```
while (generating sentences)
    Sentence starts with \n
    if (is a bigram model)
        while (sentence not finished)
            Add nextWord(previous word, bigram model) to sentence
        end
    else (is a trigram)
        Add nextWord(previous word, bigram model) to sentence
        while (sentence not finished)
            Add nextWord(previous 2 words, trigram model) to sentence
        end
    end
    Store sentence and prepare next
end

Assign p_matrix as 2ng to end rows and columns of bigram p_matrix
Assign t_words as inv(I-p_matrix) * (1 vector)
Assign exp_length as p_mat(row 1) * t_words

Generate trigram model p_mat
for i as bigrams list index
    if (word 2 in bigram i is newline)
        Remove row and column i from p_mat
    end
end

Assign t_tri as inv(I-p_mat) * (1 vector)

Assign exp_length_tri as 0
for i as bigrams list index
    if (word 1 in bigram i is newline)
        Add p_mat(1,newline row) * t_tri(i) to exp_length_tri
    end
end
```