Quiz-3

Graded

Student

Abhinav Shripad

Total Points

24 / 24 pts

Question 1

Counting alignments

12 / 12 pts

+ **2.4 pts** do not know how to approach this problem

✔ **+ 1 pt** Claiming the correct explicit answer (m+n
n)

APPROACH-1 Setting up recursive relation

✔ **+ 1 pt** Clearly defining the recursion variables

✔ **+ 1.5 pts** Correct Base Cases

✔ **+ 5 pts** Correct Recursive Relation

✔ **+ 3.5 pts** Brief justification for the recursive relation

APPROACH 2: Setting up a bijection

+ **2 pts** Defining convention to count equivalent alignments only once

+ **3 pts** Associating unique merged string with each alignment

+ **3 pts** Associating unique alignment with each merged string

+ **1.5 pts** Brief justification of associating unique merged string with each alignment

+ **1.5 pts** Brief justification of associating unique alignment with each merged string

+ **0 pts** wrong/no solution

Question 2

Probability

12 / 12 pts

+ 2.4 pts Written "I do not know how to approach this problem"

✔ **+ 1 pt** Ti,j denotes probability of obtaining exactly i heads from first j coins

---

Click here to replace this description.

✔ **+ 1 pt** Base Case 1: $T(0, j) = \prod_{k \leq j}(1 - p_k)$ where j belongs to 1 to n

✔ **+ 1 pt** Base Case 2 : $T(1, 1) = p_1$

✔ **+ 1 pt** Base Case 3: $T(i, j) = 0$ for i < j

✔ **+ 2 pts** Correct recursive case

✔ **+ 2 pts** Brief justification/proof for recursive case

✔ **+ 2 pts** Brief (but correct) O(n^2) algorithm description

✔ **+ 1 pt** Final answer = $\sum_{i=\lfloor \frac{n}{2} \rfloor + 1}^{n} T_{i,n}$

✔ **+ 1 pt** Brief justification of O(n^2) running time

+ 0 pts Incorrect

COL351: Analysis and Design of Algorithms
Quiz 3

Name: Abhinav Rajesh Shripad	Oct 16, 2024
Entry number: 2022CS11596	Total points: 24

Please write your answers within the box provided. Answers written outside the boxed region will not be graded.

Problem 1 [12 points]

Let X and Y be two strings of length n and m, respectively. How many distinct alignments between X and Y are there? Justify your answer.

Note: Your calculation should exclude alignments that match two gaps with each other. Additionally, if X = AB and Y = CD, then the alignments `(AB_,C_D)` and `(A_B,CD_)` are considered equivalent and should be counted as a single alignment.

Answer = ${}^{n+m}C_n = {}^{n+m}C_m$

let dp(n, m) denote valid solutions for string of length n and m.

where can character 1 of X map to/go?

(1) to __

(2) to character i for B

→ Proof of correctness

if (1) → dp(n−1, m)

if (2) → dp(n−1, m−i) for $1 \leq i \leq m$

because if A → B[i] then all the character before B[i] ie B[j] j<i are fixed to a dash. and we need to align A[1:] and B[i:] of length n−1 and m−i

Reccurence:

$$dp(n,m) = dp(n-1,m) + \sum_{i=0}^{m-1} dp(n-1,i)$$

$$= \sum_{i=0}^{m} dp(n-1,i) \quad \text{---} \quad \mathrm{I}$$

Claim:- $dp(n,m) = {}^{n+m}C_n$

~~Proof:- Induction of n+m~~

~~Base case n+m = 1, wlog n=1~~

~~A only alignment possible~~ → ~~A~~ ← X

— ← Y

~~1 ≥ 1, hence correct~~

~~Assume true for ∀m, n+m = 1, 2 ... k ≥ 1~~

~~∀n ≤ k-1 and n+m = 1, 2, 3 ... k~~

for $n+m=k$

$$dp(n,m) = \sum_{i=0}^{m} dp(n-1,i)$$

$$= \sum_{i=0}^{m} \binom{n-1+i}{i}$$

(Inductive hypothesis because $n-1+i \leq n+m$)

Name: Abhinav Rajesh Shripad	Oct 16, 2024
Entry number: 2022CS11596	Total points: 24

~~By hockey-stick Identity~~

Replace $m \to m-1$ in (I) to get

$$dp(n, m-1) = \sum_{i=0}^{m-1} dp(n-1, i) \quad \cdots \text{(II)}$$

(I) - (II) $\to$ ~~dp(n,m-1)~~ gives

$$dp(n,m) - dp(n,m-1) = \sum_{i=0}^{m} dp(n-1,i) - \sum_{i=0}^{m-1} dp(n-1,i)$$

$$= dp(n-1,m)$$

$$dp(n,m) = dp(n,m-1) + dp(n-1,m) \quad \cdots \text{(III)}$$

Proof of $dp(n,m) = {}^{n+m}C_n$

~~Induction of n+m~~. Strong induction on $n+m$

Base Case :- $n+m=1$, wlog $n=1$, Trivially 1 as only 1 alignment possible $\begin{matrix} A \leftarrow X \\ - \leftarrow Y \end{matrix}$

for $n+m = 1, 2, 3 \cdots k-1$ assume true

$$\to dp(n,m) = dp(n-1,m) + dp(n,m-1)$$

$$= \binom{n+m-1}{n-1} + \binom{n+m-1}{n}$$

$$= \binom{n+m}{n} \quad \text{(Pascal's triangle Identity)}$$

Hence Proved

Problem 2 [12 points]

Given an integer n and nonnegative numbers $p_1, \dots, p_n \in [0,1]$, you want to determine the probability of obtaining strictly more heads than tails when n biased coins are tossed independently at random, where p_i is the probability that the i^{th} coin comes up heads. Give an $\mathcal{O}(n^2)$ algorithm for this task. Assume you can multiply and add two numbers in $[0,1]$ in $\mathcal{O}(1)$ time. Justify the correctness and running time of your algorithm.

Algorithm:-

make $(n+1) \times (n+1)$ dp table, $dp(i,j)$

$dp(i,j)$ = probability to get exactly j heads out of first i tosses of coin with probabilities $p_1, p_2, \dots p_i$

Base Case:- $dp(0,j) = 0 \quad j > 0$

$dp(0,0) = 1$

Recurrence Relation:

$$dp(i,j) = (1-p_i)\,dp(i-1,j) + p_i\, dp(i-1,j-1) \quad \text{---} (1) \quad i \geq 1,\ j > 0$$

$$dp(i,0) = dp(i-1,0)\,(1-p_i) \cdots (2) \quad i \geq 1$$

~~Answer = ...~~ $\boxed{\text{Answer} = \sum_{2n \geq 2j > n} dp(n,j)}$

Order of computation

Increasing order of $i \to 1$ to n

Increasing order of $j \to 0$ to n

Name: Abhinav Rajesh Shripad	Oct 16, 2024
Entry number: 2022CS11596	Total points: 24

Pseudocode :-

Input :- P (array), n size

Output :- probability to get more heads than tails

```
dp [n+1] [n+1]  # Initialization
~~dp [0] [i] = 0 for~~
for i in range (1, n):
        dp (0, i) = 0

dp(0,0) = 0

for i in range(1,n): ------ (I)
        dp(i,0) = (1- p[i]) * dp (i-1,0)
        for j in range (1,n) :- - - - (II)
             dp(i,j) = (1-p[i]) dp(i-1,j) +
                          p[i] * dp[i-1,j-1] --- (III)

answer = 0
for j in range (1,n):
      if   2j > n:
             answer += dp[n,j]

return answer.
```

$T.C. = \text{I} \times \text{II} \times \text{III}$ 2 transitions per entry

$= O(n) \times O(n) \times O(2) = O(n^2)$

$S.C. = O(\text{dp table}) = O(n^2)$

Proof of correctness :- dp(i, j) correctly computes ~~by by tad~~ probablity of j heads in first i tosses.

by induction on i :- base case i = 0 true by initialcondn.

for general i, if ith toss is head, need j-1 head in first i-1 tosses, if ith toss is tail, need j heads. By induction hypothesis dp(i-1, j) and dp(i-1, j-1) is correct.

→ dp(i, j) satisfies the reccurence made.

~~to Note use cas~~ For more heads than tails, we need j heads > (n − j) tails

→ 2j > n thus we add dp(n, j) where 2j > n.