CSC236 fall 2016

time complexity

Danny Heap
heap@cs.toronto.edu / BA4270 (behind elevators)
http://www.cdf.toronto.edu/~csc236h/fall/
416-978-5899

Using Introduction to the Theory of Computation, Chapter 3

Outline

complexity of recursive functions

Notes

binary search

Recursive $T(n)$

A - array
x - value searched for
b - begin
e - end.

Should prove T is montonic increasing

$e - b + 1 = n$

```
def recBinSearch(x, A, b, e) :
  if b == e :
    if x <= A[b] :
      return b
    else :
      return e + 1
  else :
    m = (b + e) // 2 # midpoint
    if x <= A[m] :
      return recBinSearch(x, A, b, m)
    else :
      return recBinSearch(x, A, m+1, e)
```

$\to c_1$

c_2

$\max - c_3$

c'

$$T(n) = \begin{cases} c' & n = 1 \\ 1 + \max\left(T(\lceil n/2 \rceil), T(\lfloor n/2 \rfloor)\right) & \end{cases}$$

$c'' \to 1$

$\max\left(T(m-b+1), T(e-m)\right)$

See Notes

must prove $m - b + 1 = \lceil \frac{n}{2} \rceil$, $e - (m+1) + 1 = \lfloor n/2 \rfloor$

Lower bound on $T(n)$... by unwinding

$$T(n) = \begin{cases} c' & n = 1 \\ 1 + \max(T(\lceil n/2 \rceil), T(\lfloor n/2 \rfloor)) \end{cases}$$

suppose $n = 2^k$, some $k \in \mathbb{N}$

$$\begin{aligned}
T(n) = T(2^k) &= 1 + T(2^{k-1}) \\
&= 1 + 1 + T(2^{k-2}) \\
&= 1 + 1 + 1\ T(2^{k-3}) \\
&\vdots \\
&= \underbrace{1 + 1 + \ldots + 1 +}\ T(2^{k-k}) \\
&= k + c' = \lg(n) + c'
\end{aligned}$$

Idea $T(n) \in \Theta(\lg n)$. ← this needs to be proved.

To prove $T \in \Omega(\lg n)$ must show

$$\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow T(n) \geq c \lg n$$

Lower bound on $T(n)$... by proof

Pick $B = 2$, Pick $c = 1$

Let $n \in \mathbb{N}$, $n \geq B$; assume $T(i) \geq c\lg(i)$ $\forall i$ $B \leq i < n$

$$T(n) = 1 + T(\lceil n/2 \rceil)$$ # T is monotonic $\uparrow$ $n \geq B > 1$

$$\geq 1 + c\lg(\lceil n/2 \rceil)$$ # $B \leq \lceil \frac{n}{2} \rceil < n$, $n \geq 2$

By IH.

$$\geq 1 + c\lg\left(\frac{n}{2}\right)$$ # $\lg$ monotonic $\uparrow$

$$= 1 + c(\lg n - \lg 2)$$

$$= 1 + c(\lg n - 1)$$

$$= 1 - c + c\lg n$$

$$\geq c\lg n$$ # $1 - c \geq 0$ # $c = 1$.

Base case ↓

$P(2)$: $T(2) = 1 + T(1) = 1 + c' \geq \lg 2$ ✓

Upper bound on $T(n)$

Prove $T \in O(\lg n)$

trouble! Pick $c =$ ____ . Pick $B =$ ____ .

Let $n \in \mathbb{N}$, $n \geq B$. assume $\forall i$, $B \leq i < n$, $T(i) \leq c \lg n$.

$$T(n) = 1 + T(\lceil \tfrac{n}{2} \rceil) \quad \# \ n > 1,\ T \text{ monotonic}$$

$$\leq 1 + c \lg(\lceil \tfrac{n}{2} \rceil) \quad \# \text{ By IH } B \leq \lceil n/2 \rceil < n$$

$$\leq 1 + c \lg\left(\frac{n+1}{2}\right)$$

$$= 1 + c(\lg(n+1) - \lg 2)$$

$$= \underline{1 - c + c \lg(n+1)}$$

:(

make claim $\underline{\text{stronger}}$ e.g. $T(n) \leq c \cdot \lg(n-1)$

$\leq c \lg n$.

try to make this work with original IH

Notes

$$m - b + 1 = \left\lfloor \frac{e+b}{2} \right\rfloor - b + 1$$

$$= \left\lfloor \frac{e+b-2b+2}{2} \right\rfloor \quad \# \ \lfloor x \rfloor + k = \lfloor x + k \rfloor \quad \# \text{ if } k \in \mathbb{Z}$$

$$= \left\lfloor \frac{e-b+1+1}{2} \right\rfloor$$

$$= \left\lceil \frac{e-b+1}{2} \right\rceil \quad \# \ \left\lfloor \frac{k+1}{2} \right\rfloor = \left\lceil \frac{k}{2} \right\rceil \quad \# \ \forall k \in \mathbb{N}$$

$$= \left\lceil \frac{n}{2} \right\rceil$$

$$e - m = e - \left\lfloor \frac{e+b}{2} \right\rfloor = e + \left\lceil -\frac{e+b}{2} \right\rceil \quad \# \ -\lfloor x \rfloor = \lceil -x \rceil$$

$$= \lfloor n/2 \rfloor \qquad = \left\lceil e - \frac{e+b}{2} \right\rceil \quad \# \ \lceil k + x \rceil = \lceil x \rceil + k, \ k \in \mathbb{Z}$$

$$= \left\lceil \frac{e-b}{2} \right\rceil = \left\lfloor \frac{e-b+1}{2} \right\rfloor \quad \# \ \left\lfloor \frac{k+1}{2} \right\rfloor = \left\lceil \frac{k}{2} \right\rceil$$