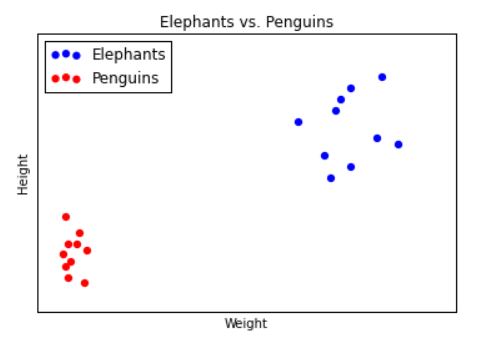
1. **(20 points) K-means clustering Give an example of k-means clustering that has a clearly non-optimal scoring configuration (in comparison to an obviously better scoring one), where the Lloyd algorithm produces no improvement.**

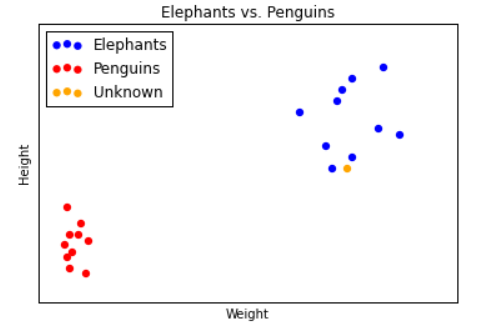
**Answer:**

Below shows the data comparison between Elephants and Penguins. Twenty animals has been plotted. Blue shows elephants and red plot shoes Penguins, they are represented by their weight and height.

Now as we can see that the coordinate points of both the elephants form two clusters, i.e., elephants are bigger and heavier whereas penguin are smaller and lighter. Now we got a new data point, but we don’t know whether it’s an elephant or a penguin.



**We have plotted it as orange below.**



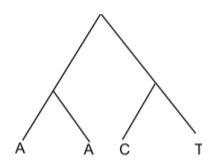
If we were to take guess, then we would say that orange datapoint belongs to an elephant and not to a penguin. We say that because the orange datapoint seems to be near to elephant’s cluster.

This is the essence of clustering. Some labeled data is taken for example height and weight, where we label each animal as elephant or penguin. Next we use algorithm to figure out which datapoints belongs to which cluster based on their height and weight. We look and understand what label each cluster corresponds to. Then we take orange datapoints and check where it best fits and assign a label to it.

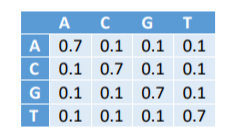
These processes we call it as k-means clustering because we think that there are k clusters and each cluster is defined by its mean or center point. To find out the cluster we will use Lloyd’s algorithm, first step we start with k random centroid. Next step we find the datapoints that are closer to that centroid than to any other centroid, we call that datapoints as its cluster. Next we take the mean of that cluster and name it as new cluster. This process we repeat until algorithm stops moving the centroids. We do this in order to minimize the total sum of distances from every centroid to the points in its cluster — that is our metric for how well the clusters split up the data. We say that algorithm is met when the centroid stops moving but here Lloyd’s algorithm gives us local optimum and it does not ensures with the global optimum. This is critical pitfall. Due to these different runs on kmeans algorithm with same setting will end up giving different output.

1. **(20 points) Likelihood of a Phylogenetic Tree**

**Given the following tree structure T and leaf node bases.**



**Assuming the following constant mutation probably matrix and root node has equal prior probability for the bases, compute the overall likelihood of the tree.**

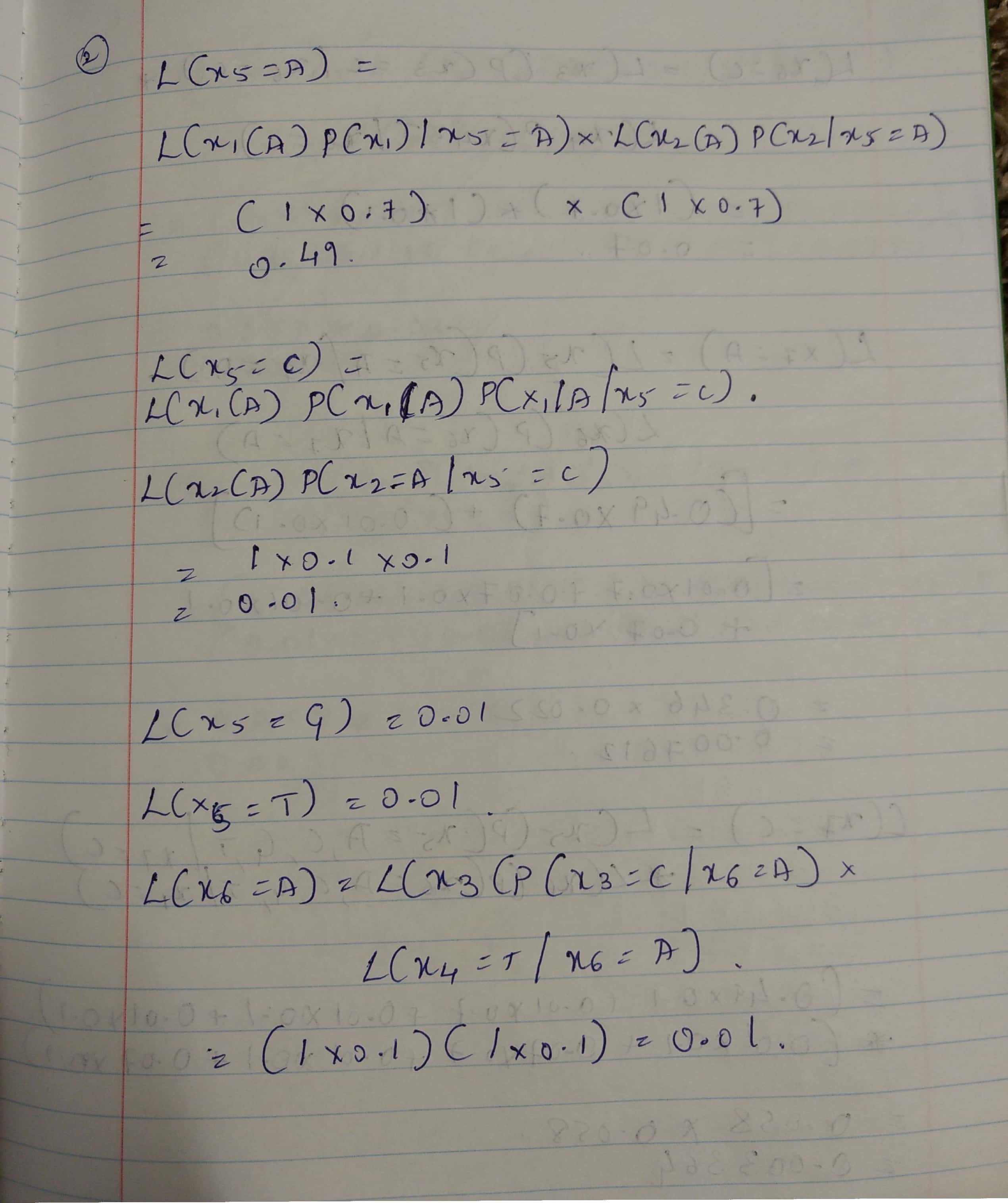


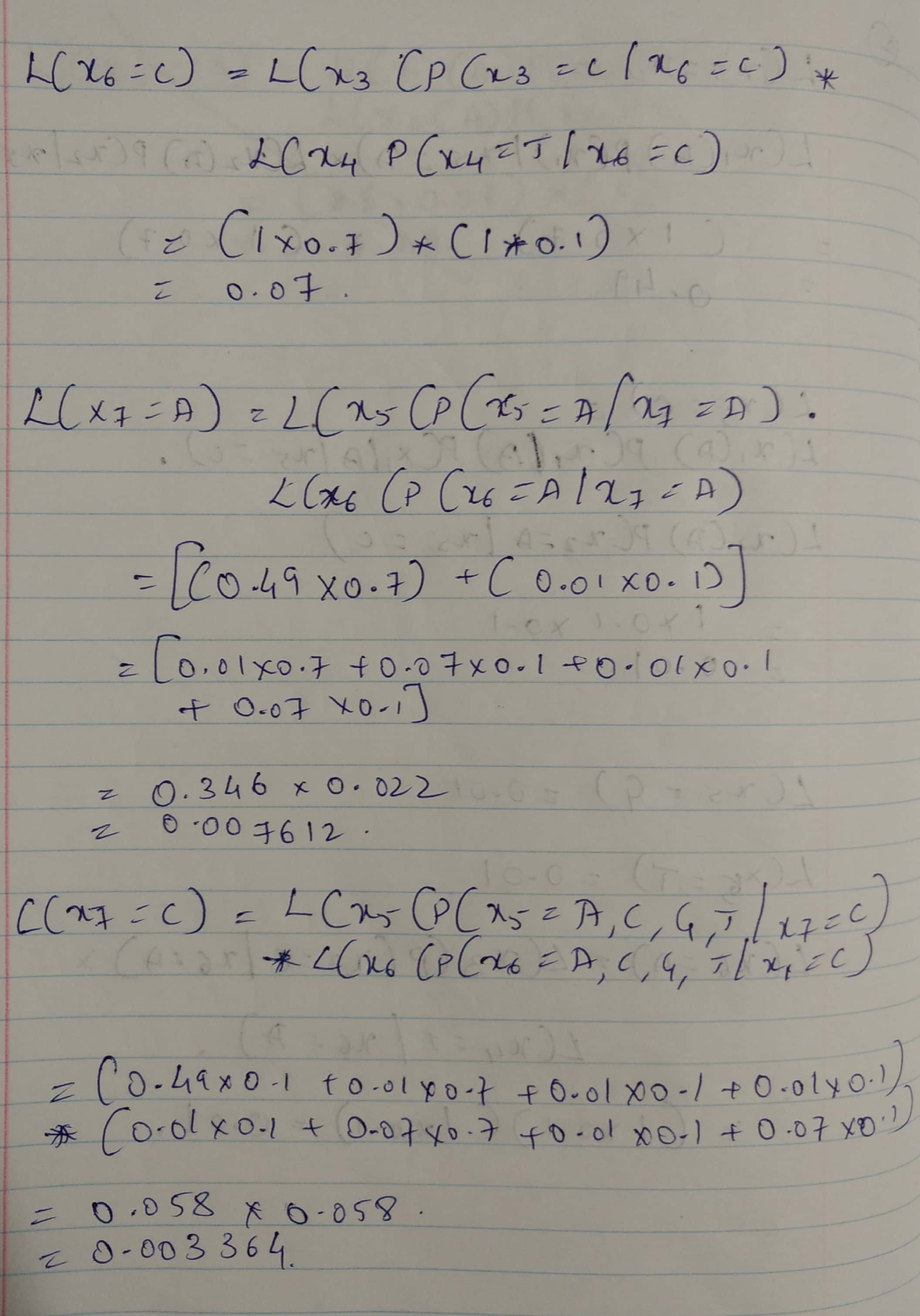
**Answer:**

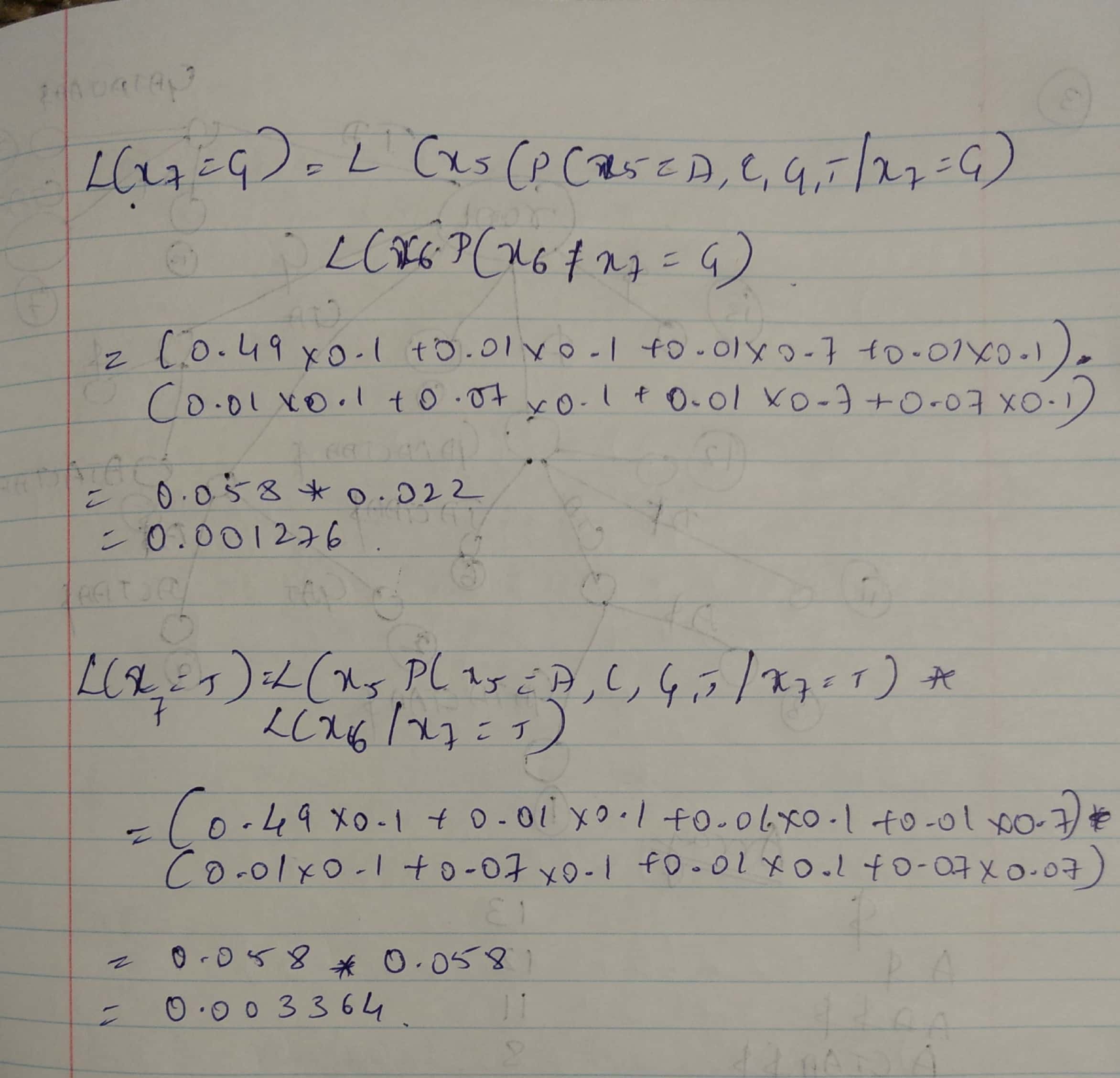
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **A** | **C** | **G** | **T** |
| **L(x1)** | 1 | 0 | 0 | 0 |
| **L(x2)** | 1 | 0 | 0 | 0 |
| **L(x3)** | 0 | 1 | 0 | 0 |
| **L(x4)** | 0 | 0 | 0 | 1 |
| **L(x5)** | 0.49 | 0.01 | 0.01 | 0.01 |
| **L(x6)** | 0.01 | 0.07 | 0.01 | 0.07 |
| **L(x7)** | 0.007612 | 0.003364 | 0.001276 | 0.003364 |

Overall Likelihood = 0.25(0.007612+0.003364+0.001276+0.003364)

= 0.003904



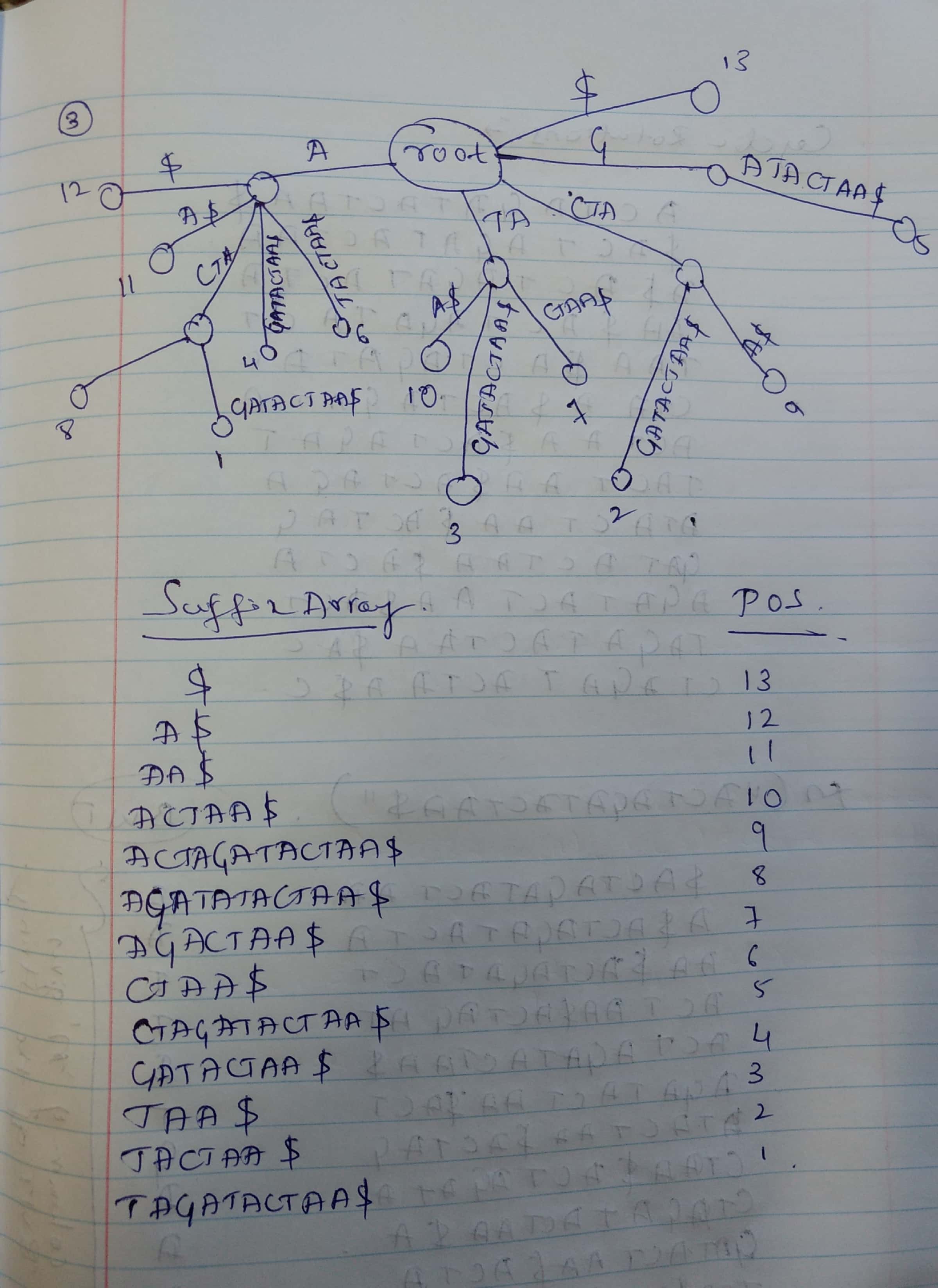


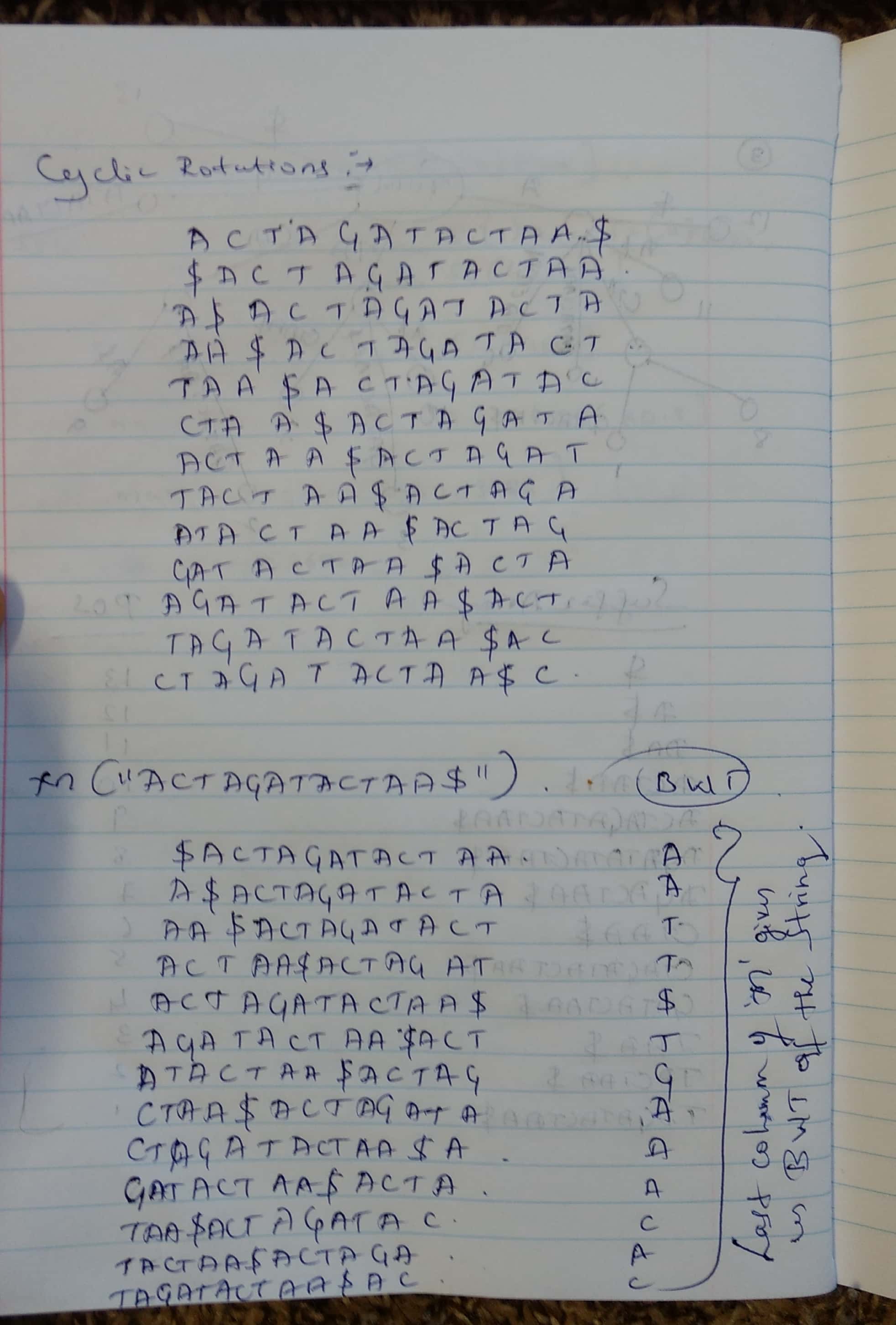


1. **(20 points) Suffix Tree and Array Construct the explicit suffix tree of string ‘ACTAGATACTAA$’. Convert it into a suffix array and find the BurrowsWheeler transform of the string**

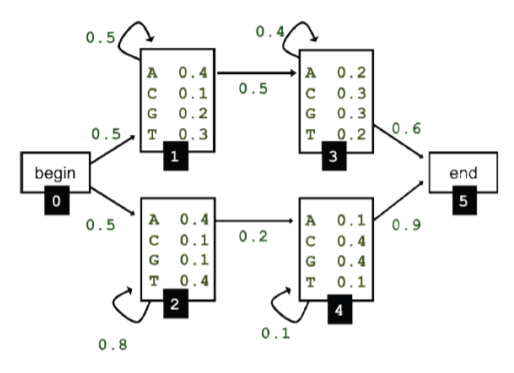
**Answer:**

1. **ACTAGATACTAA$**
2. **CTAGATACTAA$**
3. **TAGATACTAA$**
4. **AGATACTAA$**
5. **GATACTAA$**
6. **ATACTAA$**
7. **TACTAA$**
8. **ACTAA$**
9. **CTAA$**
10. **TAA$**
11. **AA$**
12. **A$**
13. **$**





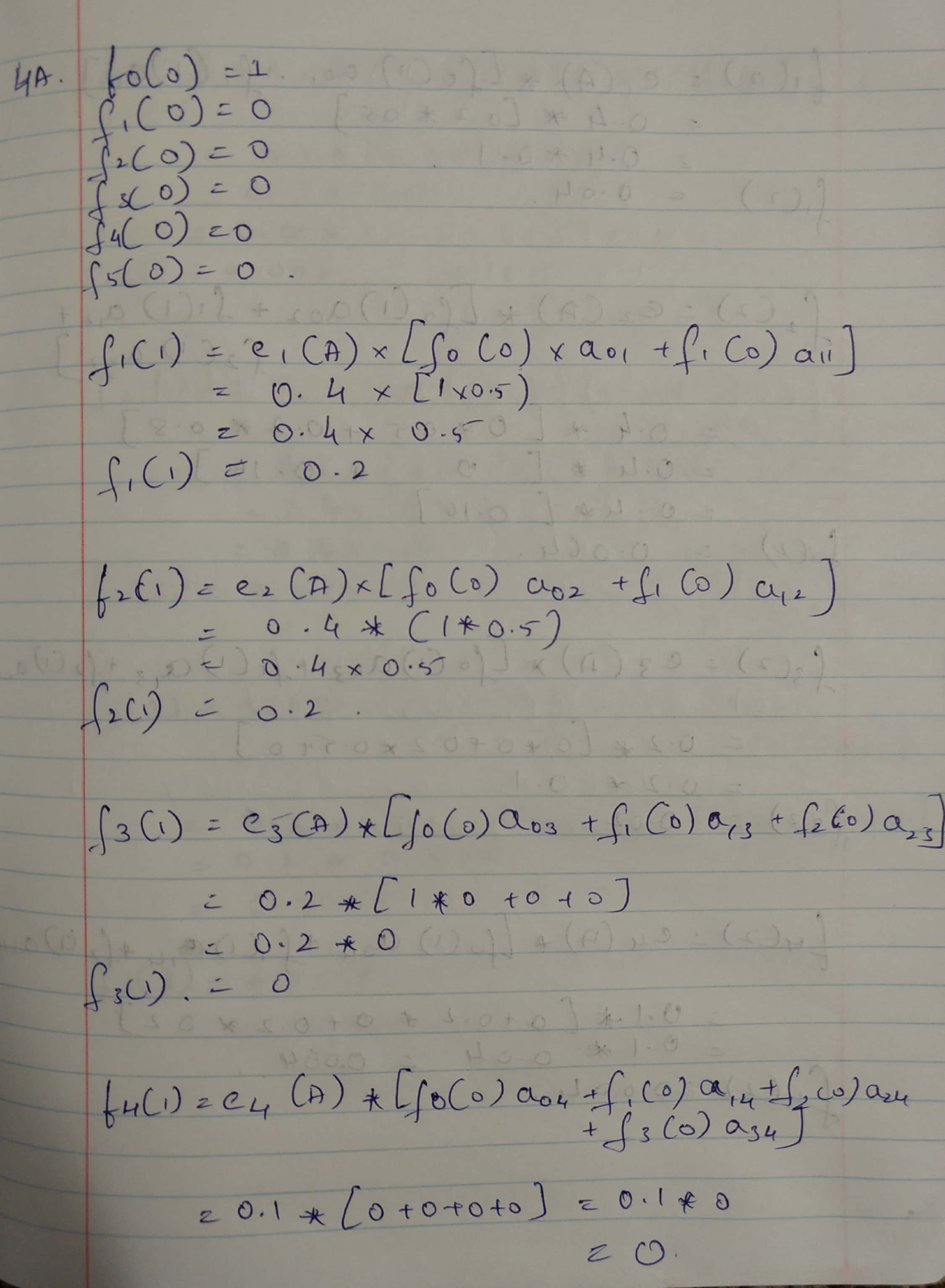
1. (20 points) Inference in Hidden Markov models You are given the following Hidden Markov model with four states with emission probabilities and two silent states, one at the beginning and one at the end. Each state emits one of the four symbols, {A,T,G,C} with the specified emission probabilities and transitions to another state with specified transition probabilities.

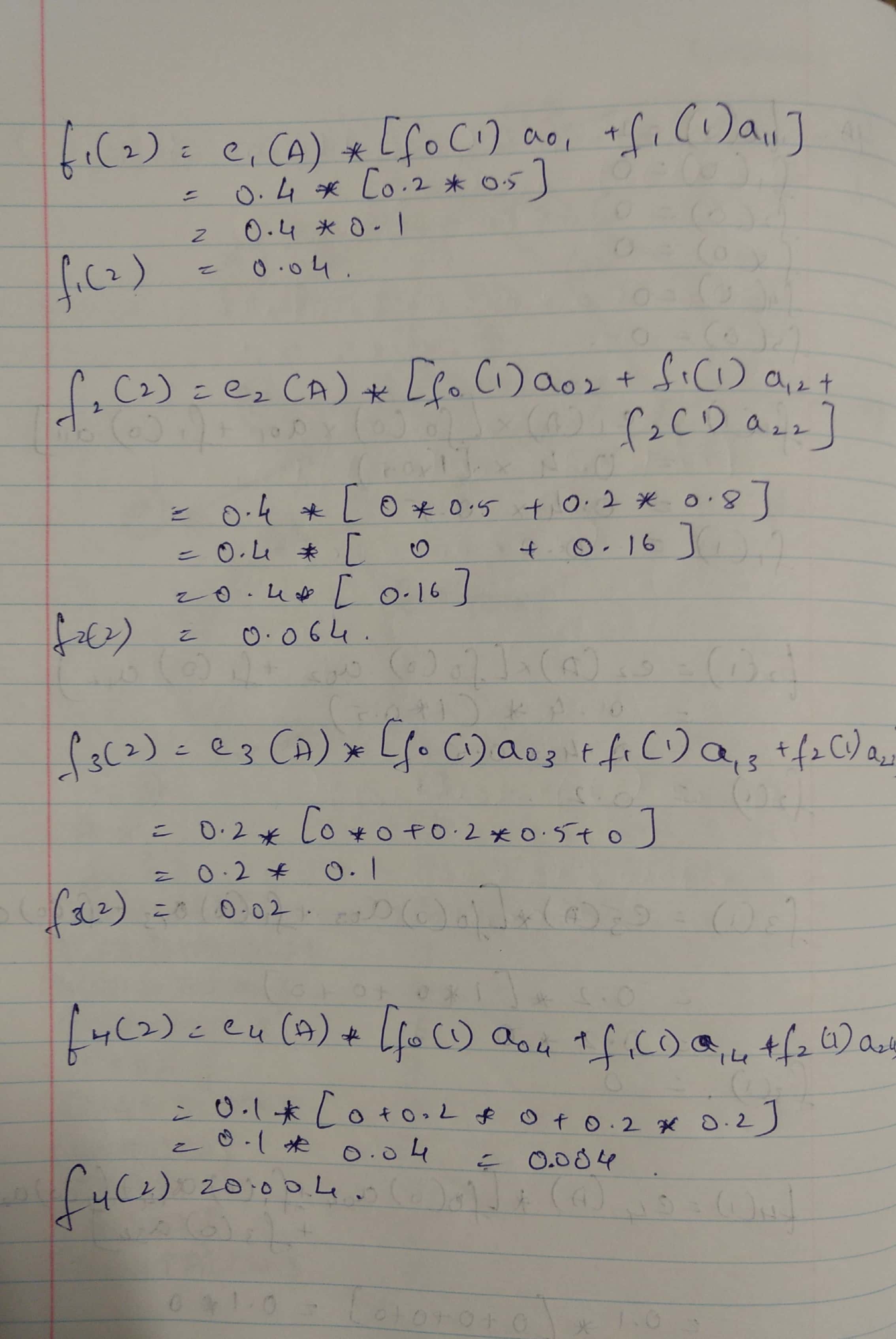


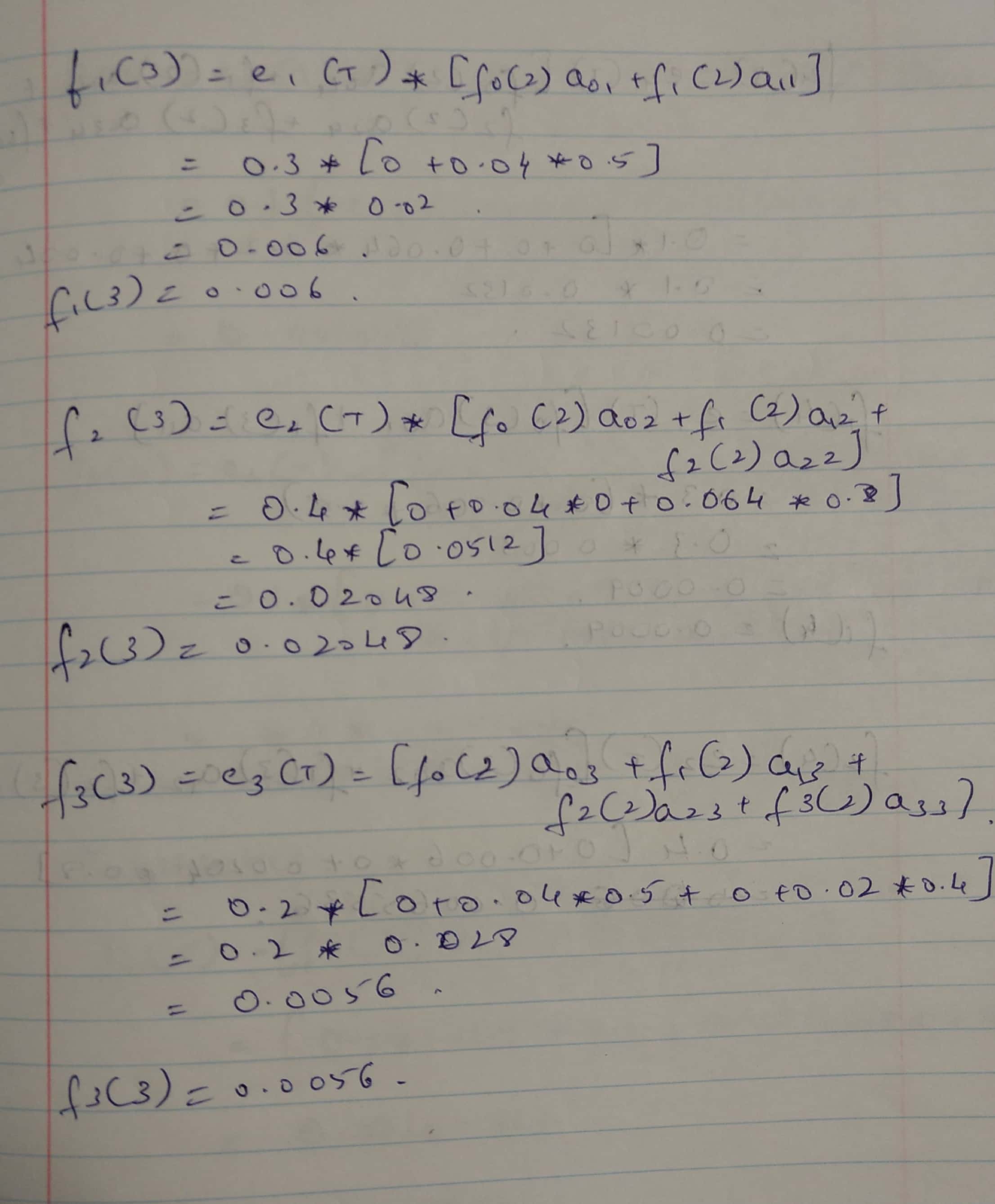
1. **Apply the forward algorithm to infer the probability of observing the sequence AATT. Show each step of the recursion by populating a table of probabilities, 𝑓𝑙(𝑖) with rows representing the position in the sequence, 𝑖 and columns representing the state 𝑙.**

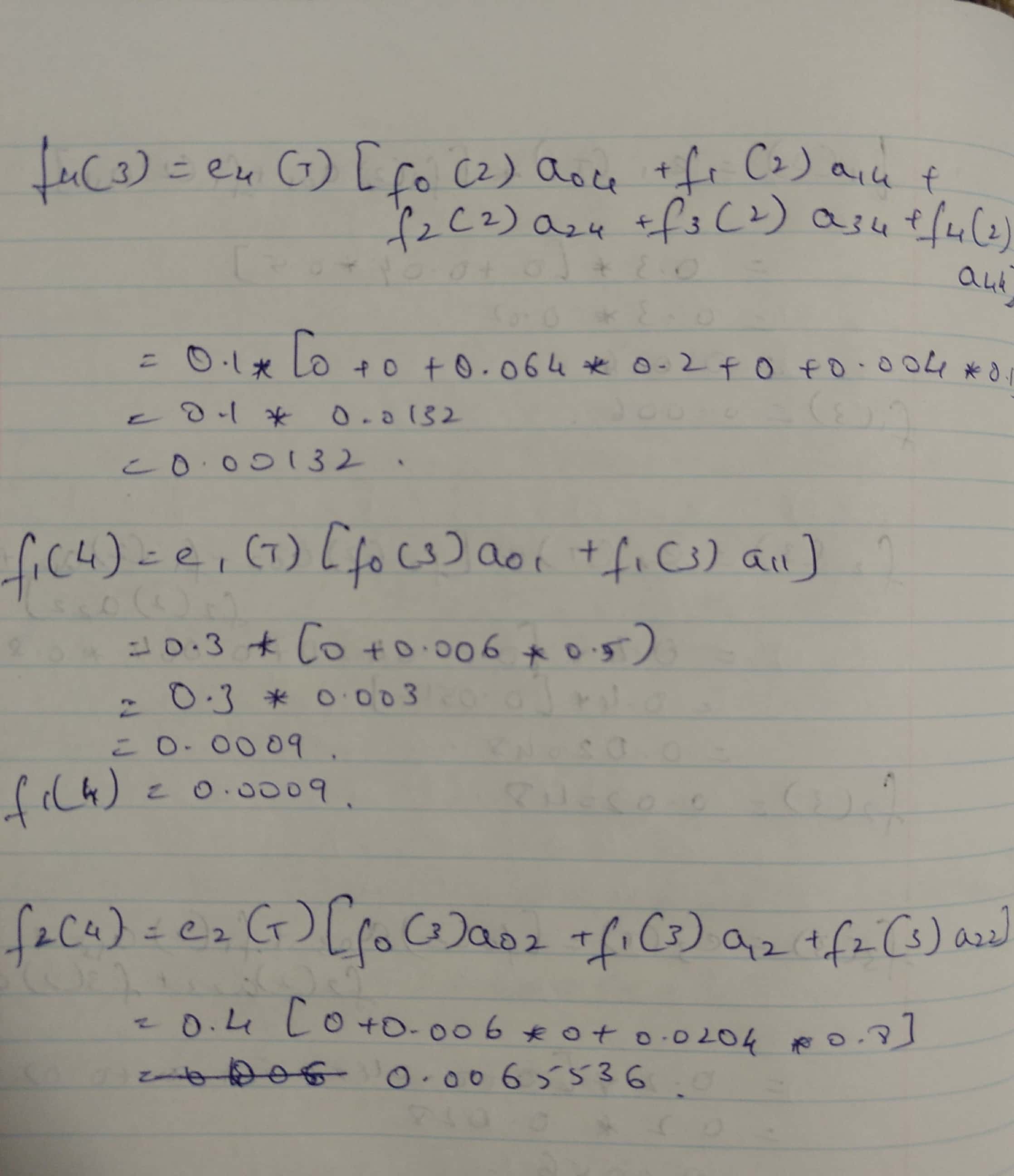
**Answer:**

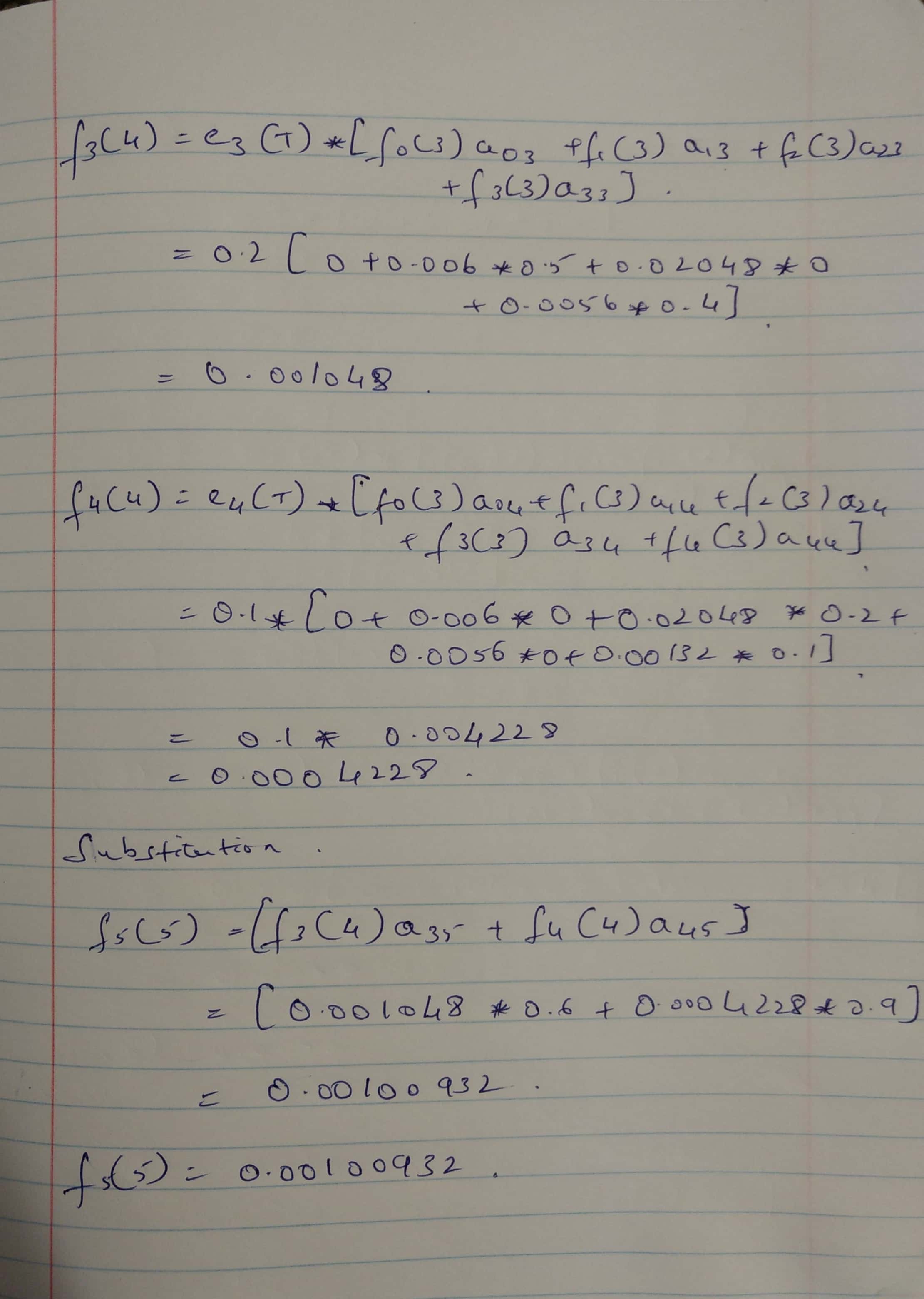
**So, the probability of AATT is:** 0.00100932





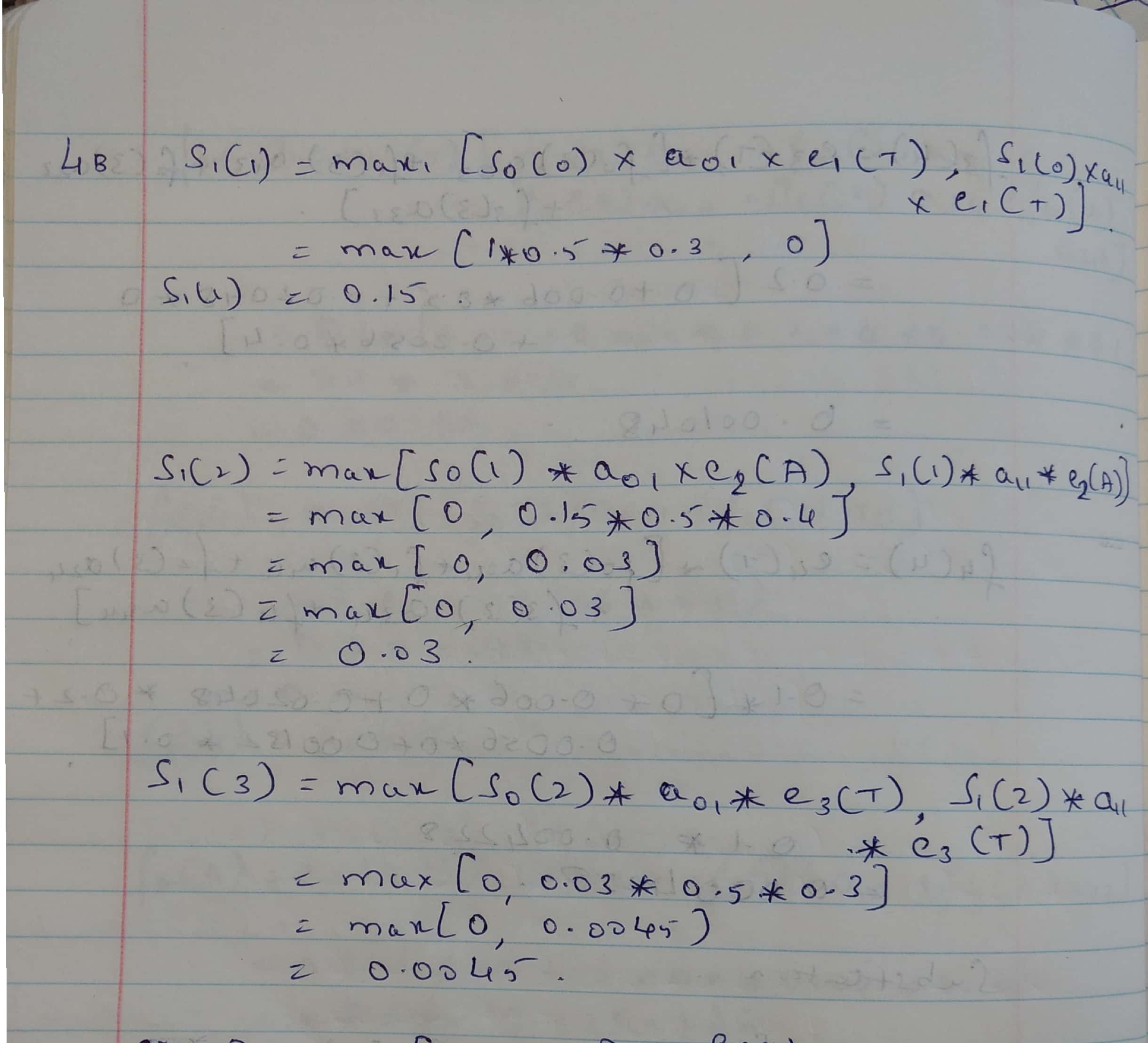


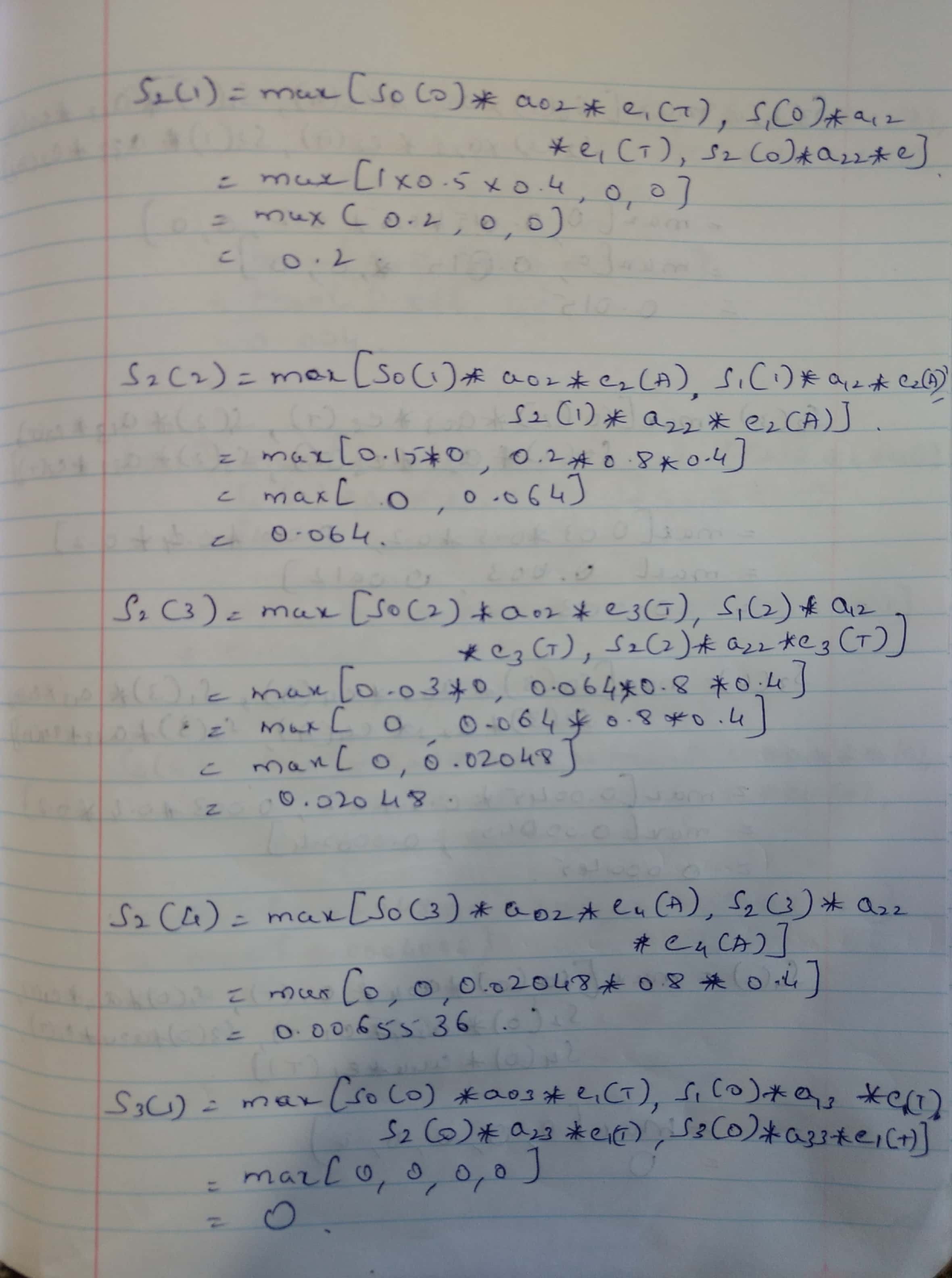


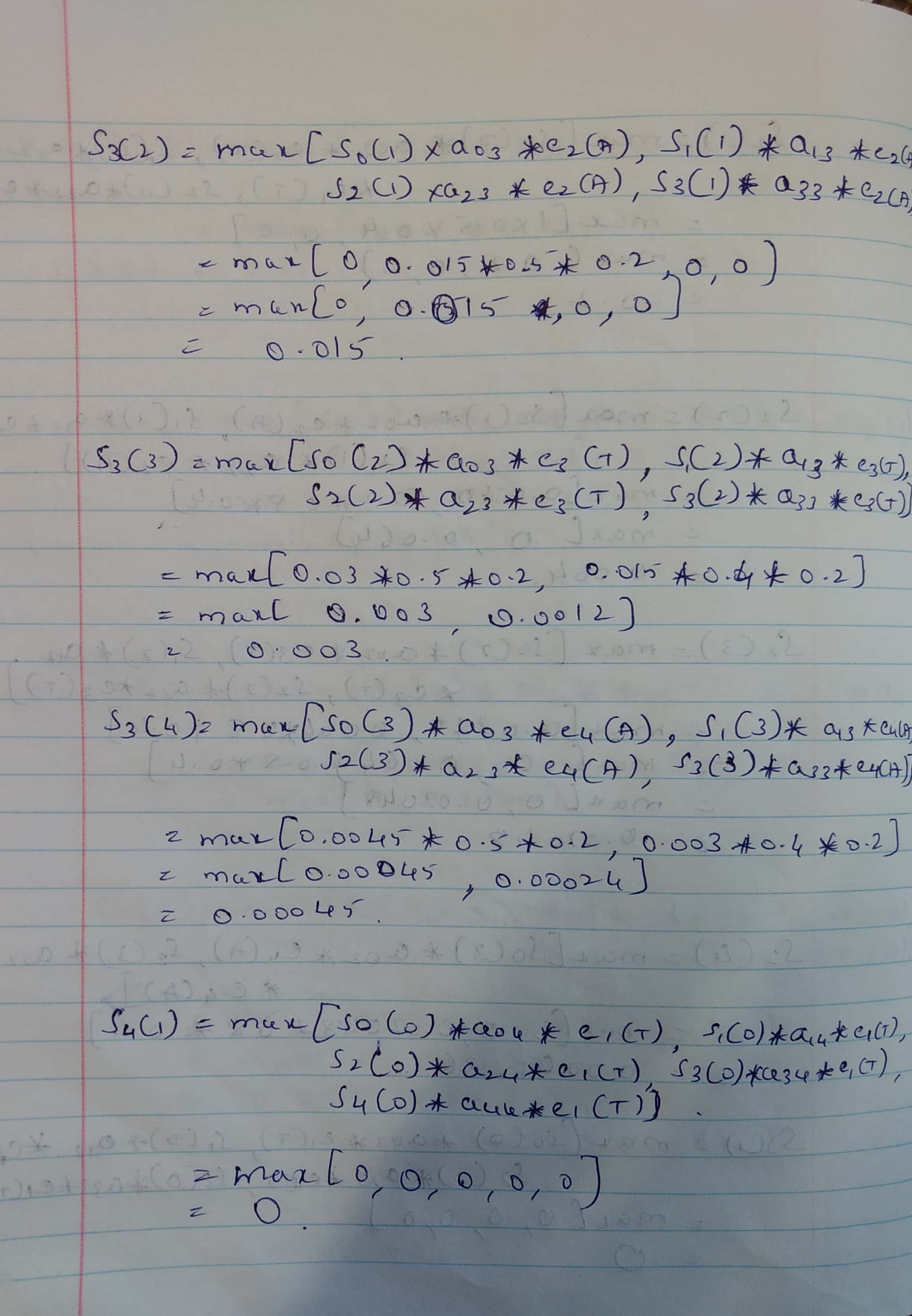


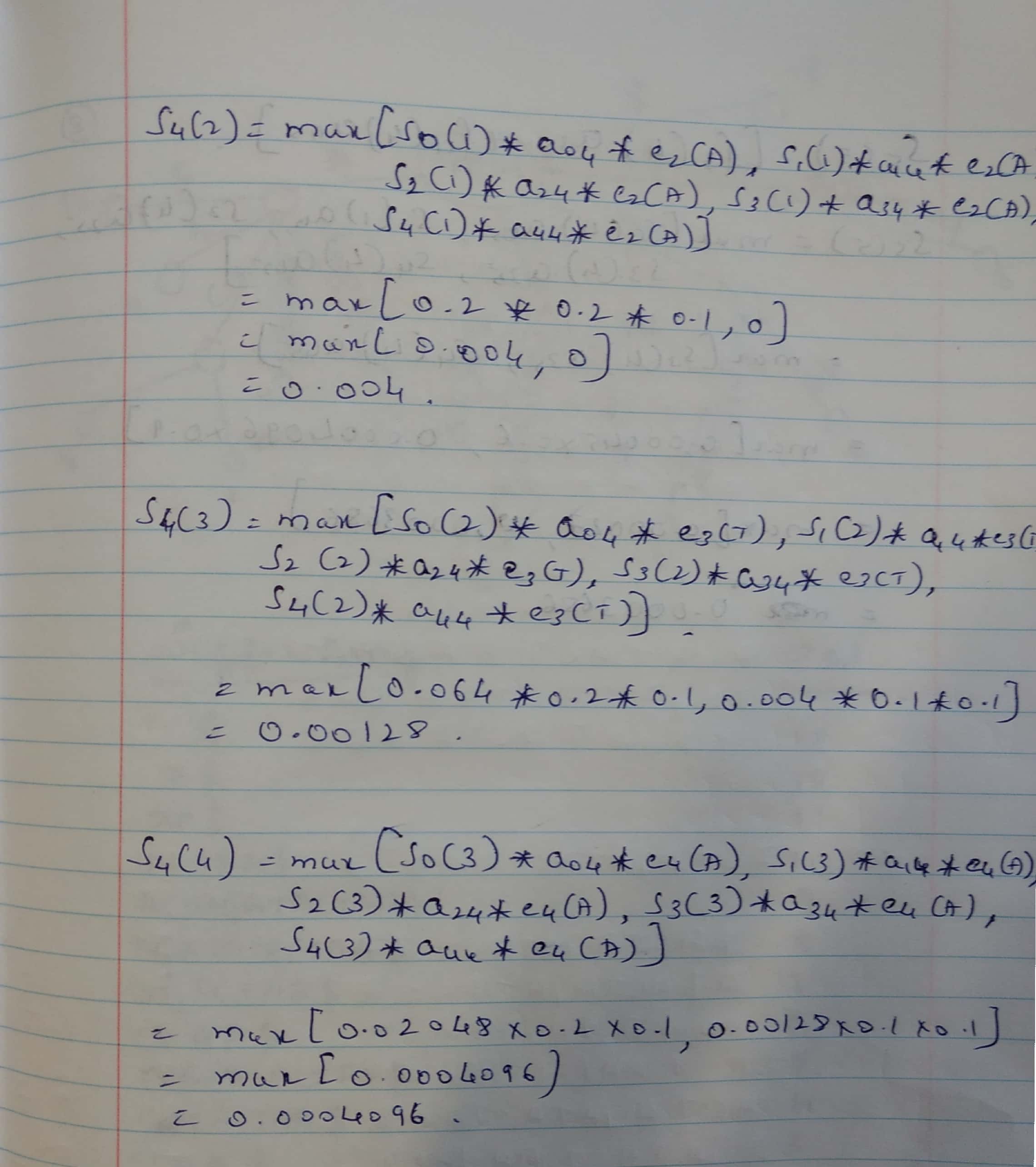
1. **Apply the Viterbi algorithm to infer the most likely state sequence of states of the sequence TATA. Show the steps of how the Viterbi algorithm is applied at each position.**

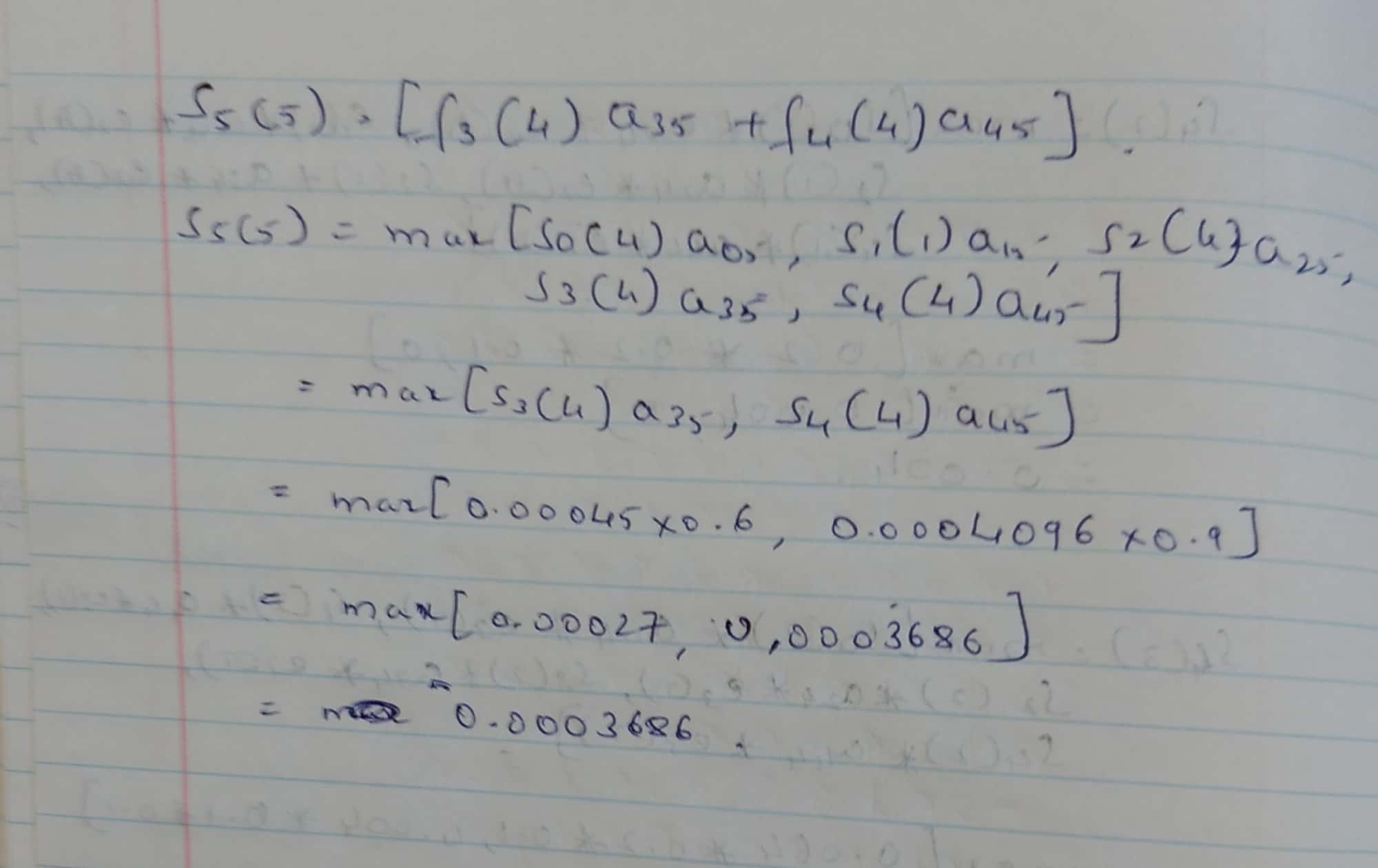
**Answer:**











1. **Squared Error Distortions:**

