HMM

(All the screenshots for all questions are from our terminal)

The state transition matrix A is

	H	C
H	0.7	0.3
C	0.4	0.6

The observation matrix B is

	S	M	L
H	0.1	0.4	0.5
C	0.7	0.2	0.1

The initial state distribution π is

	H	C
π	0.6	0.4

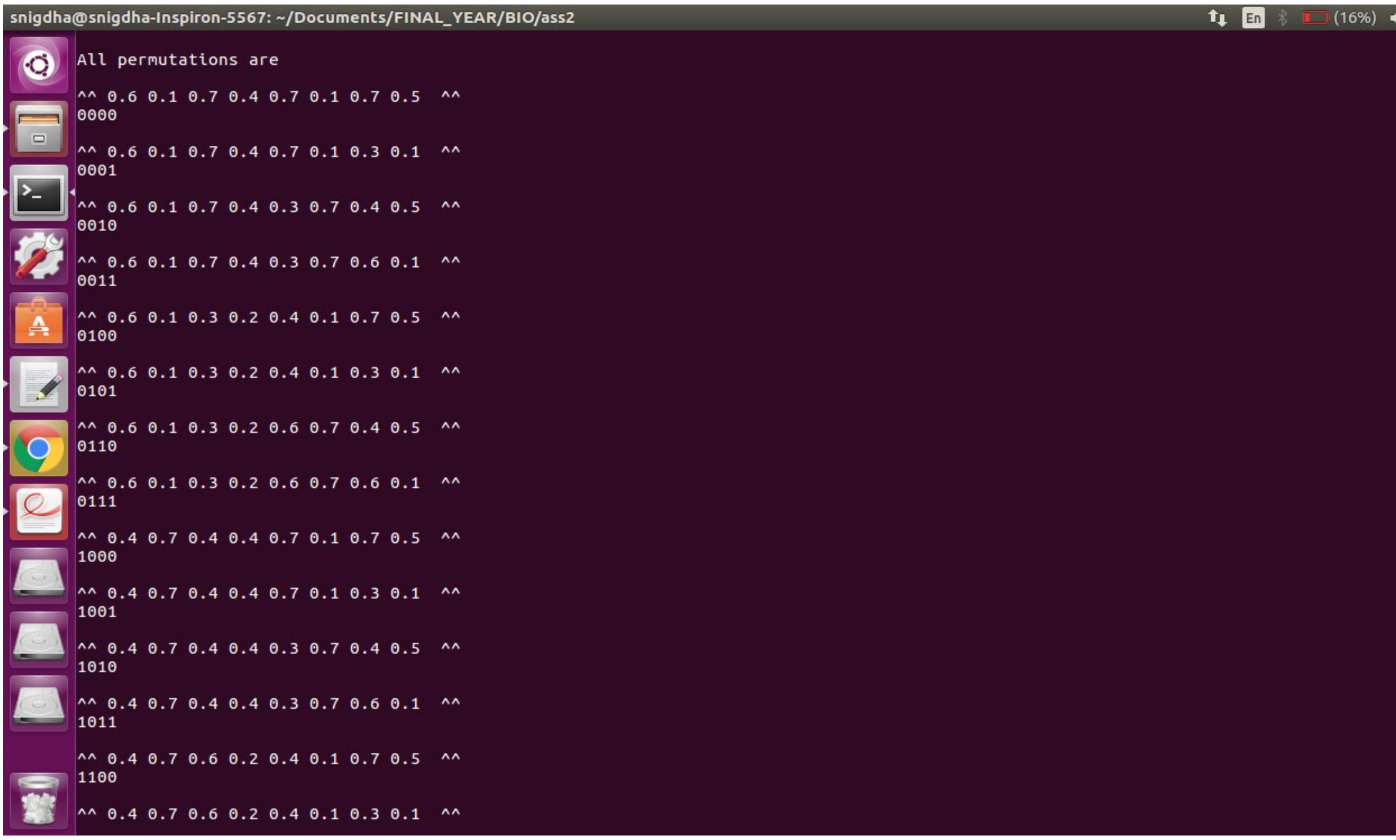
This observation sequence is (S,M,S,L) = (0,1,0,2)

An HMM is defined by A, B and π (and, implicitly, by the dimensions N and M). The HMM is denoted by λ = (A, B, π).
Consider a generic state sequence of length four X = (x0, x1, x2, x3) with corresponding observations O = (O0, O1, O2, O3).
Then πx0 is the probability of starting in state x0. Also, bx0 (O0) is the probability of initially observing O0 and ax0,x1 is the probability of transiting from state x0 to state x1.
Continuing, we see that the probability of the state sequence X is given by

P(X, O) = πx0 bx0 (O0)ax0,x1 bx1 (O1)ax1,x2 bx2 (O2)ax2,x3 bx3 (O3).

NOTE : In our code, ‘0’ stands for H & ‘1’ stands for ‘C’

So for some of the example sequences, probabilities are in the image below:



```
snigdha@snigdha-Inspiron-5567: ~/Documents/FINAL_YEAR/BIO/ass2
All permutations are
^^ 0.6 0.1 0.7 0.4 0.7 0.1 0.7 0.5  ^^
0000
^^ 0.6 0.1 0.7 0.4 0.7 0.1 0.3 0.1  ^^
0001
^^ 0.6 0.1 0.7 0.4 0.3 0.7 0.4 0.5  ^^
0010
^^ 0.6 0.1 0.7 0.4 0.3 0.7 0.6 0.1  ^^
0011
^^ 0.6 0.1 0.3 0.2 0.4 0.1 0.7 0.5  ^^
0100
^^ 0.6 0.1 0.3 0.2 0.4 0.1 0.3 0.1  ^^
0101
^^ 0.6 0.1 0.3 0.2 0.6 0.7 0.4 0.5  ^^
0110
^^ 0.6 0.1 0.3 0.2 0.6 0.7 0.6 0.1  ^^
0111
^^ 0.4 0.7 0.4 0.4 0.7 0.1 0.7 0.5  ^^
1000
^^ 0.4 0.7 0.4 0.4 0.7 0.1 0.3 0.1  ^^
1001
^^ 0.4 0.7 0.4 0.4 0.3 0.7 0.4 0.5  ^^
1010
^^ 0.4 0.7 0.4 0.4 0.3 0.7 0.6 0.1  ^^
1011
^^ 0.4 0.7 0.6 0.2 0.4 0.1 0.7 0.5  ^^
1100
^^ 0.4 0.7 0.6 0.2 0.4 0.1 0.3 0.1  ^^
```

Using the above formula we output Table 1 which calculates the probability & normalized probability of all possible state sequences.
‘printAllKLength’ calls the recursive function ‘printAllKLengthRec’ which in turn calls ‘table1_probability’. ‘k ’ is the size of the observation sequence. In our e.g. it is 4. So we have to calculate probabilities of all possible state sequences from HHHH to CCCC i.e. in our code from 0000 to 1111.

<u>So table-1 is displayed as follows:</u>

TABLE-1 is

probability	normalized probability
0.0004121000	0.0427431000
0.0000351000	0.0036641000
0.0007061000	0.0732741000
0.0002121000	0.0219821000
0.0000501000	0.0052341000
0.0000041000	0.0004491000
0.0003021000	0.0314031000
0.0000911000	0.0094211000
0.0010981000	0.1139821000
0.0000941000	0.0097701000
0.0018821000	0.1953981000
0.0005641000	0.0586191000
0.0004701000	0.0488491000
0.0000401000	0.0041871000
0.0028221000	0.2930961000
0.0008471000	0.0879291000

Then for table 2:
Let λ = (A, B, π) be a given model and let O = (O0, O1, . . . , OT −1) be a series of observations. We want to find P(O | λ). Let X = (x0, x1, . . . , xT −1) be a state sequence.

We perform the forward algorithm, or α-pass. Alpha is defined as
αt(i) = P(O0, O1, . . . , Ot , xt = qi | λ) & calculate using,

$$\alpha_t(i) = \left[\sum_{j=0}^{N-1} \alpha_{t-1}(j) a_{ji} \right] b_i(\mathcal{O}_t)$$

The ‘get_alpha’ computes the ‘α’ matrix. It returns the sum of the last row.

Then, we perform the backward algorithm, or β-pass. Beta is defined as
βt(i) = P(Ot+1, Ot+2, . . . , OT −1 | xt = qi , λ) & calculate using,

$$\beta_t(i) = \sum_{j=0}^{N-1} a_{ij} b_j(\mathcal{O}_{t+1}) \beta_{t+1}(j).$$

In our code, ‘get_beta’ function calculates ‘β’ matrix.

Now, for Gamma t = 0, 1, . . . , T − 1 and i = 0, 1, . . . , N − 1, & is defined as,
γt(i) = P(xt = qi | O, λ)
Since αt(i) measures the relevant probability up to time t and βt(i) measures the relevant probability after time t, it is calculated as,

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(\mathcal{O} \mid \lambda)}.$$

In our code, ‘get_gamma’ function first calculates ‘γ-dash’ matrix & then ‘γ’ matrix by dividing each element of ‘γ-dash’ by the last row sum of ‘α’ matrix.

Below is the screenshot of the alpha,beta,gamma-dash & the final gamma matrix.

```
ALPHA is
0.0600001000 0.2800001000
0.0616001000 0.0372001000
0.0058001000 0.0285601000
0.0077421000 0.0018881000

Last row sum of ALPHA is 0.0096301000

BETA is
0.0302001000 0.0279201000
0.0812001000 0.1244001000
0.3800001000 0.2600001000
1.0000001000 1.0000001000

GAMMA DASH is
0.0018121000 0.0078181000
0.0050021000 0.0046281000
0.0022041000 0.0074261000
0.0077421000 0.0018881000

GAMMA is
0.1881701000 0.8118301000
0.5194321000 0.4805681000
0.2288781000 0.7711221000
0.8039791000 0.1960211000
```