CSE 102 5-7-24

Recall

$$\ln(n+1) \le H_n \le 1 + \ln(n)$$

divide by $\ln(n)$

$$\frac{\ln(n+1)}{\ln(n)} \le \frac{H_n}{\ln(n)} \le \frac{1+\ln(n)}{\ln(n)}$$

$$\downarrow \qquad \downarrow \qquad \downarrow \quad \text{as } n \to \infty$$

$$1 \qquad 1 \qquad 1$$

so

$$\lim_{n\to\infty} \frac{H_n}{\ln(n)} = 1 \quad \therefore H_n \sim \ln(n)$$

$$\therefore \quad H_n = \Theta(\ln(n))$$

Recall :

$$t(n) = 2(n+1)H_n - 4n$$

Thus

$$t(n) = \Theta(n \log n)$$

Actually

$$t(n) \sim 2n\ln(n)$$

i.e. $t(n) = 2n\ln(n) + o(n \ln n)$

Randomization of Quicksort

RandPartition(A, p, r)

1. $i = \text{Rand}(p, r)$ // $p \leq i \leq r$
2. $A[i] \leftrightarrow A[r]$
3. return Partition(A, p, r)

RandQuicksort(A, p, r)

1. if $p < r$
2. $\quad q = \text{RandPartition}(A, p, r)$
3. $\quad \text{RandQuicksort}(A, p, q-1)$
4. $\quad \text{RandQuicksort}(A, q+1, r)$

chapter 9

Problem:

find both min and max in an array $A[1 \ldots n]$

$\min(m_1, m_2)$

1. if $m_1 < m_2$
2. return m_1
3. return m_2

$\max(M_1, M_2)$

1. if $M_1 > M_2$
2. ...
3. ...

<u>min-max(A, p, r)</u> Pre: $p \le r$

```
if p = r
    return (A[p], A[p])
else
    q = ⌊(p+r)/2⌋
    (m1, M1) = min-max(A, p, q)
    (m2, M2) = min-max(A, q+1, r)
    return (min(m1, m2), max(M1, M2))
```

Let $T(n)$ = # comparisons (Best, avg, worst) by min-max$(A, 1, n)$

$$T(n) = \begin{cases} 0 & n = 1 \\ T\left(\lceil \frac{n}{2} \rceil\right) + T\left(\lfloor \frac{n}{2} \rfloor\right) + 2 & n \geq 2 \end{cases}$$

Exercise

- use Master thm to show

$$T(n) = \Theta(n)$$

- show $T(n) = 2n - 2$ is the exact solution by direct substitution.

Exercise (hw)

Design a recursive algorithm that finds (min, max) in $A[1 \cdots n]$ in exactly $\lceil \frac{3n}{2} \rceil - 2$ comparisons.

$$2n-2 \sim 2n$$

$$\lceil \tfrac{3n}{2} \rceil - 2 \sim \tfrac{3}{2}n \leftarrow \text{better}$$

See sec. 9.1 for a description of an iterative algorithm.

9.2 Selection Problem

Defn

The i^{th} order statistic of an array $A[1 \ldots n]$ consisting of distinct elements is the i^{th} smallest element. Equivalently it is the unique element that is greater than exactly $(i-1)$ other elements. Equivalently: if A were sorted, it is $A[i]$.

Problem

Given $A[1 \ldots n]$ and i $(1 \le i \le n)$,
find the i^{th} order statistic.

One solution: sort first and
return $A[i]$. runtime: $\Theta(n \log n)$.

Another solution: RandSelect( )

RandSelect(A, p, r, i)

$\begin{cases} \text{Pre: } 1 \le i \le r-p+1 \\ \text{Pre: distinct} \end{cases}$

```
if p = r
    return A[p]
q = RandPartition(A, p, r)
k = q - p + 1    // len of A[p...q]
if k = i
    return A[q]
else if i < k
    return RandSelect(A, p, q-1, i)
else    // k < i
    return RandSelect(A, q+1, r, i-k)
```

Recall: RandPartition(A, p, r) does

$$\overbrace{A[p \cdots (q-1)]}^{\text{len}=k} < A[q] < A[(q+1) \cdots r]$$

Exercises

- write non-randomized version.
- Prove correctness
- Write a recurrence for worst case runtime.

answer $T(n) = \begin{cases} 0 & n=1 \\ T(n-1) + (n-1) & n \geq 2 \end{cases}$

- Show $T(n) = \frac{n(n-1)}{2}$, so $T(n) = \Theta(n^2)$.

Let $t(n)$ be the average case # comparisons by RandSelect(A,1,n,i)

Assume each permutation of $A[1\cdots n]$ is equally likely as input.

Then

$$t(n) = \sum_{q=1}^{n} \frac{1}{n}\cdot\Big((n-1) + P(i<q\le n)\,t(q-1) + P(1\le q<i)\cdot t(n-q)\Big)$$

here
$$\begin{cases} P(i<q\le n) = \frac{n-i}{n} \\ P(1\le q<i) = \frac{i-1}{n} \end{cases}$$

Thus

$$t(n) = (n-1) + \frac{1}{n}\cdot\sum_{q=1}^{n}\Big(\big(\tfrac{n-i}{n}\big)t(q-1) + \big(\tfrac{i-1}{n}\big)t(n-q)\Big)$$

algebra (exercise)

$$\boxed{t(n) = (n-1) + \left(\frac{n-1}{n^2}\right) \cdot \sum_{q=1}^{n-1} t(q)}$$

Exercise :

show that $t(n) = \Theta(n)$

Hint: show by induction :

$$\forall n \geq 1 \ : \ t(n) \leq 2n$$