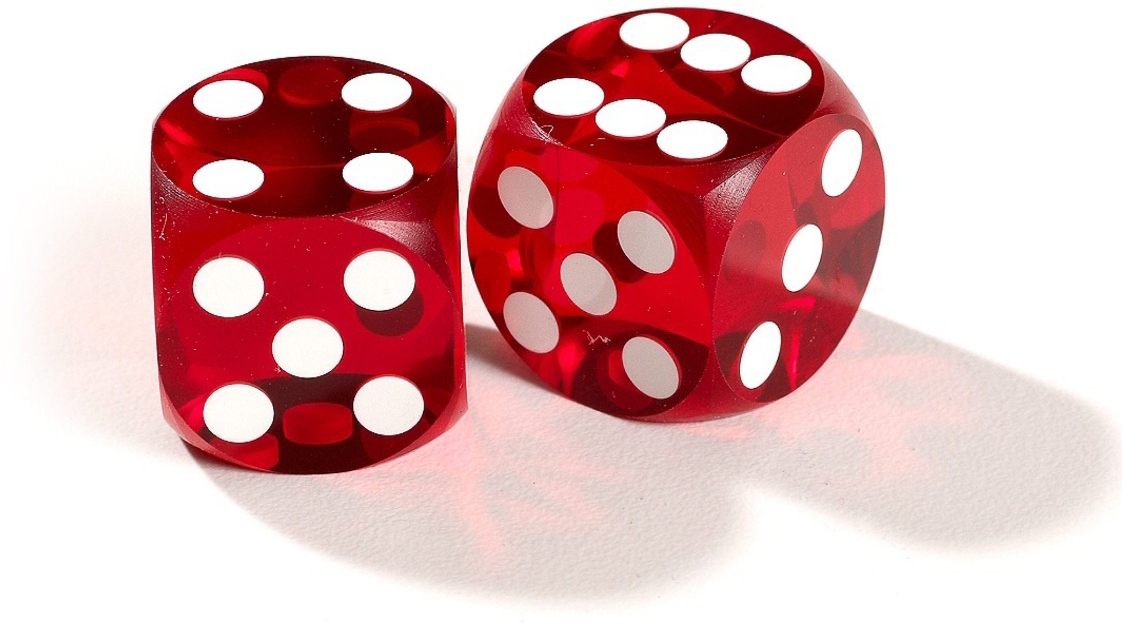
Data Structures and Algorithm Analysis (CSC317)



Randomized algorithms

Hiring problem

- We always want the best hire for a job!
- Using employment agency to send one candidate at a time
- Each day, we interview one candidate
- We must decide immediately if to hire candidate and if so, fire previous!

Hiring problem

- Always want the best hire…
- Cost to interview (low)
- Cost to fire/hire… (expensive)

Hire-Assistant(n)

```
1.     best = 0      //least qualified candidate
2.     for i = 1 to n
3.         interview candidate i
4.         if candidate i better than best
5.             best = i
6.             hire candidate i
```

Hiring problem

- Always want the best hire… fire if better candidate comes along…
- Cost to interview (low) c_i
- Cost to fire/hire… (expensive) c_h

$$c_h > c_i$$

n Total number candidates

m Total number hired

$O(?)$

Hiring problem

- Always want the best hire… fire if better candidate comes along…
- Cost to interview (low) c_i
- Cost to fire/hire… (expensive) c_h

$$c_h > c_i$$

n Total number candidates

m Total number hired

Different type of cost – not run time, but cost of hiring

Hire-Assistant(n)

```
best = 0      //least qualified candidate
for i = 1 to n
    interview candidate i
    if candidate i better than best
        best = i
        hire candidate i
```

Different type of cost, but flavor of max problems, which candidate is best/winning

Hiring problem

- Always want the best hire… fire if better candidate comes along…
- Cost to interview (low) c_i
- Cost to fire/hire… (expensive) c_h

$$c_h > c_i$$

n Total number candidates

m Total number hired

$$O(c_i n + c_h m)$$

Different type of cost – not run time, but cost of hiring

Hiring problem

- Always want the best hire… fire if better candidate comes along…
- Cost to interview (low) c_i
- Cost to fire/hire… (expensive) c_h

$$c_h > c_i$$

n Total number candidates

m Total number hired

When is this most expensive?

Hiring problem

- Always want the best hire… fire if better candidate comes along…
- Cost to interview (low) c_i
- Cost to fire/hire… (expensive) c_h

n Total number candidates

m Total number hired

When is this most expensive?
When candidates interview in reverse order, worst first…

Hiring problem

1. *best* = 0 //least qualified candidate
2. **for** i = worst_candidate **to** best_candidate…

$$O(c_i n + c_h n)$$

When is this most expensive?
When candidates interview in reverse order, worst first…

Hiring problem

1. *best* = 0 //least qualified candidate
2. **for** i = worst_candidate **to** best_candidate…

$$c_h > c_i$$

$$O(c_i n + c_h n) = O(c_h n)$$

When is this most expensive?
When candidates interview in reverse order, worst first…

Hiring problem

- Always want the best hire… fire if better candidate comes along…
- Cost to interview (low) c_i
- Cost to fire/hire… (expensive) c_h

n Total number candidates

m Total number hired

When is this least expensive?

Hiring problem

1. *best* = 0 //least qualified candidate
2. **for** i = best_candidate **to** worst_candidate…

$$O(c_i n + c_h)$$

When is this least expensive?
When candidates interview in order, best first…

Hiring problem

1. *best* = 0 //least qualified candidate
2. **for** i = best_candidate **to** worst_candidate…

$$O(c_i n + c_h)$$

When is this least expensive?
When candidates interview in order, best first…
But we don't know who is good or bad a priori…

Hiring problem

- Cost to interview (low c_i)
- Cost to fire/hire… (expensive c_h)

n Total number candidates

m Total number hired

$$O(c_i n + c_h m)$$

Depends on order of candidates!

Constant no matter order of candidates

Hiring problem and randomized algorithms

$$O(c_i n + c_h m)$$

Depends on order of candidates!

Constant no matter order of candidates

Randomized order can do us well **on average (we will show!)**

Hiring problem and randomized algorithms

$$O(c_i n + c_h m)$$

Depends on order of candidates!

- Can't assume in advance that candidates are in random order – there might be bias
- But we can add randomization to our algorithm – by asking the agency to send names in advance, and randomizing

Hiring problem - randomized

- We always want the best hire for a job!
- Employment agency sends list of n candidates in advance
- Each day, we choose randomly a candidate from the list to interview
- We do not rely on the agency to send us randomly, but rather take control in algorithm

Randomized hiring problem(n)

1. randomly permute the list of candidates
2. Hire-Assistant(n)

Including random number generator

What do we mean by randomly permute?

Including random number generator

Random(a,b)

Function that returns an integer between a and b, inclusive, where each integer has equal probability

Most programs: pseudorandom-number generator

Including random number generator

Random(a,b)

Function that returns an integer between a and b, inclusive, where each integer has equal probability

Random(0,1)?

Including random number generator

Random(a,b)

Function that returns an integer between a and b, inclusive, where each integer has equal probability

Random(0,1)?

0 with prob .5
1 with prob .5

Including random number generator

Random(a,b)

Function that returns an integer between a and b,
Inclusive, where each integer has equal probability

Random(3,7)?

3 with prob 1/5
4 with prob 1/5
5 with prob 1/5
…
Like rolling a (7-3+1) dice

How do we randomly permute?

Random permutation

Randomize-in-place(A,n)

1. For i = 1 to n
2. swap A[i] with A[Random(i,n)]

i

Random choice from A[i..n]

A = 1 2 3 4 5 6

A = 3 2 1 4 5 6

Random permutation

Randomize-in-place(A,n)

1. For i = 1 to n
2. swap A[i] with A[Random(i,n)]

i

Random choice from A[i..n]

A = 1 2 3 4 5 6

A = 3 2 1 4 5 6

A = 3 6 1 4 5 2

A = 3 6 1 4 5 2

….

Probability review

- Sample space S: space of all possible outcomes (e.g., that can happen in a coin toss)

- Probability of each outcome: p(i)>=0

$$\sum_{i \in S} p(i) = 1$$

- Example: coin toss $S = \{H, T)$

$$p(H) = p(T) = \frac{1}{2}$$

Probability review

- Event: a subset of all possible outcomes that can happen; subset of sample space S

$$E \subseteq S$$

Probability of event:

$$\sum_{i \in E} p(i)$$

- Example: coin toss that gave Heads E={H}

Probability review

- Event: a subset of all possible outcomes that can happen; subset of sample space

Example: rolling two dice

- List all possible outcomes
- List the event or subset of outcomes for which the sum of the dice is 7
- Probability of event that sum of dice 7

Probability review

- Event: a subset of all possible outcomes that can happen; subset of sample space

Example: rolling two dice

- List all possible outcomes
  S={(1,1},(1,2),(1,3), …(2,1),(2,2),(2,3), …(6,6)}

- List the event or subset of outcomes for which the sum of the dice is 7
  E = {(1,6),(6,1),(2,5),(5,2),(3,4),(4,3)}

- Probability of event that sum of dice 7: 6/36

Probability review

- Random variable: attaches value to an outcome/s

Example: value of one dice

Example: sum of two dice

Probability review

- Random variable: attaches value to an outcome/s

CAPITAL X : typically capital letter to denote random variable

small x : typically lower case for particular value/s that random variable takes

$p(X = x)$ Probability that random variable X takes on value x

Probability review

- Random variable: attaches value to an outcome/s

Example: r.v. X represents sum of two dice

$p(X = x)$ Probability that sum of two dice takes on value x

$$p(X = 7) = \frac{6}{36}$$

Probability review

- Expectation or Expected Value of Random variable:

= Average

Probability review

- Expectation or Expected Value of Random variable:

= Average

$$E[X] = \sum_x p(X = x)x$$

Sum over each possible value, weighted by probability of that value

Probability review

- Expectation or Expected Value of Random variable:

$$E[X] = \sum_x p(X = x)x$$

Example: X represents sum of two dice

$$E[X] = ?$$

Probability review

- Expectation or Expected Value of Random variable:

$$E[X] = \sum_x p(X = x)x$$

Example: X represents sum of two dice

$$E[X] = ?$$

Probability review

- Expectation or Expected Value of Random variable:

Example: X represents sum of two dice

Sum 2: {(1,1)}; 3:{(1,2);(2,1)); 4:{(1,3),(3,1),(2,2)};
5:{(1,4),(4,1),(2,3),(3,2)}; 6:{(1,5),(5,1),(4,2),(2,4),
(3,3)}; 7:{(1,6);(6,1);(5,2);(2,5);(3,4);(4,3)};
8:{(2,6);{6,2),{5,3},{3,5},{4,4}}; 9:{(3,6},{6,3},{5,4},{4,5});
10:{(6,4);(4,6);(5,5)};11:{(6,5);(5,6)}; 12:{(6,6)}

$$E[X] = 2\frac{1}{36} + 3\frac{2}{36} + 4\frac{3}{36} + 5\frac{4}{36} + 6\frac{5}{36}$$

$$+7\frac{6}{36} + 8\frac{5}{36} + 9\frac{4}{36} + 10\frac{3}{36} + 11\frac{2}{36} + 12\frac{1}{36} = \frac{252}{36} = 7$$

Probability review

- Expectation or Expected Value of Random variable:

Example: X represents sum of two dice

$$E[X] = 2\frac{1}{36} + 3\frac{2}{36} + 4\frac{3}{36} + 5\frac{4}{36} + 6\frac{5}{36}$$

$$+7\frac{6}{36} + 8\frac{5}{36} + 9\frac{4}{36} + 10\frac{3}{36} + 11\frac{2}{36} + 12\frac{1}{36} = \frac{252}{36} = 7$$

Easier way?

Probability review

- Linearity of Expectation

$X_1,..X_n$ random variables then:

$$E[\sum_{i=1}^{n} X_i] = \sum_{i=1}^{n} E[X_i]$$

Does not require independence of random variables

Probability review

- Linearity of Expectation

$$E[\sum_{i=1}^{n} X_i] = \sum_{i=1}^{n} E[X_i]$$

Example: X1, X2 represent the value of each dice

$$E[X_1] = E[X_2] = 1\frac{1}{6} + 2\frac{1}{6} + 3\frac{1}{6} + 4\frac{1}{6} + 5\frac{1}{6} + 6\frac{1}{6} = \frac{21}{6}$$

$$E[X_1 + X_2] = E[X_1] + E[X_2] = \frac{42}{6} = 7$$

Probability review

- Indicator Random Variable (indicating if occurs…)

$X_A = I\{A\} =$ 1 if A occurs
0 if A does not occur

Lemma: For event A

$$E[X_A] = \Pr\{A\}$$

Pr = probability

$$E[X_A] = 1\Pr(A) + 0(1 - \Pr(A)) = \Pr(A)$$

Probability review

- Indicator Random Variable

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

Example: Expected number Heads when flipping fair coin

$$X_H = I\{H\}$$

$$E[X_H] = 0\frac{1}{2} + 1\frac{1}{2} = \frac{1}{2}$$

$$= \Pr(H)$$

Probability review

- Indicator Random Variable & Linearity of Expectation

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

$$X = \sum_{i=1}^{n} X_i$$

Then by linearity of expectation:

$$E[\sum_{i=1}^{n} X_i] = \sum_{i=1}^{n} E[X_i]$$

Probability review

- Indicator Random Variable & Linearity of Expectation

Example: Expected number Heads when flipping 3 fair coins

$$X_H = I\{H\} = \begin{cases} 1 \text{ if Heads occurs} \\ 0 \text{ if Heads does not occur} \end{cases}$$

$$E[3X_H] = 3E[X_H] = 3\frac{1}{2} = 1.5$$