1. Give a clear statement of the problem.

   Given the assumption that ten one hundred nucleotide sequences have exactly one eight nucleotide long motif sequence each, we are tasked with the goal of finding which start locations within each sequence are most likely to be the actual start locations for each motif in that sequence.

2. Give a brief description of the method of your procedure. This should be a few sentences describing the problem, solution, and results. It should be similar to an abstract in a scientific paper.

   By passing in my assumptions and starting with random start indexes, I computed a Motif Model and a Background Model in order to assist in the computation of a probabilities posterior ratio. This ratio enabled me to then perform a roulette wheel selection style of randomness in selecting a more accurate start position for each motif in each sequence. By iterating multiple times using the Gibbs Sampling algorithm, I was able to create a distribution of probabilities that noted the likelihood of each location in each sample being the start position for the motif sequence. Results show that the correct start position for the motifs is selected for each sequence at least ~80% of the time, if not better. The graphs and output below provide a more in-depth breakdown of the results.

3. Present the procedure in detail, including any relevant mathematics. If the solution involves programming, give an overview in words of how the program operates. The presentation should contain sufficient detail so that a reader can follow each step without difficulty. *

```
procedure main:
        parse in sequences
        iterate 5 time:
                call gibbs function
        calculate sum of all counts
        calculate average motif model & background model
        print motif models with 50% or greater probability in each sequence
        display graphs of probabilities for each sequence

procedure gibbs:
        #Note: i refers to sequence, j refers to length of a sequence, w refers to motif length
        initialize A(estimated start positions for each sequences motif sequence)
        set random initial values for A between 0 and len(sequence)-w
        perform burnin step for 1...burnin:
                for each sequence, i:
                        A for current sequence is 0
                        compute Motif Model function
                        compute Background Model function
```

```
            compute the probability from the motif model, function PM()
            compute the probability from the background model, function PB()
            compute ratio function
            roulette wheel selection based off of ratios
            set new position selection as the new A[i]
    initialize 10x100matrix C(counts of selection at each position for each sequence)
    perform sampling step for 1...burnin*2:
        for each sequence, i:
            A for current sequence is 0
            compute Motif Model function
            compute Background Model function
            compute the probability from the motif model, function PM()
            compute the probability from the background model, function PB()
            compute ratio function
            roulette wheel selection based off of ratios
            set new position selection as the new A[i]
            add 1 to count matrix for that sequence and position

procedure Motif Model:
    initialize a 4x8 matrix
    for each position:
        initialize a temp counter for A,C,T,G
        for each sequence:
            add one to the correct nucleotide count
        compute temp (nucleotide count + pseudocount of 1) / (sum of counts + 4)
        adding computed values to the 4x8 matrix as they are found
    return matrix

procedure Background Model:
    initialize a 4x1 matrix and a temp counter for A,C,T,G
    for each position:
        for each sequence:
            add one to the correct nucleotide count if the position is not within the location of the current predicted motif  sequence and is not in the last 7 spots of the sequence
        compute temp (nucleotide count + pseudocount of 1) / (sum of counts + 4)
        adding computed values to the 4x8 matrix as they are found
    return matrix

procedure PM:
    initialize a list
    for each position, j...len(seq)-7:
        initialize a temp to 1.0
```

```
                for each position from j to j+8:
                        multiply current temp by motif model value at [i][i-j]
                add temp value to list
        return list

procedure PB:
        initialize a list
        for each position, j...len(seq)-7:
                append the background model value for the nucleotide at that position
        return list

procedure ratios:
        initialize a list
        for each position:
                compute and append the PM value by the PB value for that position
        return list
```

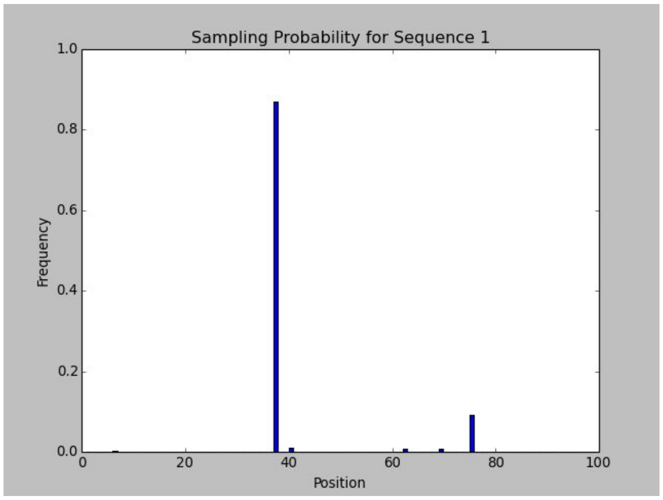
4. If the solution involves programming code, output must be clearly labeled with each item having a descriptive title. Programming code must be documented with comments describing each major step in the program and all relevant variables. Source code should be submitted in a separate file.

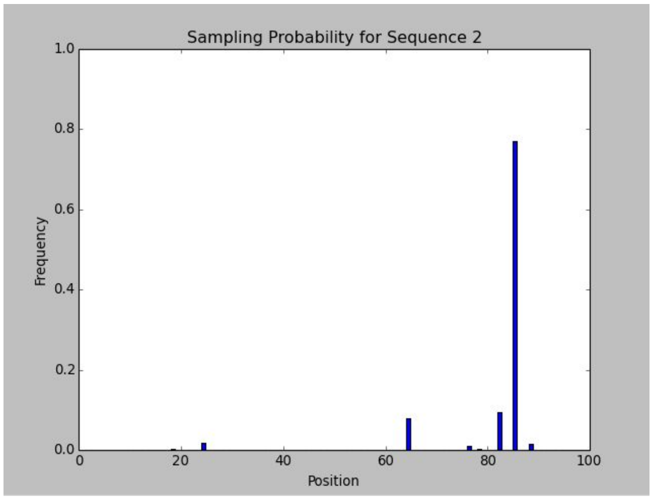
   Output:

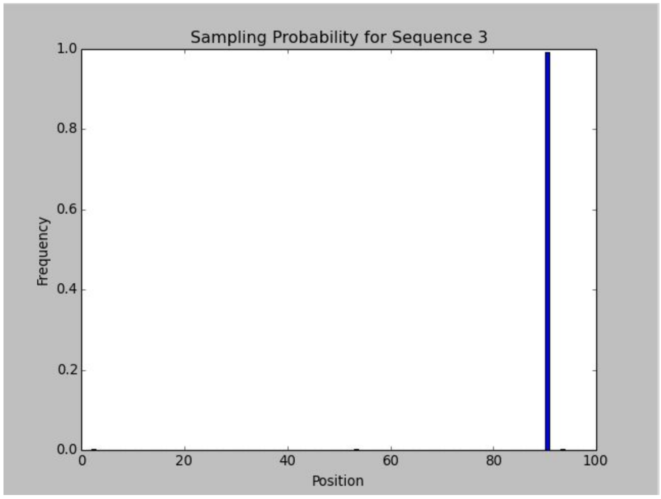
```
Average Motif Model

A: 0.7692307692307693 0.07692307692307693 0.6923076923076923 0.7692307692307693
0.07692307692307693 0.07692307692307693 0.7692307692307693 0.07692307692307693

C: 0.07692307692307693 0.07692307692307693 0.07692307692307693
0.07692307692307693 0.15384615384615385 0.07692307692307693 0.07692307692307693
0.07692307692307693

T: 0.07692307692307693 0.7692307692307693 0.07692307692307693
0.07692307692307693 0.6923076923076923 0.6923076923076923 0.07692307692307693
0.7692307692307693

G: 0.07692307692307693 0.07692307692307693 0.15384615384615385
0.07692307692307693 0.07692307692307693 0.15384615384615385 0.07692307692307693
0.07692307692307693

Average Background Model
A 0.2626131953428202
```

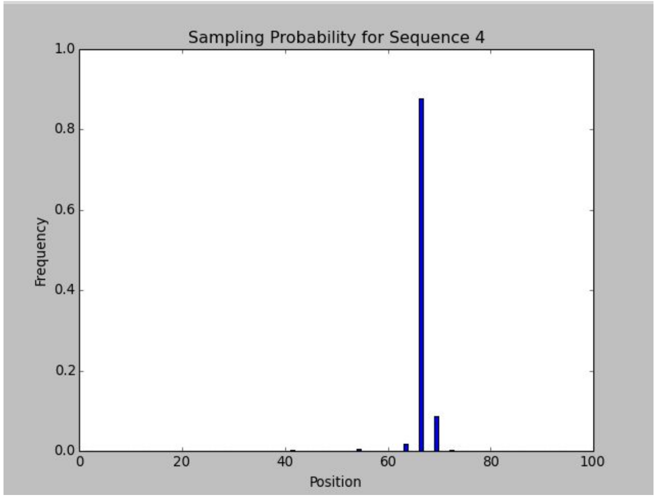
```
C 0.24708926261319536
T 0.222509702457956
G 0.2677878395860285

Motif Locations, Sequences and Frequencies
Seq 1: 37 ATAACTAT 0.8707
Seq 2: 85 ATAATGAT 0.7698
Seq 3: 90 ATAATTAT 0.9914
Seq 4: 66 ATGATTAT 0.8779
Seq 5: 16 ATAATTAT 0.9896
Seq 6: 10 ATAATTAT 0.986
Seq 7: 28 ATAATTAT 0.9952
Seq 8: 64 ATAATTAT 0.9961
Seq 9: 36 ATAATTAT 0.9661
Seq 10: 85 ATAATTAT 0.9808
```

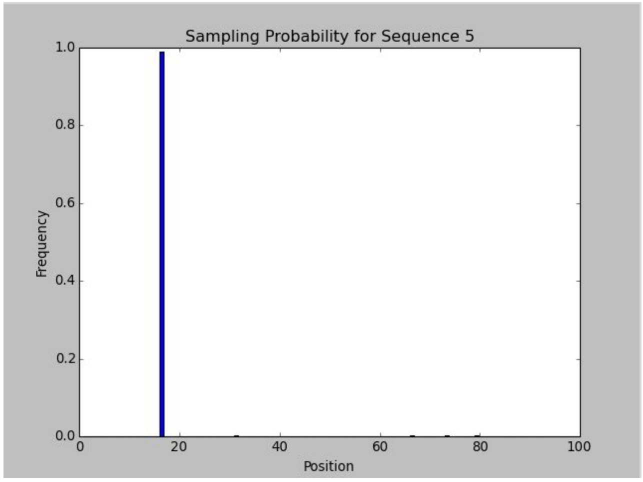
5. Graphs must be titled and have each axis labeled clearly

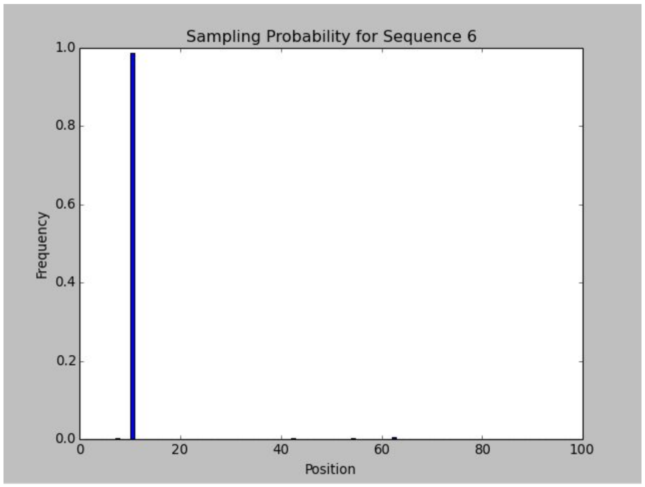


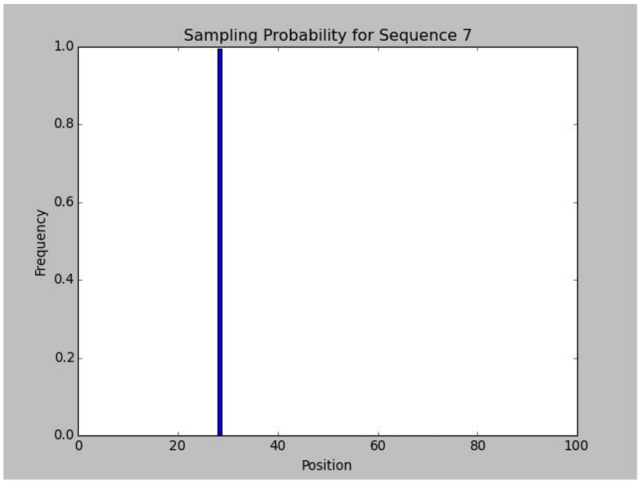



Sampling Probability for Sequence 3
Frequency
Position
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100

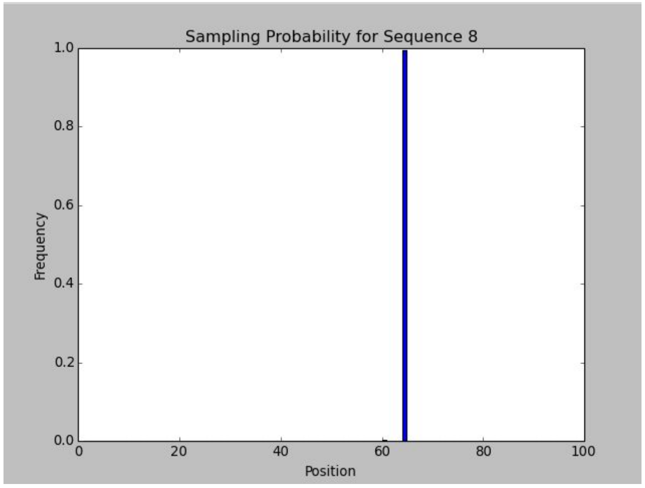

Sampling Probability for Sequence 4
Frequency
Position
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100


Sampling Probability for Sequence 5
Frequency
Position
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100

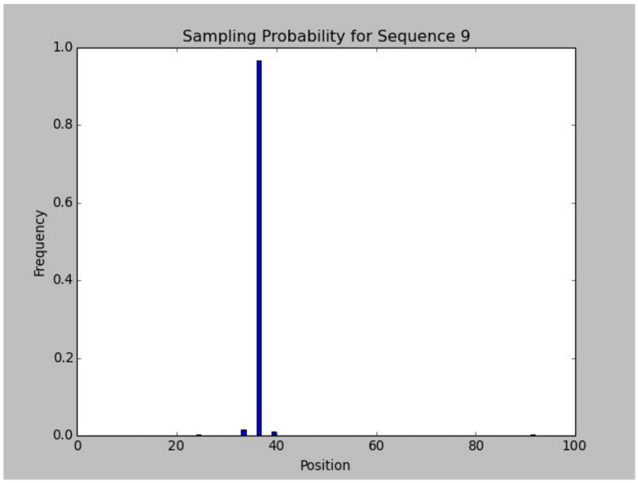

Sampling Probability for Sequence 6
Frequency
Position
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100



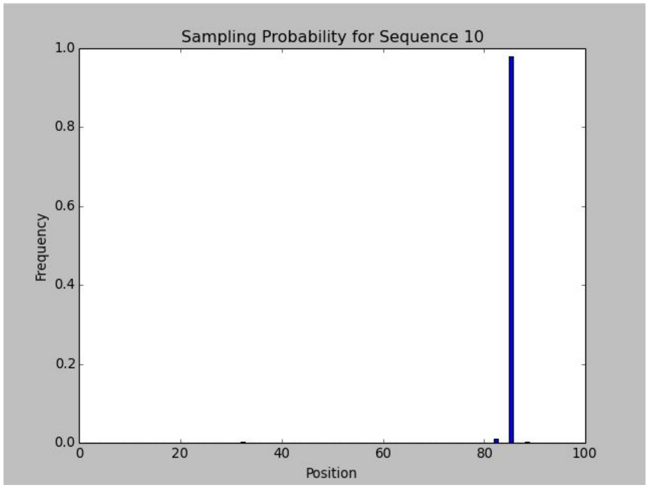
Sampling Probability for Sequence 7



Sampling Probability for Sequence 8



Sampling Probability for Sequence 9



Sampling Probability for Sequence 10