CS3343 Analysis of Algorithms Fall 2019

Homework 3

Due 9/28/18 before 11:59pm (Central Time)

1. Probability (7 points)

For the following problem, clearly describe the sample space and the random variables you use. Be sure to justify where you get your expected values from.

Consider playing a game where you roll n fair six-sided dice. For every 1 or 6 you roll you win \$30, for rolling any other number you lose \$$3n$ (where n is the total number of dice rolled).

(1) First assume $n = 1$ (i.e., you only roll one six-sided die).
   (a) (1 point) Describe the sample space for this experiment.
   (b) (1 point) Describe a random variable which maps an outcome of this experiment to the winnings you receive.
   (c) (1 point) Compute the expected value of this random variable.

(2) Now assume $n = 7$ (i.e., you roll 7 six-sided dice).
   (a) (1 point) Describe the sample space for this experiment (you don't need to list the elements but describe what is contained in it).
   (b) (1 point) Describe a random variable which maps an outcome of this experiment to the winnings you receive. (Hint: Express your random variable as the sum of seven random variables.)
   (c) (1 point) Compute the expected value of this random variable using the linearity of expectation. Based on this would you play this game?

(3) (1 point) What is the largest value of n for which you would still want to play the game? Justify your answer.

2. Variants of quicksort (6 points)

Suppose company X has created 3 new variants of quicksort but, because they are unsure which is asymptotically best, they have hired you to analyze their runtimes. Compute the asymptotic runtimes of the variants and use this to justify which is best:

- Variant 1: Partitioning the array now takes $\Theta(n^{1.1})$ time but the array will always be divided perfectly in half.
- Variant 2: Partitioning the array now only takes $\Theta(\sqrt{n})$ time but all of the numbers except the pivot will be partitioned into a single array.
- Variant 3: Partitioning the array still takes $\Theta(n)$ time but the array will always be divided into a $\frac{n}{x}$ and $\frac{(x-1)n}{x}$ portion (where x is some constant value > 1).

Reminder: $\log n \in o(n^c)$ for all $c > 0$.

3. Expected Runtimes - OTPP (6 points)

Let's suppose you've been hired to help design a new game show called "One True Programming Pair". The participants will compete to see which pair of programmers are the best at cooperative coding.

Your job is to take an array of programmers and find a set of candidate pairs that will compete in the game show. Inspired by the hireAssistant algorithm you develop the following algorithm. It adds a pair to the candidate list if the pair is better than the best pair seen so far.

Algorithm 1 void findCandidatePairs(person $A[1 \dots n]$)

```
1: best = −1; //This will contain core of the best pair seen so far
2: i = 1;
3: while i ≤ n − 1 do
4:     j = i + 1;
5:     while j ≤ n do
6:         //Check if pair i,j is better than best pair seen so far
7:         if best < evalPair(A[i], A[j]) then
8:             best = evalPair(A[i], A[j]);
9:             Add the pair i, j to to list of candidate pairs
10:            //Note: a person is permitted to be in multiple candidate pairs
11:        end if
12:        j++;
13:    end while
14:    i++;
15: end while
```

(1) (1 point) What is the smallest number of times that line 9 will run for findCandidatePairs on an array A of n people?

(2) (1 point) What is the largest number of times that line 9 will run for findCandidatePairs on an array A of n people?

(3) (1 point) Let X_{ij} be a random variable which, given our array A, is 1 if the pair i and j will added to the candidate list (i.e. line 9 runs). Let's assume the probability that $X_{ij} = 1$ is $\frac{1}{ij}$ (see part 6 for thinking about the actual probability). Compute $E(X_{ij})$.

(4) (1 point) Let X be the random variable which, given our array A, which is the number of pairs added to findCandidatePairs. Define X using a nested sum of the random variable X_{ij}.

(5) (2 points) Compute $E(X)$.

Hint: $\sum_{x=s_1}^{f_1} \sum_{y=s_2}^{f_2} A_x B_y = \left(\sum_{x=s_1}^{f_1} A_x \right) * \left(\sum_{y=s_2}^{f_2} B_y \right)$ where $s_1 \leq f_1$, $s_2 \leq f_2$, and they are all integers. A_x is a function on x and is not related to y. Likewise B_y is a function on y and is not related to x.

(6) (0 points) **Challenge Question:** Assuming a uniform probability distribution, $P(X_{ij} = 1)$ is actually $\frac{2}{2n(i-1)-i(i+1)+2j}$. Try deriving this probability for yourself.

(1) Once best pair shows first.

(2) $\frac{n(n-1)}{2}$ = all pairs were added since each time got better than previous.

$\binom{n}{2}$ → n Choose 2

(3) $E(X_{ij}) = \frac{1}{ij} \cdot 1 + \cancel{\frac{1}{ij} \cdot 0}$

$= \frac{1}{ij}$

(4) $\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} X_{ij}$

(5) $E(X) = E\left(\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} X_{ij}\right) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} E(X_{ij})$

$= \sum_{i=1}^{n-1} \sum_{j=1}^{n} \frac{1}{ij}$

$= \left(\sum_{i=1}^{n-1} \frac{1}{i}\right)\left(\sum_{j=1}^{n} \frac{1}{j}\right)$

$= O(\log n) \cdot O(\log n)$

$= O(\log^2 n)$

(1) First assume n = 1 (i.e., you only roll one six-sided die).
(a) (1 point) Describe the sample space for this experiment.
(b) (1 point) Describe a random variable which maps an outcome of this experiment to the winnings you receive.
(c) (1 point) Compute the expected value of this random variable.

(a) Sample space would be $S = \{1, 2, 3, 4, 5, 6\}$

(b) we can assume a random variable to a x which maps an outcome of winnings which can either 1 or 6 on the dice

(c) possible outcome would be $\frac{2}{6} = \boxed{\frac{1}{3}}$

Now assume n = 7 (i.e., you roll 7 six-sided dice).
(1 point) Describe the sample space for this experiment (you don't need to list the elements t describe what is contained in it).
(1 point) Describe a random variable which maps an outcome of this experiment to the nnings you receive. (Hint: Express your random variable as the sum of seven random riables.)
(1 point) Compute the expected value of this random variable using e linearity of expectation. Based on this would you play this game?

(a) Sample space when n = 7 would be 6^7
6 sides 7 dice

(b) Random variable x is dice 1 or 6
possible value for x
$\{0, 1, 2, 3, 4, 5, 6, 7\}$

(c) expected value would be sum of random variable $\frac{\{0, 1, 2, 3, 4, 5, 6, 7\}}{7 \text{ \# of dice}}$

$\frac{28}{7} = 4$ $4 \times 30 = 120$ in winning

(3) (1 point) What is the largest value of n for which you would still want to play the game? Justify your answer.

we could play for as long as we want because the more dice the more chance of winning.

2. Variants of quicksort (6 points) Suppose company X has created 3 new variants of quicksort but, because they are unsure which is asymptotically best, they have hired you to analyze their runtimes. Compute the asymptotic runtimes of the variants and use this to justify which is best:

• Variant 1: Partitioning the array now takes Θ(n1.1) time but the array will always be divided perfectly in half.

• Variant 2: Partitioning the array now only takes Θ(√n) time but all of the numbers except the pivot will be partitioned into a single array.

√

• Variant 3: Partitioning the array still takes Θ(n) time but the array will always be divided into a n x and (x−1)n x value > 1). Reminder: log n ∈ o(nc) for all c > 0. portion (where x is some constant

variant 1: since array is divided in to 2 halfs it would be $2T(n/2)$ and since partitioning takes $O(n^{1.1})$ recurrence relation as follows

$$2T(n/2) + \Theta(n^{1.1})$$

using master theorem:

$a = 2 \quad b = 2 \quad k = 1.1$

$\log_2 2 = 1 \qquad 1 < 1.1$

so case 3: $n^{1.1} \in \Omega(n^{1+\varepsilon})$ $\varepsilon = 0.1$

$n^{1.1} \in \Omega(n^{1.1})$

reg. (and $af(n/b) \le cf(n)$

$af(n/2) = 2f(n/2) = 2n/2 = 2^{1.1}/2^{1.1}$

$\frac{2}{2^{1.1}} f(n) = cf(n) \qquad c = 2/2^{1.1}$

variant 2: one part of array has pivot and every other in 1 single array

$$T(n) = T(n-1) + O(n)$$

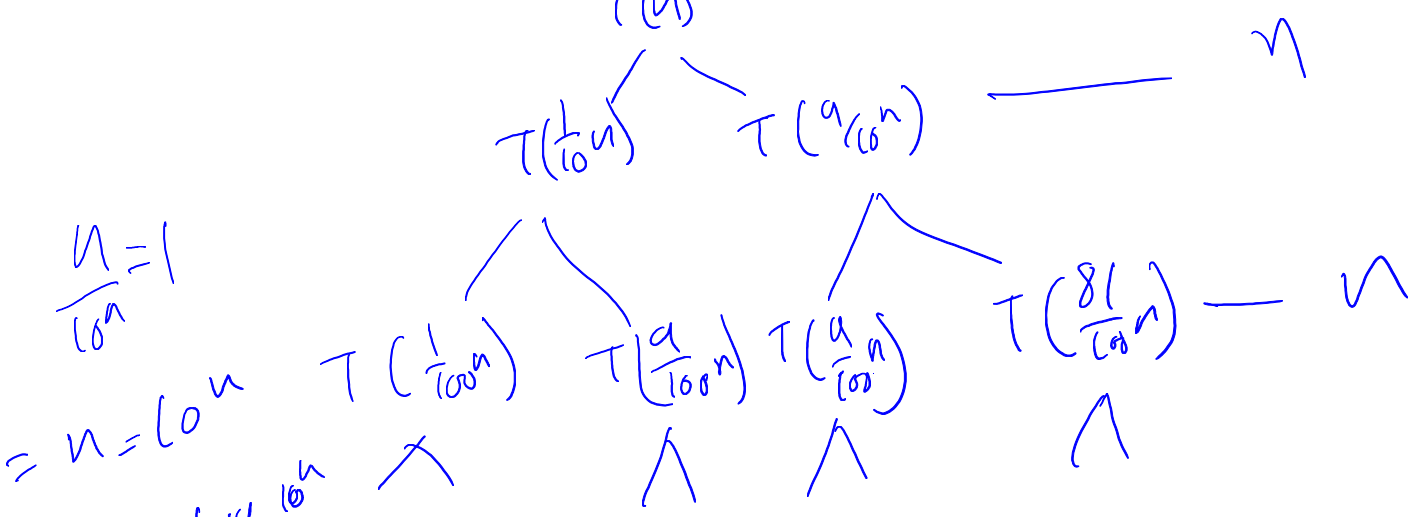
{

Variant 3: partition takes $O(n)$ and is divided into

$\frac{n}{x}$ and $\frac{(x-1)n}{x}$

assume $\frac{1}{10}$ and $\frac{9}{10}$

$$T(n) = \frac{1}{10}n + \frac{9}{10}n + \theta(n)$$



$\frac{n}{10^n} = 1$

$= n = 10^n$

$\log_{10} n = \log_{10} 10^n$

$n = \log_{10} n$

$T(1)$

$cn \log_{10} n \leq T(n) \leq cn \log_{10} n$

$T(n) \in \theta(n \log n)$

$\Rightarrow$ Best Variant

(1) (1 point) What is the smallest number of times that line 9 will run for findCandidatePairs on an array A of n people?
(2) (1 point) What is the largest number of times that line 9 will run for findCandidatePairs on an array A of n people?
(3) (1 point) Let Xij be a random variable which, given our array A, is 1 if the pair i and j will added to the candidate list (i.e. line 9 runs). Let's assume the probability that Xij = 1 is 1 the actual probability). Compute E(Xij).
ij (see part 6 for thinking about
(4) (1 point) Let X be the random variable which, given our array A, which is the number of pairs added to findCandidatePairs. Define X using a nested sum of the random variable Xij.
(5) (2 points) Compute E(X). Hint:
□f1 □f2 x=s1 y=s2 AxBy = □ □f1 x=s1 Ax □ * □ □f2 y=s2 By □ where s1 ≤ f1, s2 ≤ f2,
and they are all integers. Ax is a function on x and is not related to y.
Likewise By is a function on y and is not related to x.

(1) 1 because they could be the best pair and on one else is better than them.

(2) n times because every pair could be better than the i's best pair.

(3) assume $x_{ij} = 1$ is $\frac{1}{ij}$