**Name: Deepthi Thalagundamatada**

**Student email ID: dt7637@uncw.edu**

**CSC515: Artificial Intelligence**

**Project Report: MARKOV CHAIN**

**MARKOV CHAIN MODEL:**

**QUESTION:** Create a word generator for a language whose example words are {spare, spear, pares, peers, reaps, peaks, speaker, keeper, pester, paste, tapas, pasta, past, straps, tears, terse, steer, street, stare, rates, streak, taste, tapa, peat, eat, ate, tea, seat}.

       Use your program to generate at least 100 random words.

        Are any words repeated?

        What is the probability of generation for each word?

        What are the shortest and longest words generated?

        What is the longest word that can be generated and what is the probability of its occurrence? If you cannot answer this directly, can you provide a bound for the probability of occurrence?

**Markov** has been named after **Andrey** who is the creator of the Markov chain**.**

**Markov** is the probabilistic state machine which is trained with data set. Each states will leads to other state. The previous state do not have any relevance to any subsequent state transitions, which means only the current state is related to the next state and it is not derived from the previous state.

To generate a new random word, the ﬁrst step is to deﬁne which letter to use ﬁrst. For this, the vector is used. A letter is selected randomly (per the earlier discussion using roulette wheel selection). With the ﬁrst letter selected, we emit this letter and then select the next letter using the current letter’s row in the matrix. Next letter selection was explored, When the letter selected is the NULL (or \0) character, the word is complete.

Markov chain, named after its creator, Andrey Markov. A Markov chain is a kind of probabilistic state machine that that can be easily trained given a set of training data. Each state can probabilistically lead to other states, but prior states have no relevance to subsequent state transitions (only the current state)

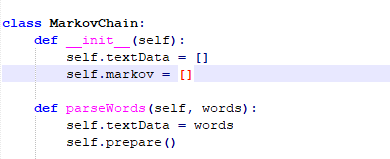
First Step towards random word generation is to define which letter to use first which I made it random.

After the first letter selection, letter is emitted and the next letter is selected according to the matrix formed or the route wheel. While programming we can use list or arrays to do this process.

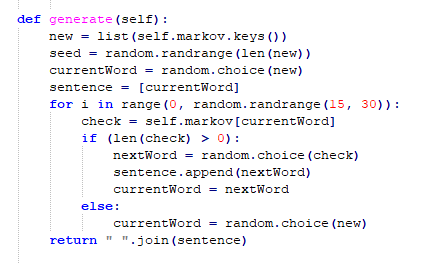
**Implementation:**

**STEP1:** Declaring class and assigning them to array to parse

**STEP2:** PARSE words



**Generating Random words:**



For the given words of questions, the result I got is:

Are any words repeated?

NO

What is the probability of generation for each word?

98%

What are the shortest and longest words generated?

Longest word generated: Tapasteerse

Shortest word: Tasta

**Result:**

