Homework 2- Theory Questions

1) If we assume a Jukes-Cantor substitution model, we will have a transition rate matrix in the form of Q:

$$Q = \begin{pmatrix} -3\alpha & \alpha & \alpha & \alpha \\ \alpha & -3\alpha & \alpha & \alpha \\ \alpha & \alpha & -3\alpha & \alpha \\ \alpha & \alpha & \alpha & -3\alpha \end{pmatrix}$$; $\pi_A = \pi_G = \pi_C = \pi_T = 0.25$, or in our case

If we consider an extremely low substitution rate compared to the time scale of the tree, I think that not too much would happen, since the probability of mutating is very small in every case, we should observe that the sequences still contain near to their beginning amounts of T and C content. They should still both remain close to the original/ancestral 50% in both cases.

2) If the substitution rate is extremely high, we should see the sequences differ quite largely to their ancestral state. I assume that we would observe something around 25% frequency for all nucleotides.

3) According to my code, it would take approximately 601 mya until the transition probability matrix would not change anymore. I got this result using the following code:

```
beta = 0.035
alpha1 = 0.044229
alpha2 = 0.021781
pi = c(0.22, 0.26, 0.33, 0.19)

Q <- create_TN93_Q_matrix(pi, alpha1, alpha2, beta)

for (t in seq(1, 10^3, 50)){
  max_diff <- max(abs(expm(t*Q) - expm(2*t*Q)))
  thresh <- 10^-7
  if (max_diff < thresh){
    print(t)
    break
  }
}
```

4) Since we are following a Markov chain model it makes sense to follow its guidelines to find a solution. A key point of the model is memorylessness. This requires a distribution that does not account for historic events. The exponential distribution would be a good fit to randomly draw the time when the next substitution takes place.
Exponential PDF is: $f(t) = \lambda * e^{-\lambda * t}$
We would set: $-q_{ii} = \lambda$

5) In this case there are 3 nucleotides in between which are to be sampled, the probabilities can be taken from the Q matrix, respectively.
Implementation would look like this:
In the case of looking at a nucleotide A, we would use the probabilities q_{ij} in the respective row, except where i==j.
sample(c("T", "C", "G"), size = 1, prob = c(q_{31}, q_{32}, q_{34}))