CS140: Algorithms

Z Sweedyk
Lecture 4
1/30/01

Today

- Select
- Average case analysis

Select

- Input: Set of (distinct) integers S and an integer k
- Output: k^{th} smallest integer in S

Select: special cases

- k=1
- k=n
- k=n/2

How fast can we solve these cases?

Select: Take 1

Select(S,k)
Let x=S[0]
Partition S into
$S_1 = \{y \in S-\{x\} \mid y < x\}$
$S_2 = \{y \in S-\{x\} \mid y > x\}$
If $||S_1|| \geq k$ then return Select(S_1,k)
Else if $||S_1|| = k-1$ then return x
Else return Select(S_2, $k-||S_1||-1$)

Analysis of Select(S,k)

In worst we get

$$T(n) \leq T(\_\_\_) + \_\_\_ = \Theta(\_\_\_)$$

Suppose …

- Suppose we could choose x so that the recursive call is always on a set of size n/b, for some constant b>1.
- Then T(n)=T(___)+___

  =_________

To warm up …

Suppose we could choose x so that the recursive call is **typically** on a set of size n/b, for some constant b>1.

Average-case analysis

What does average-case mean?

- Randomized algorithm on worst-case input
- Deterministic algorithm with a known input distribution
- Deterministic algorithm on worst-case input using amortized cost

A brief tour of (discrete) probability theory…

- Sample space, events, probability
- Discrete probability distributions
- Discrete random variables
- Expectation
- Conditional Probability/Expectation

Experiment 1

- Experiment: A fair coin is flipped
- Sample space:____________________
- Events:__________________________
- Probabilities:______________________

Experiment 2

- Experiment: Two fair coins are flipped
- Sample space: _____________________
- Events:___________________________
- Probabilities:_______________________

Experiment 3

- Experiment: A fair die is tossed
- Sample space: ________________
- Events: ______________________
- Probabilities:__________________

Discrete Probability Distribution

Assigns a real number to every subset of the sample space such that:

- $P(A) \geq 0$ for any event A
- $\Sigma_{A \in S}\ P(A) = 1$
- $P(A \text{ or } B) = P(A)+P(B)$ for disjoint events A,B

Discrete Random Variable X

- Assigns a real number to each outcome.
- Experiment: Toss fair coin
  - If head then X=1
  - If tail then X=0
- Sample space = ________________
- Probabilities:
  - What is P(X=0)?
  - What is P(X=1)?
  - What is $P(X \geq 1)$?

Expectation

- $E[X] = \sum_{x \in S} x\ P(X=x)$
- $E[X^2] = \sum_{x \in S} x^2\ P(X=x)$
- $Var(X) = E[(X-E[X])^2]$

Discrete Random Variable X
Example continued: P(0)=P(1)=1/2

Expectation:

$E[X] = \sum_{x \in S} x\ P(X=x)=$______________

$E[X^2] = \sum_{x \in S} x^2\ P(X=x)=$_____________

$Var(X) = E[(X-E[X])^2]=$_____________

Example

- A biased coin is tossed n times:
  P(H)=p, P(T)=1-p=q
- X is the number of heads
- P(X=k) =________________
  (Binomial distribution)
- E[X] =___________________
- $E[X^2]$ = __________________

Conditional Probability

- Let A and B be events such that P(B)>0
- Then $P(A|B) = P(A\cap B)/P(B)$

Conditional Probability: Example

Experiment: Toss a fair die

- A is the event that a 2 is rolled
- B is the event the number rolled is even
- C is the event the number rolled is odd
- What is P(A|B)?
- What is P(B|A)?
- What is P(A|C)?
- What is P(C|A)?

Conditional Probability: Properties

Let $A_1, A_2, \ldots, A_k$ be disjoint events that partition the sample space.

Then for any event A

$P(A) = \Sigma_{i=1..k} P(A \mid A_i) P(A_i)$

Average-case analysis: T(n)

What does average-case mean?

- Randomized algorithm on worst-case input
  - $T(n)=Max_{\text{inputs I of size n}}E(\text{\# steps on input I})$

Average-case analysis: T(n)

What does average-case mean?

- Deterministic algorithm with a known input distribution
  - E[# steps on input I], where I is chosen at random from all inputs of size n

Average-case analysis: T(n)

What does average-case mean?

- Deterministic algorithm on worst-case input using amortized cost
  - We'll define this next time

Average-case analysis

What does average-case mean?

- **Randomized algorithm on worst-case input**
- Deterministic algorithm with a known input distribution
- Deterministic algorithm on worst-case input using amortized cost

Randomized Select in Expected Linear Time

Randomized Select(S,k)

Choose x randomly from S

Partition S into

S_1 = {y ε S-{x} | y < x}

S_2 = {y ε S-{x} | y > x}

If $||S_1|| \geq$ k then return Randomized Select(S_1,k)

Else if $||S_1||$ = k-1 then return x

Else return Randomized Select(S_2, k-$||S_1||$-1)

Analysis of Randomized-Select

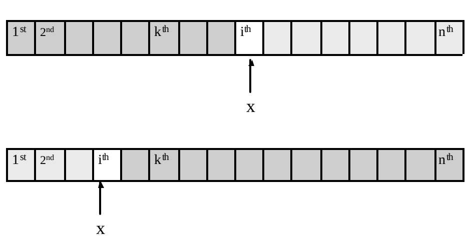
E[T(n)] = E[T(N)] + cn

Where N is the size of the input to the recursive call.

Analysis of Randomized-Select

$$E[T(n)] = E[T(N)] + cn$$
$$= \Sigma_{i=1 \ldots n} E[T(N) \mid rank(x)=i]\, P(rank(x)=i) + cn$$
$$= (1/n)\, \Sigma_{i=1 \ldots n} E[T(N) \mid rank(x)=i] + cn$$

N ≤ max(i-1, n-i) when rank(x)=i



Analysis of Randomized-Select

$$E[T(n)] = E[T(N)] + cn$$
$$= \Sigma_{i=1 \ldots n} E[T(N) \mid rank(x)=i]\, P(rank(x)=i) + cn$$
$$= (1/n)\, \Sigma_{i=1 \ldots n} E[T(N) \mid rank(x)=i] + cn$$
$$\leq (1/n)\, \Sigma_{i=1 \ldots n} E[T(max(i-1,n-i)] + cn$$
$$\leq (2/n)\, \Sigma_{i=\lceil (n+1)/2 \rceil .. n} E[T(i-1)] + cn$$

Analysis of Randomized-Select

$E[T(n)] \le (2/n)\Sigma_{i=\lceil(n+1)/2\rceil..n} E[T(i-1)]+cn$
$= O(n)$

Prove

What is wrong with this picture? A proof that $n^2=O(n)$!!!!

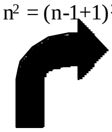
- $1^2=O(1)$
- Assume $(n-1)^2=O(n-1)$
- Then $n^2 = (n-1+1)^2 = (n-1)^2 + 2(n-1) + 1$
  $= O(n-1) + O(n-1) + O(1)$
  $= O(n)$

Back to Basics

- Claim: There exists constant c and M such that $n^2 \le cn$ for all $n \ge M$.
- Proof:
  - Assume c and M are chosen so $M^2 \le cM$.
  - Let n-1be at least M. Suppose $(n-1)^2 \le c(n-1)$.
  - Then $n^2 = (n-1+1)^2 = (n-1)^2 + 2(n-1) + 1$
    $\le c(n-1) + 2(n-1) + 1$
    $= cn + 2(n-1) - c + 1$
- Claim follows iff $c \ge 2(n-1) + 1$. But no constant c can satisfy this requirement.

What is wrong with this picture? A proof that $n^2=O(n)$!!!!

- $1^2=O(1)$
- Assume $(n-1)^2=O(n-1)$
- Then $n^2 = (n-1+1)^2 = (n-1)^2 + 2(n-1) + 1$
  $= O(n-1) + O(n-1) + O(1)$
  $= O(n)$



This is what is wrong!

Analysis of Randomized-Select

Claim: There exists constants d and M such that $E[T(n)] \le d \cdot n$ for all $n \ge M$.

- $E[T(n)] \le (2/n)\ \Sigma_{i=\lceil(n+1)/2\rceil..n} E[T(i-1)] + cn$
- $E[T(n)] \le (2/n)\ \Sigma_{i=\lceil(n+1)/2\rceil..n} d\cdot(i-1) + cn$
  $\le dn$ provided $d>2c$

Average-case analysis

What does average-case mean?

- Randomized algorithm on worst-case input
- **Deterministic algorithm with a known input distribution**
- Deterministic algorithm on worst-case input using amortized cost

Select: Take 1

What if all permutations of S are equally likely?

Select(S,k)

Let x=S[0]
Partition S into
$S_1 = \{y \;\varepsilon\; S-\{x\} \mid y < x\}$
$S_2 = \{y \;\varepsilon\; S-\{x\} \mid y > x\}$
If $\|S_1\| \geq k$ then return Select(S_1,k)
Else if $\|S_1\| = k-1$ then return x
Else return Select(S_2, $k-\|S_1\|-1$)

Analysis

- Let me wave my hands a bit …

Deterministic Select in Linear Time

Select(S,k)

Choose a "good pivot" x ε S
Partition S into
$S_1 = \{y \;\varepsilon\; S-\{x\} \mid y < x\}$
$S_2 = \{y \;\varepsilon\; S-\{x\} \mid y > x\}$
If $\|S_1\| \geq k$ then return Select(S_1,k)
Else if $\|S_1\| = k-1$ then return x
Else return Select(S_2, $k-\|S_1\|-1$)

What is a good pivot?

We say x ε S is a good pivot if its rank is between n/c and (c-1)n/c for some constant c>1.

If we always choose a good pivot we get $\Theta(n)$ running time.

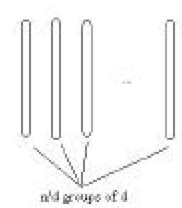
The thing …

- Median of medians pivot

The next few slides are
ANALYSIS – not the algorithm

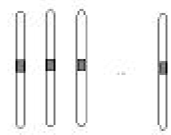
Median of medians pivot

- Divide the input into groups of d.



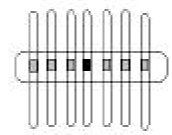
Median of medians pivot

- Divide the input into groups of d.
- Sort each group and mark its median.



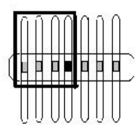
Median of medians pivot

- Divide the input into groups of d.
- Sort each group and mark its median.
- Sort the groups by their medians. Mark median of medians

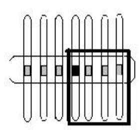


Median of medians pivot

- Elements in upper left quadrant are smaller than median of medians.

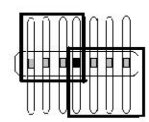


- Elements in lower right quadrant are larger than median of medians.



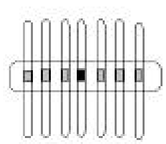
Median of medians pivot

- How many elements of S are smaller than the median of medians?
- How many are larger?



Median of medians

- Median of medians is a good pivot provided d satisfies the following:



BUT

- Finding the good pivot requires a recursive call to Select
- We hadn't counted on this …

New Analysis

1. Divide the input into groups of *5*. Find the median of each group.
2. Find the median of the medians.
3. Partition the input around the median of medians.
4. Recurse on appropriate set of the partition.

1. O(1) time per group, O(n) groups → O(n)
2. T(n/5)
3. O(n)
4. T(3n/4)

Linear selection

$T(n) = T(n/5) + T(3n/4) + O(n) = \Theta(n)$

BUT BE CAREFUL OF DETAILS!