HW5

Q1

Credit: Ganesh Ramani

1).

Let x be A random variable representing Number of customers that receive their own Hat.

Let x_i be Indicator random variable representing the event that ith customer receives his own hat.

We can say $x_i = \begin{cases} 1 & \text{Receives own Hat} \\ 0 & \text{Receives other Hat.} \end{cases}$

$X = x_1 + x_2 + --- + x_n$. ($x$ = Total customer receiving own Hat)

The probability of customer receiving own hat is $\frac{1}{n}$

n is the no of Hats

$P\{x_i = 1] = \frac{1}{n}$

So $E[x_i] = \frac{1}{n}$.

$$E[x] = E\left[\sum_{i=1}^{n} x_i\right]$$

$$= \sum_{i=1}^{n} \frac{1}{n}$$

$$= n \cdot \left(\frac{1}{n}\right)$$

$$= 1.$$

We Expect that only 1 customer will receive His Hat Back.

Q2

Credit: Sai Beethnabotla

Solution 2: Indicator Random Variables

Given that for $A[1 \dots n]$
if $i < j$, then $A[i] > A[j]$, then (i, j) is called inversion of A.

Let X_{ij} be an indicator random variable.

Probability that the first is greater than the second is $= 1/2$.

$$\Rightarrow P_r[X_{ij} = 1] = 1/2$$

By Lemma 5.1, $E[X_{ij}] = 1/2$

Expected number of inversions for 'n' elements

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

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

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

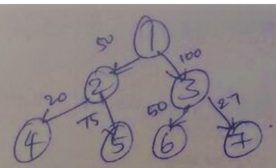
$$E[X] = \frac{1}{2} \sum_{i=1}^{n-1} (n-i)$$

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

$$E[X] = \frac{n(n-1)}{4}.$$

Q3 Credit: Navaneeth Chandrasekaran

3.



No. of Root – Leaf paths available

$$= 4 \begin{bmatrix} 1-2-4 \\ 1-2-5 \\ 1-3-6 \\ 1-3-7 \end{bmatrix}.$$

∴ Lets Add the leaf to path weights for each path and divide it by $(1/4)$ for the expectation.

$$E[\text{Weight path length, Leaf} \to \text{root}] = [70 + 125 + 150 + 127] \times$$

$$= 472 \times \frac{1}{4} = 118.$$

Expected weight path length to a leaf from the root is $\underline{\underline{118}}$.

Q4 Credit: Sokchea Peng

(4) a) By all elements are equal, then when Partition return $q = r$ and all elements in $A[p \dots q-1]$ are equal. We get recurrence $T(n) = T(n-1) + T(0) + \Theta(n)$ for the running time, and so $T(n) = \Theta(n^2)$

b) The Partition Procedure:

Partition (A, p, r)

$x = A[p]$

$i = h = p$

for $j = p+1$ to r

$A[h+1 \ldots j-1] > x$, $A[j \ldots r]$ unknown.

if $A[j] < x$

$y = A[j]$

$A[j] = A[h+1]$

$A[h+1] = A[i]$

$A[i] = y$

$i = i+1$

$h = h+1$

else if $A[j] = x$

exchange $A[h+1]$ with $A[j]$

$h = h+1$

return (i, h)

c) Randomize-Partition is the same as Randomized-Partition, but with the call to partition replaced by a call to Partition. Quicksort (A, p, r)

if $p < r$

$(q, t) =$ Random Partition (A, p, r)

Quicksort $(A, p, q-1)$

Quicksort $(A, t+1, r)$