COMP 251

Algorithms & Data Structures (Winter 2021)

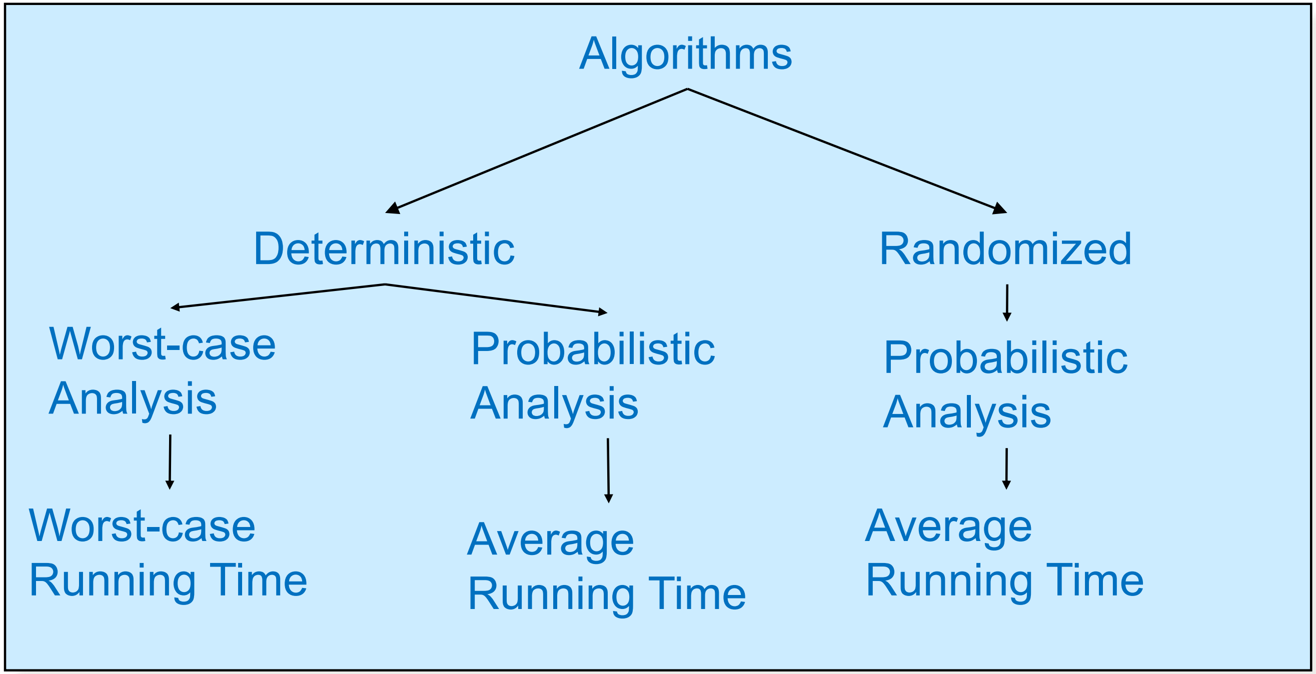
Extras – Randomization and Probabilistic Analysis

School of Computer Science
McGill University

Slides of Langer (2014) & Cormen et al., 2009 & Comp251-Fall McGill & Kleinberg & *Tardos*, 2006 & Lin & Devi (UNC)

Deterministic VS Randomized

- Deterministic Algorithm : Identical behavior for different runs for a given input.
- Randomized Algorithm : Behavior is generally different for different runs for a given input.



2 different views

1. To consider the world as behaving randomly:
   - One can consider traditional algorithms that confront randomly generated input.
     - Average-case analysis => to study the behavior of an algorithm on an “average” input (subject to some underlying random process), rather than a worst-case input.
2. To consider algorithms that behave randomly:
   - Same worst-case input as always, but we allow our algorithm to make random decisions as it processes the input.
     - The role of randomization in this approach is purely internal to the algorithm and does not require new assumptions about the nature of the input.

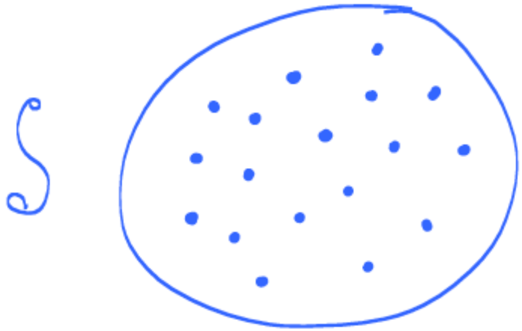
2 different views

When we say quicksort is O(n logn) in the "average case", we could mean two different things.

1. A quicksort algorithm that chooses pivots in some deterministic way has average performance O(n logn), namely averaging over all possible inputs which are equally like to occur.
   - Randomized input
2. For any given input, a quicksort algorithm that chooses pivots randomly takes time O(n log n).
   - Randomized algorithm

Probabilistic Analysis

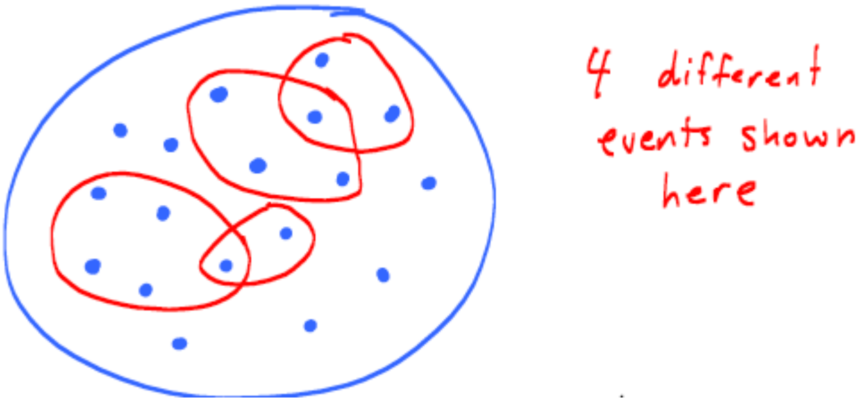
- Definition: Sample Space.
  - A set of possible outcomes of some “experiment” (loose definition)



    - Flip a coin {H, T}
    - Flip a coin 3 times {HHH, HHT, HTH, HTT, THH, THT, TTH, TTT}
    - Roll a dice once {1, 2, 3, 4, 5, 6}
    - Roll a dice twice {(1,1), (1,2), …. , (5,6), (6,6)}
    - Take Comp251 {A, A-, B+, B, ….. , C, D, F }
    - When is your birthday? {1, 2, 3, ….. , 365 }
    - Instance of a sorting algorithm of size 3 {(1,2,3), (1,3,2), …. (3,2,1)}

Probabilistic Analysis

- Definition: Event.
  - A subset of a sample space.



- You flip a coin 3 times and you get 2 heads {THH, HTH, HHT}
- Roll a dice twice and the sum is less than 5 {(1,1), (1,2), (2,1), (1,3), (2,2), (3,1)}
- You pass Comp251 {A, A-, B+, B, ….. , C, D}
- Your birthday is after February 5th? {36, 37 ….. , 365 }

Probabilistic Analysis

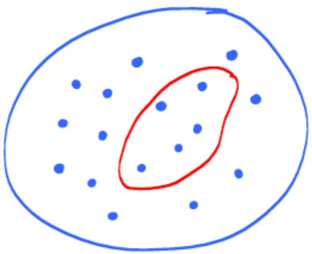
- Definition: Probability distribution on a sample space.
  - A mapping from events of S to real numbers satisfying the following probability axioms:

mapping $p : S \to [0, 1]$

  - Gives the probabilities of occurrence of different events for an experiment.
    - $\Pr\{A\} \geq 0$ *for any event A*
    - $\Pr\{S\} = 1$
    - $\Pr\{A \cup B\} = \Pr\{A\} + \Pr\{B\}$ *for any two mutually exclusive events A and B*

Probabilistic Analysis

- Definition: Discrete Probability Distribution.
  - A probability distribution is discrete if it is defined over a finite or countable infinite sample space.
  - If S is finite and every elementary event $s \in S$ has probability $\Pr\{s\} = \frac{1}{|S|}$, then we have the uniform probability distribution on S.
    - In such a case the experiment is often described as "picking an element of S at random".
      - As an example consider the process of flipping a fair coin.



Let $E \subseteq S$ be some event.

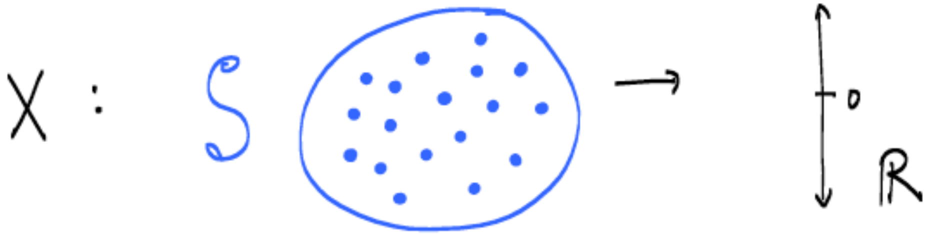
$$p(E) = \sum_{s \in E} p(s)$$

If $p(\ )$ is a uniform distribution

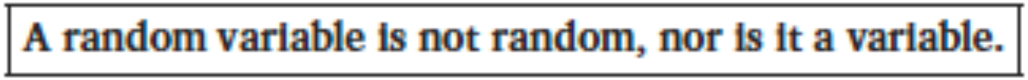
the $p(E) = \frac{|E|}{|S|}$.

Probabilistic Analysis

- Definition: Random Variable.
  - Is a function from the underlying sample space to the natural numbers.



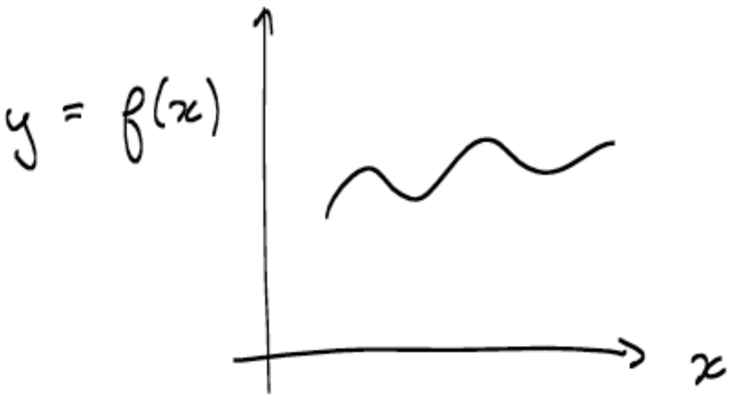
    - It associates a real number with each possible outcome of an experiment, which allows us to work with the probability distribution induced on the resulting set of numbers.


A random variable is not random, nor is it a variable.

- Although random variables are formally not variables at all, we typically describe and manipulate them as if they were variables.

Probabilistic Analysis

- Definition: Random Variable.



f is a mapping from R to R
f is not a variable

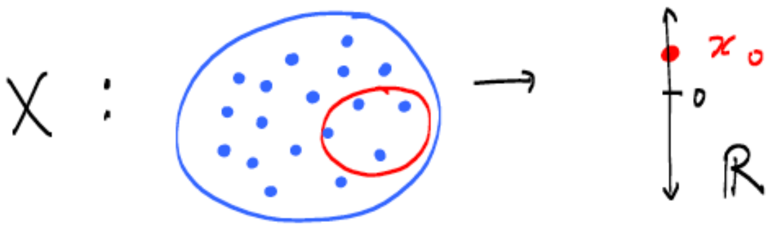
However, when we say y = f(x), then y is a “variable”. In the case of a random variable, the x values (and hence the y values) occur with some probability.

Probabilistic Analysis

- Definition: Random Variable.

Define the event $X = x_0$ to be

$\{ s \in S$ such that $X(s) = x_0 \}$



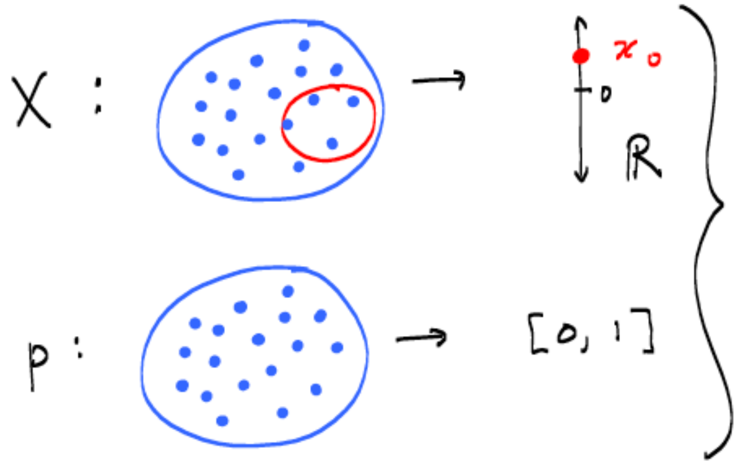
- You flip a coin 3 times and you get $X = x_0$ heads e.g., $x_0=2$.
  - {THH, HTH, HHT}
- You roll a dice twice and the sum is $X = x_0$ e.g., $x_0=5$.
  - {(1,4),(2,3),(3,2),(4,1)}

Probabilistic Analysis

- Definition: Distribution on the random variable X.

$$\Pr(X = x_0) \equiv \sum_{\{s \,:\, X(s) = x_0\}} p(s)$$

- Think of it as probabilities on the values of the random variable X.

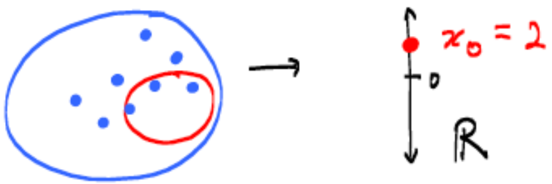


Note: we need two mappings to define this distribution $\Pr(X = x)$

Probabilistic Analysis

- Definition: Distribution on the random variable X.
- You flip a coin 3 times. What is the distribution of the number of heads?

$x_0 = 2 \ \{ THH, HTH, HHT \}$



$|S| = 8$

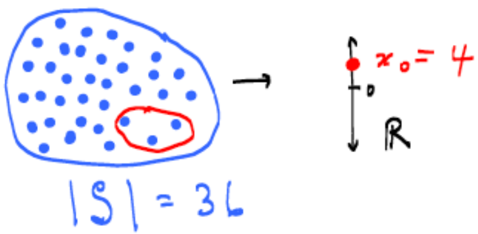
$$\Pr(X = x) = \begin{cases} \frac{1}{8}, & x = 0 \\ \frac{3}{8}, & x = 1 \\ \frac{3}{8}, & x = 2 \\ \frac{1}{8}, & x = 3 \end{cases}$$

Probabilistic Analysis

- Definition: Distribution on the random variable X.

You roll a die twice and the sum is $X = x_0$

e.g. $x_0 = 4$



$$\Pr(X = x) = \begin{cases} 1/36, & x = 2 \\ 2/36, & x = 3 \\ 3/36, & x = 4 \\ 4/36, & x = 5 \\ 5/36, & x = 6 \\ 6/36, & x = 7 \\ 5/36, & x = 8 \\ 4/36, & x = 9 \\ 3/36, & x = 10 \\ 2/36, & x = 11 \\ 1/36 & x = 12 \end{cases}$$

Probabilistic Analysis

- Definition: Expected value of a random variable.

  $$E[X] = \sum_x x \Pr\{X=x\}$$

  - The "average" of the values taken by the random variable.
  - What is the expected value of a roll of a dice?
    - $\sum_x x \Pr\{X=x\} = 1 * \frac{1}{6} + 2 * \frac{1}{6} + 3 * \frac{1}{6} + 4 * \frac{1}{6} + 5 * \frac{1}{6} + 6 * \frac{1}{6} = 3.5$
      - E(X) does not have to be one of the values taken by x.
  - What is the expected value of the sum of two rolls of a dice?
    - $\sum_x x \Pr\{X=x\} = 2 * \frac{1}{36} + 3 * \frac{2}{36} + 4 * \frac{3}{36} + \ldots + 12 * \frac{1}{36}$

Probabilistic Analysis

- Definition: Linearity of Expectation.

$$E[X+Y] = E[X]+E[Y], \text{ for all } X, Y$$

  - The expected value of the sum is the sum of the expected values.
  - Suppose we roll four dices. What is the expected value of the sum?
    - The sample space S has 6^4= 1296 outcomes, each with a probability $\frac{1}{1296}$
      - $E[X_1+X_2+X_3+X_4] = E[X_1]+E[X_2]+E[X_3]+E[X_4]$,
      - $= 3.5 + 3.5 + 3.5 + 3.5$
      - $= 14$

Probabilistic Analysis

- Definition: Indicator Random Variables.
- Indicator Random Variable for an event A of a sample space is defined as:

$$I\{A\} = \begin{cases} 1 & \text{if } A \text{ occurs,} \\ 0 & \text{if } A \text{ does not occur.} \end{cases}$$

  - A simple yet powerful technique for computing the expected value of a random variable.
  - Convenient method for converting between probabilities and expectations.
  - Helpful in situations in which there may be dependence.
  - Takes only 2 values, 1 and 0.

Probabilistic Analysis

- Definition: Indicator Random Variables.

Lemma 5.1

Given a sample space S and an event A in the sample space S, let $X_A = \mathrm{I}\{A\}$. Then $\mathrm{E}[X_A] = \Pr\{A\}$.

- the expected value of an indicator random variable associated with an event A is equal to the probability that A occurs.

Proof:

Let $\bar{A} = S - A$ (Complement of A)

Then,

$$\begin{aligned} \mathrm{E}[X_A] &= \mathrm{E}[\mathrm{I}\{A\}] \\ &= 1 \cdot \Pr\{A\} + 0 \cdot \Pr\{\bar{A}\} \\ &= \Pr\{A\} \end{aligned}$$

Probabilistic Analysis

- Definition: Indicator Random Variables.

Problem: Determine the expected number of heads in n coin flips.

Method 1: Without indicator random variables.

Let X be the random variable for the number of heads in n flips.

Then, $E[X] = \sum_{k=0..n} k \cdot \Pr\{X=k\}$. We can solve this with a lot of math.

$$
\begin{aligned}
E[X] &= \sum_{k=0}^{n} k \cdot \Pr\{X = k\} \\
&= \sum_{k=0}^{n} k \cdot b(k; n, p) \\
&= \sum_{k=1}^{n} k \binom{n}{k} p^k q^{n-k}
\end{aligned}
$$

$$
\begin{aligned}
&= np \sum_{k=1}^{n} \binom{n-1}{k-1} p^{k-1} q^{n-k} \\
&= np \sum_{k=0}^{n-1} \binom{n-1}{k} p^k q^{(n-1)-k} \\
&= np \sum_{k=0}^{n-1} b(k; n-1, p) \\
&= np
\end{aligned}
$$

Probabilistic Analysis

- Definition: Indicator Random Variables.

Problem: Determine the expected number of heads in n coin flips.

- **Method 2 :** Use Indicator Random Variables
  - Define n indicator random variables, X_i, $1 \le i \le n$.
  - Let X_i be the indicator random variable for the event that the i^{th} flip results in a Head.
  - $X_i = I\{\text{the } i^{th} \text{ flip results in } H\}$
  - Then $X = X_1 + X_2 + \ldots + X_n = \sum_{i=1..n} X_i$.
  - By Lemma 5.1, $E[X_i] = \Pr\{H\} = ½$, $1 \le i \le n$.
  - Expected number of heads is $E[X] = E[\sum_{i=1..n} X_i]$.
  - By linearity of expectation, $E[\sum_{i=1..n} X_i] = \sum_{i=1..n} E[X_i]$.
  - $E[X] = \sum_{i=1..n} E[X_i] = \sum_{i=1..n} ½ = n/2$.

2 different views

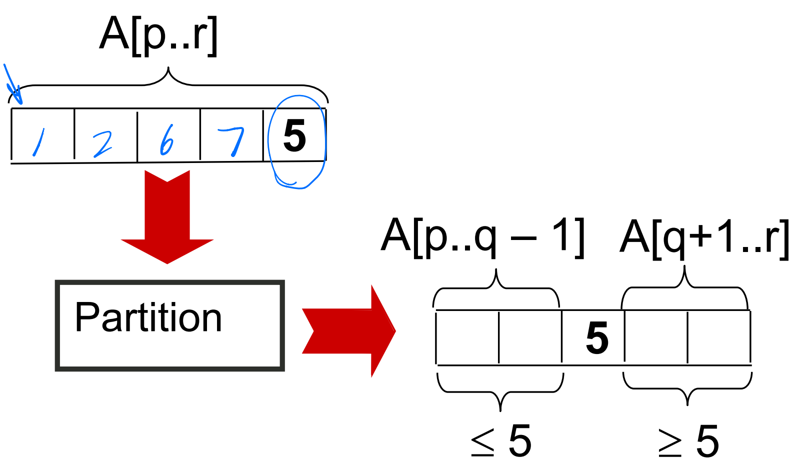
1. To consider the world as behaving randomly:
    - One can consider traditional algorithms that confront randomly generated input.
        - Average-case analysis => to study the behavior of an algorithm on an "average" input (subject to some underlying random process), rather than a worst-case input.
2. To consider algorithms that behave randomly:
    - Same worst-case input as always, but we allow our algorithm to make random decisions as it processes the input.
        - The role of randomization in this approach is purely internal to the algorithm and does not require new assumptions about the nature of the input.

QuickSort: Review

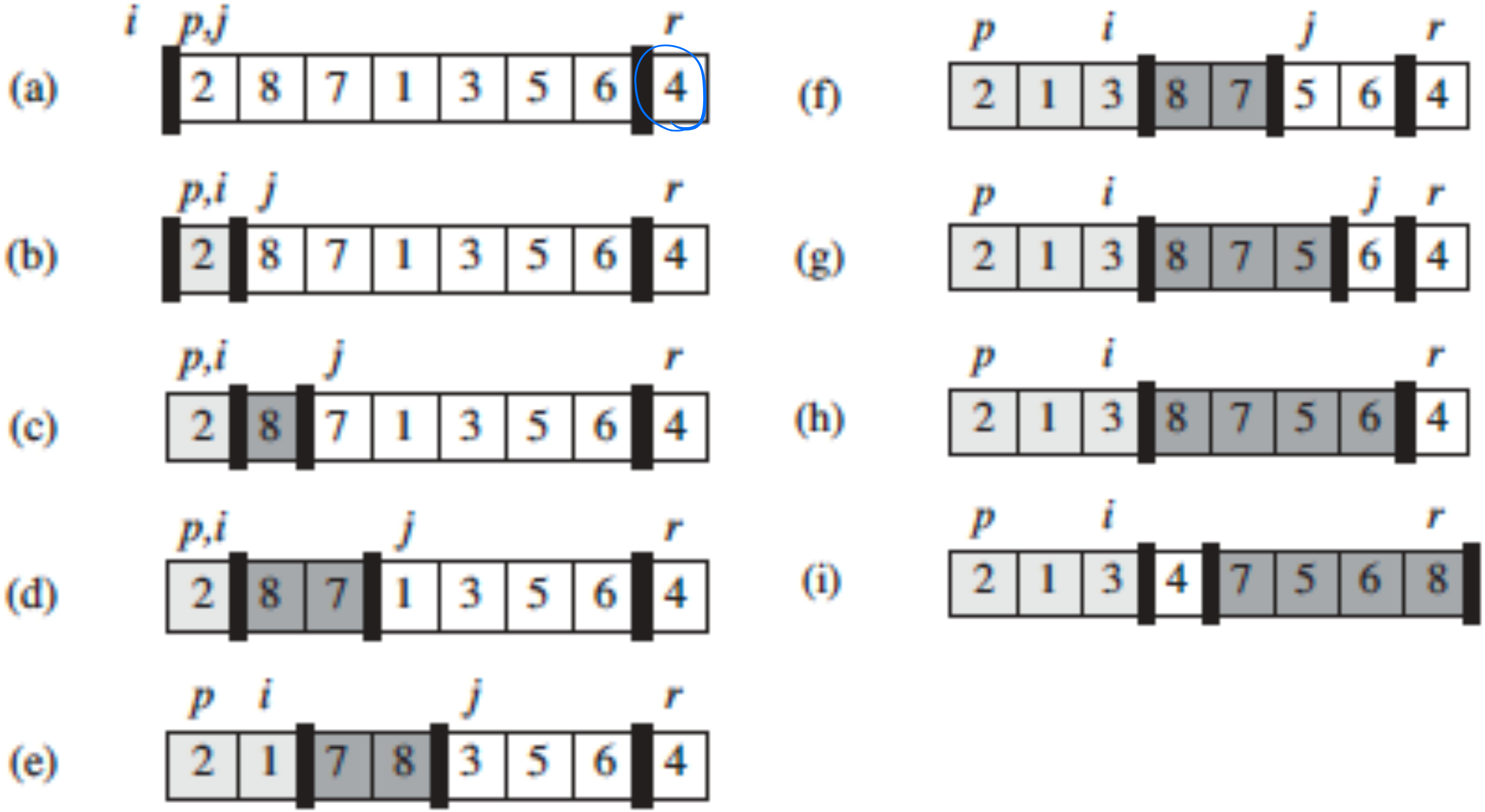
```
Quicksort(A, p, r)
    if p < r then
        q := Partition(A, p, r);
        Quicksort(A, p, q – 1);
        Quicksort(A, q + 1, r)
    fi
```

```
Partition(A, p, r)
    x, i := A[r], p – 1;
    for j := p to r – 1 do
        if A[j] ≤ x then
            i := i + 1;
            A[i] ↔ A[j]
        fi
    od;
    A[i + 1] ↔ A[r];
    return i + 1
```

left all elements less than x

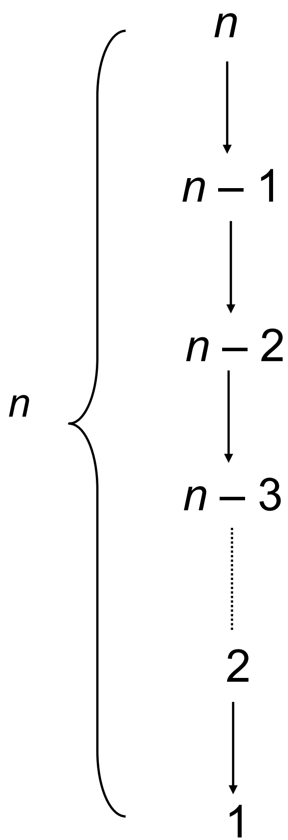


QuickSort: Review



QuickSort: Worst-case Partition

Recursion tree for
worst-case partition

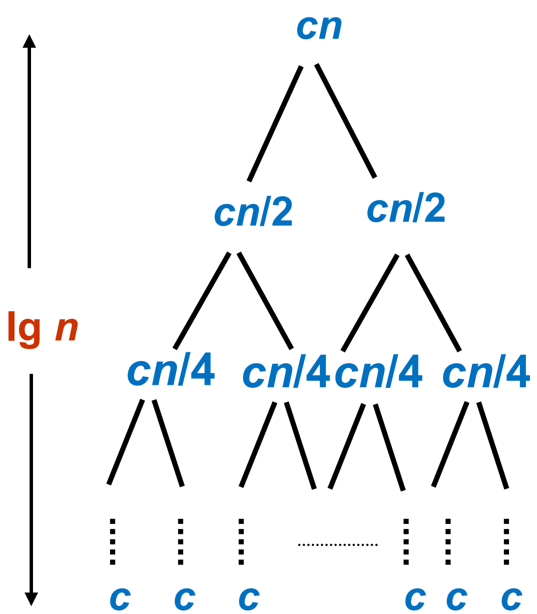


Split off a single element at each level:

$$
\begin{aligned}
T(n) &= T(n-1) + T(0) + \text{PartitionTime}(n) \\
&= T(n-1) + \Theta(n) \\
&= \textstyle\sum_{k=1 \text{ to } n} \Theta(k) \\
&= \Theta\left(\textstyle\sum_{k=1 \text{ to } n} k\right) \\
&= \Theta(n^2)
\end{aligned}
$$

The $\Theta(n^2)$ running time occurs when the input array is already completely sorted—a common situation in which insertion sort runs in O(n) time.

QuickSort: Best-case Partition



- Each subproblem size $\leq n/2$.
- Recurrence for running time
  - $T(n) \leq 2T(n/2) + \text{PartitionTime}(n)$
    $= 2T(n/2) + \Theta(n)$
- $\boldsymbol{T(n) = \Theta(n \lg n)}$

QuickSort: Unbalanced Partition Analysis

What happens if we get poorly-balanced partitions,

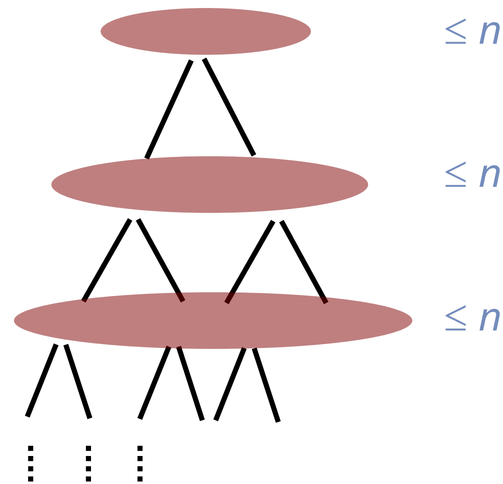
e.g., something like: $T(n) \le T(9n/10) + T(n/10) + \Theta(n)$?

Still get $\Theta(n \lg n)$!! (As long as the split is of constant proportionality.)

Intuition: Can divide n by $c > 1$ only $\Theta(\lg n)$ times before getting 1.

$$\left.\begin{array}{l} n \\ \downarrow \\ n/c \\ \downarrow \\ n/c^2 \\ \downarrow \\ \vdots \\ \downarrow \\ 1 = n/c^{\log_c n} \end{array}\right\}$$

Roughly $\log_c n$ levels;
Cost per level is $O(n)$.



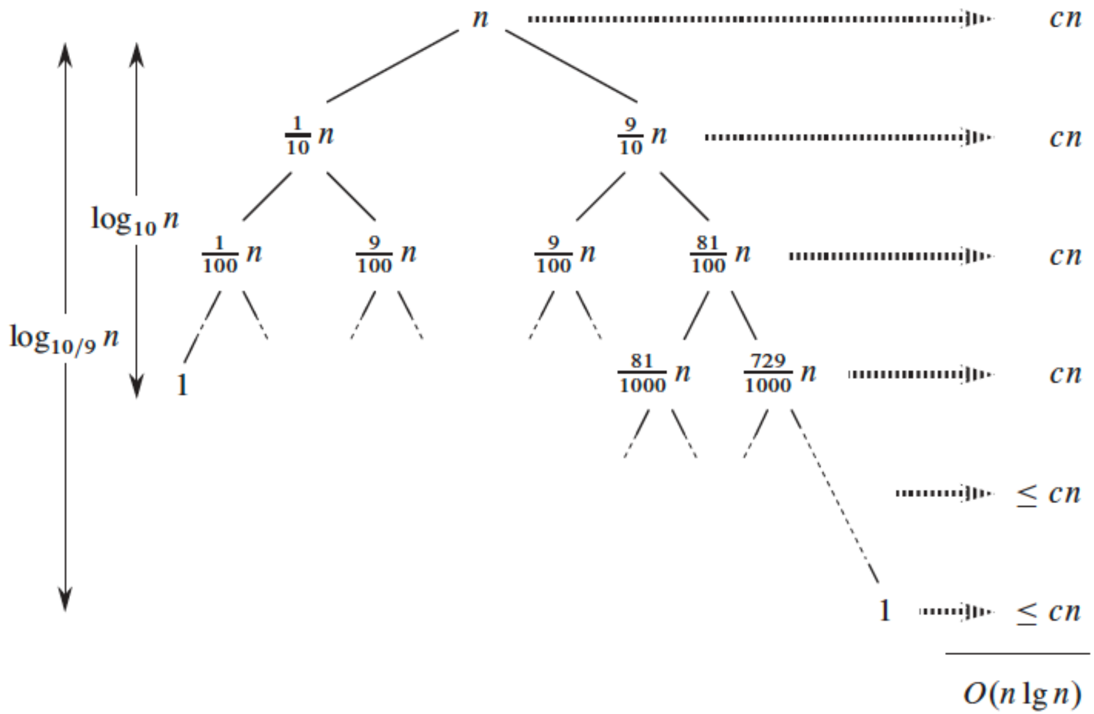
(**Remember:** Different base logs are related by a constant.)

QuickSort: Unbalanced Partition Analysis

What happens if we get poorly-balanced partitions,

e.g., something like: $T(n) \leq T(9n/10) + T(n/10) + \Theta(n)$?

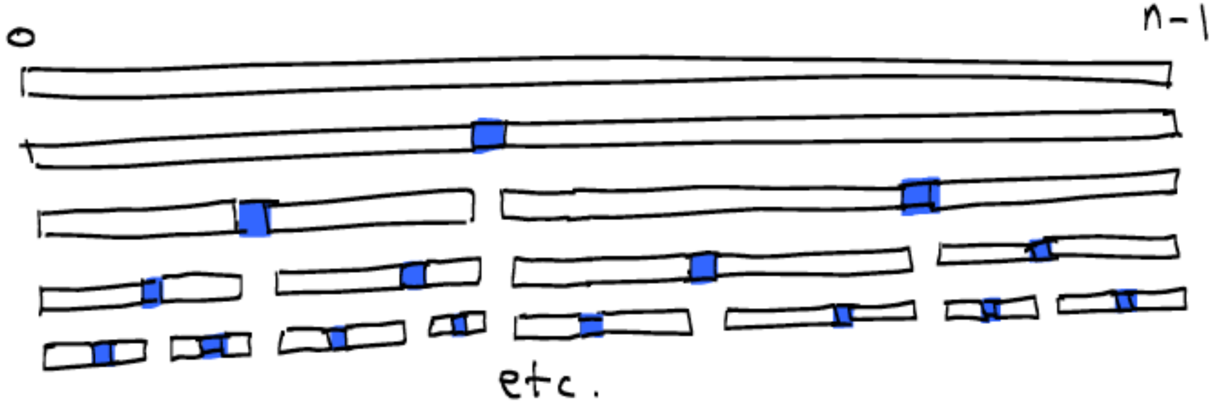
Still get $\Theta(n \lg n)$!! (As long as the split is of constant proportionality.)

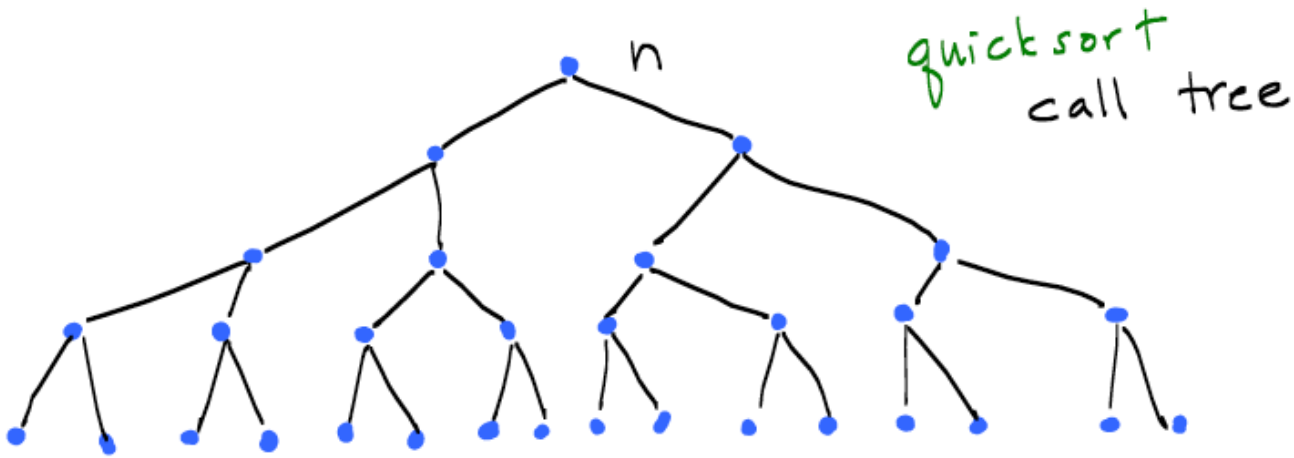


QuickSort: Average-case Analysis

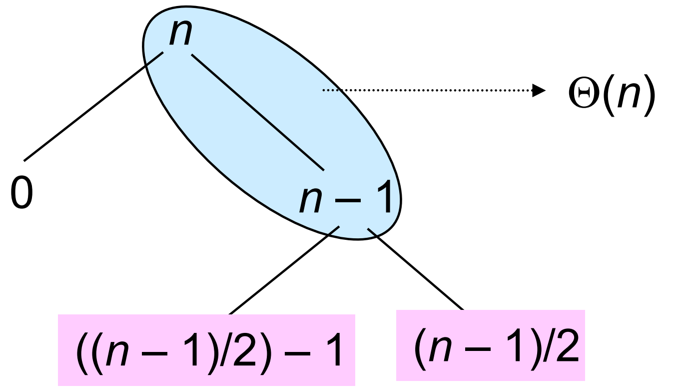
- The behavior of quicksort depends on the relative ordering of the values in the array elements given as the input, and not by the particular values in the array.
- On a random input array, the partitioning is unlikely to happen in the same way at every level.
  - Split ratio is different for different levels.
    (Contrary to our assumption in the previous slide.)
- Partition produces a mix of "good" and "bad" splits, distributed randomly in the recursion tree.
- What is the running time likely to be in such a case?

QuickSort: Average-case Analysis



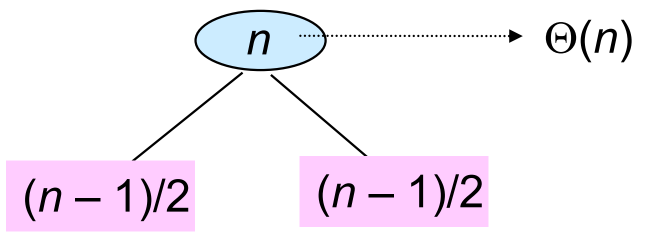


QuickSort: Average-case Analysis



Bad split followed by a good split:
Produces subarrays of sizes 0, $(n-1)/2-1$, and $(n-1)/2$.
Cost of partitioning :

$$\Theta(n) + \Theta(n\text{-}1) = \Theta(n).$$



Good split at the first level:
Produces two subarrays of size $(n-1)/2$.
Cost of partitioning :

$$\Theta(n).$$

Situation at the end of case 1 is not worse than that at the end of case 2.
When splits alternate between good and bad, the cost of bad split can be absorbed into the cost of good split.
Thus the running time is $O(n \lg n)$, though with larger hidden constants.

QuickSort: Randomized

- Want to make running time independent of input ordering.
- How can we do that?
  - » Make the algorithm randomized.
  - » Make every possible input equally likely.
    - Can randomly shuffle to permute the entire array.
    - For quicksort, it is sufficient if we can ensure that every element is equally likely to be the *pivot*.
    - So, we choose an element in $A[p..r]$ and exchange it with $A[r]$.
    - Because the *pivot* is randomly chosen, we expect the partitioning to be well balanced on average.

QuickSort: Randomized

Want to make running time independent of input ordering.

```
Randomized-Partition(A, p, r)
    i := Random(p, r);
    A[r] ↔ A[i];
    Partition(A, p, r)
```

```
Randomized-Quicksort(A, p, r)
    if p < r then
        q := Randomized-Partition(A, p, r);
        Randomized-Quicksort(A, p, q – 1);
        Randomized-Quicksort(A, q + 1, r)
    fi
```

Randomized QuickSort: Average-case Analysis

Random Variable X = # comparisons over all calls to Partition.

Q: Why is it a good measure?

Notation:

- Let $z_1, z_2, \ldots, z_n$ denote the list items (in sorted order).
- Let $Z_{ij} = \{z_i, z_{i+1}, \ldots, z_j\}$.

Let RV $X_{ij} = \begin{cases} 1 & \text{if } z_i \text{ is compared to } z_j \\ 0 & \text{otherwise} \end{cases}$

X_{ij} is an **indicator random variable**.

$X_{ij}=I\{z_i \text{ is compared to } z_j\}$.

Thus, $X = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} X_{ij}$.

Randomized QuickSort: Average-case Analysis

We have:

$$E[X] = E\left[\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} X_{ij}\right]$$

$$= \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} E[X_{ij}]$$

$$= \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} P[z_i \text{ is compared to } z_j]$$

Reminder:

$E[X_{ij}] = 0 \cdot P[X_{ij}=0] + 1 \cdot P[X_{ij}=1]$

$= P[X_{ij}=1]$

So, all we need to do is to compute P[z_i is compared to z_j].

Randomized QuickSort: Average-case Analysis

Claim: z_i and z_j are compared iff the first element to be chosen as a pivot from Z_{ij} is either z_i or z_j.

So,

$$
\begin{aligned}
P[z_i \text{ is compared to } z_j] &= P[z_i \text{ or } z_j \text{ is first pivot from } Z_{ij}] \\
&= P[z_i \text{ is first pivot from } Z_{ij}] \\
&\quad + P[z_j \text{ is first pivot from } Z_{ij}] \\
&= \boxed{\frac{1}{j-i+1} + \frac{1}{j-i+1}} \\
&= \frac{2}{j-i+1}
\end{aligned}
$$

We choose the pivot uniformly at random

Randomized QuickSort: Average-case Analysis

Therefore,

$$E[X] = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{2}{j-i+1}$$

$$= \sum_{i=1}^{n-1} \sum_{k=1}^{n-i} \frac{2}{k+1}$$

Substitute $k = j - i$.

$$< \sum_{i=1}^{n-1} \sum_{k=1}^{n} \frac{2}{k}$$

$$= \sum_{i=1}^{n-1} O(\lg n)$$

$$\sum_{k=1}^{n} \frac{1}{k} = H_n \quad (n^{\text{th}} \text{ Harmonic number})$$

$$H_n = \ln n + O(1)$$

$$= \boxed{O(n \lg n).}$$