all \; n > n_0$.

Proof:

Base Case ($n = 1$):

$T(1) = c \geq a * 1^2 = a$

The base case is true if $a \leq c$.

Inductive hypothesis: Let n > 1, and assume $T(k) \geq ak^2 \; for \; all \; 1 \leq k < n$

Inductive Step ($n > 1$):

$T(n) = T(n-2) + dn$

$\geq a(n-2)^2 + dn$

$= an^2 - 4an + 4a + dn$

$\geq an^2$

The last step is true if $4an \leq 4a + dn$. Since $n > 1$, we know $4an \geq 4a$, so this condition only holds if $4an < dn$. We select $a = \min(c, \frac{d}{4})$.

Considering both lemmas, we get $an^2 \leq T(n) \leq bn^2 \; for \; all \; n \geq 1$. So, the recurrence $T(n) \epsilon \Theta(n^2)$.

3.

Pseudocode:

SOS(A[a…b])

If $(a \geq b)$

Return A

If $(b = a + 1)$

Return pairsort(A[a], A[b])

$t_1 = a + \left\lfloor \frac{b-a}{3} \right\rfloor$

$t_2 = b - \left\lceil \frac{b-a}{3} \right\rceil$

A’ = SOS(A[$a \ldots t_2$]) || A($t_2 + 1 \ldots b$)

A’’= A’[$a \ldots t_1$] || SOS(A’[$t_1 + 1 \ldots b$])

A’’’= SOS(A’’[$a \ldots t_2$]) || A’’[$t_2 + 1 \ldots b$]

If $(b > a + 1)$

Return A’’’

End SOS

4.

$$T'(n) = \begin{cases} c \; if \; n \leq 1 \\ 3T\left(\frac{2n}{3}\right) + dn \; if \; n > 1 \end{cases}$$

Master’s Theorem:

$$a = 3$$

$$\frac{2n}{3} = \frac{n}{\left(\frac{3}{2}\right)}, b = \frac{3}{2}$$

$$f(n) = dn$$

$$dn \in O\left(n^{\log_{\frac{3}{2}} 3-\epsilon}\right) \; for\ 0 < \epsilon < 2.7095 \;\; (\log_{\frac{3}{2}} 3 = 2.7095)$$

So, we can conclude $T(n) \epsilon\ \Theta(n^{\log_{3/2} 3})$.

2.2 EXPLORE

1.

Self-reduction 1:

$$C2(S[a \dots b]) = \begin{cases} \epsilon\ if\ a \geq b\ and\ len(S[a])\ is\ odd \\ S[a]\ if\ a \geq b\ and\ len(S[a])\ is\ even \\ C2(S[a+1 \dots b])\ if\ a < b\ and\ len(S[a])\ is\ odd \\ S[a]\ ||\ C2(S[a+1 \dots b])\ if\ a < b\ and\ len(S[a])\ is\ even \end{cases}$$

Self-reduction 2:

$$C2(S[a \dots b]) = \begin{cases} \epsilon\ if\ a \geq b\ and\ len(S[a])\ is\ odd \\ S[a]\ if\ a \geq b\ and\ len(S[a])\ is\ even \\ C2(S[a \dots l])\ ||\ C2(S[r \dots b])\ if\ a < b \\ with\ l = \left\lfloor \frac{b-a}{2} \right\rfloor, r = \left\lceil \frac{b-a}{2} \right\rceil \end{cases}$$

2.

Self-reduction 1 pseudocode:

C2(S[a…b])

If ($a \geq b$ and len(S[a]) is odd)

Return ϵ

If ($a \geq b$ and len(S[a]) is even)

Return S[a]

If (a < b and len(S[a]) is odd)

Return C2(S[a+1…b])

If (a < b and len(S[a]) is even)

Return S[a] || C2(S[a+1…b])

End C2(S[a…b])

Self-reduction 2 pseudocode:

C2(S[a…b])

If ($a \geq b$ and len(S[a]) is odd)

Return ϵ .

If ($a \geq b$ and len(S[a]) is even)

Return S[a].

If (a < b)

l = $\left\lfloor \frac{b-a}{2} \right\rfloor$

r = $\left\lceil \frac{b-a}{2} \right\rceil$

Return $C2(S[a \ldots l])$ || $C2(S[r \ldots b])$.

End C2(S[a…b])

3.

The worst-case runtime for both solutions is $\Theta(n)$

2.3 EXPAND

1.

Assume n is a power of 9, $n = 9^k$. $k = \log_9 n$

$$T(n) = 3T\left(\frac{n}{9}\right) + \sqrt{n}$$

$$= 3\left(3T\left(\frac{n}{81}\right) + \sqrt{\frac{n}{9}}\right) + \sqrt{n} = 9T\left(\frac{n}{81}\right) + 3\sqrt{\frac{n}{9}} + \sqrt{n} = 9T\left(\frac{n}{81}\right) + \sqrt{n} + \sqrt{n}$$

$$= 9\left(3T\left(\frac{n}{729}\right) + \sqrt{\frac{n}{81}}\right) + \sqrt{n} + \sqrt{n} = 27T\left(\frac{n}{729}\right) + 9\sqrt{\frac{n}{81}} + \sqrt{n} + \sqrt{n}$$

$$= 27T\left(\frac{n}{729}\right) + \sqrt{n} + \sqrt{n} + \sqrt{n}$$

$$= \cdots$$

$$= 3^k T\left(\frac{n}{9^k}\right) + k\sqrt{n}$$

$$= 3^{\log_9 n} T(1) + \log_9(n)\sqrt{n}$$

$$= 3^{\log_9 n} + \log_9(n)\sqrt{n} \in \theta(\sqrt{n}\log_9 n)$$

$$g(n) = \sqrt{n}\log_9 n$$

2.

Lemma: $T(n) \le b\sqrt{n}\log_9 n \; for \; all \; n > n_0$

Proof:

Base case ($n = 9$):

$$T(9) = 3T\left(\frac{9}{9}\right) + \sqrt{9} = 3 + 3 = 6 \le b\sqrt{9}\log_9 9 = 3b$$

$$2 \le b$$

The base case is true if $b \ge 2$.

Inductive hypothesis: $Let\ n > 9, and\ assume\ T(k) \le b\sqrt{k}\log_9 k\ \ for\ all\ 9 \le k < n.$

Inductive Step ($n > 9$):

$T(n) = 3T\left(\frac{n}{9}\right) + \sqrt{n}$ definition of T(n)

$\le\ 3b\sqrt{\frac{n}{9}}\log_9 \frac{n}{9} + \sqrt{n}$ Inductive hypothesis

$\le\ b\sqrt{n}\log_9 \frac{n}{9} + \sqrt{n}$

$\le\ b\sqrt{n}\log_9 n - b\sqrt{n} + \sqrt{n}$

$\le b\sqrt{n}\log_9 n$

The last step is only true if $-b\sqrt{n} + \sqrt{n} \le 0, that\ is, if\ b \ge 1.$

Combining with the constraint from the base case, we can select $b = 2$.

By Induction, we have shown for all $n \ge 9$ that $T(n) \le b\sqrt{n}\log_9 n$. In other words, $T(n) \epsilon\ O\big(g(n)\big)$.

3.

Lemma: $T(n) \ge b\sqrt{n}\log_9 n\ \ for\ all\ n >\ n_0$

Proof:

Base Case ($n = 1$):

$T(1) = 1 \ge b\sqrt{1}\log_9 1 = b * 1 * 0 = 0$

This base case is true for all b.

Inductive hypothesis: Let $n \ge 1, assume\ T(k) \ge b\sqrt{k}\log_9 k\ \ for\ all\ 1 \le k < n.$

$T(n) = 3T\left(\frac{n}{9}\right) + \sqrt{n}$ definition of T(n)

$\geq 3b\sqrt{\frac{n}{9}}\log_9 \frac{n}{9} + \sqrt{n}$ Inductive hypothesis

$\geq b\sqrt{n}\log_9 \frac{n}{9} + \sqrt{n}$

$\geq b\sqrt{n}\log_9 n - b\sqrt{n} + \sqrt{n}$

$\geq b\sqrt{n}\log_9 n$

The last step is only true if $-b\sqrt{n} + \sqrt{n} \geq 0, that\ is, if\ b \leq 1$.

Combining with the constraint from the base case, we can select $b = 1$.

By Induction, we have proven for $n \geq 1\ that\ T(n) \geq b\sqrt{n}\log_9 n$. In other words, $T(n) \epsilon\ \Omega\big(g(n)\big)$. This combined with $T(n) \epsilon\ O\big(g(n)\big)$ above proves that $T(n) \epsilon\ \Theta\big(g(n)\big)$.