Dynamic Programming

Start Assignment

Due No Due Date **Points** 20 **Submitting** a file upload

After a file compression failure, all the white space among words in a text file was lost. Write a C++ program which utilizes dynamic programming to get all of the possible original documents (i.e. with white spaces between words) and ranks them in order of likelihood; all in minimal time.

What you have at your disposal:

- List of unsorted top 100000 English words most used up until August 2005 which includes #comments showing you the popularity class of the word in the language: **dictionary.txt** **(https://camino.instructure.com/courses/78578/files/5397016/download?download_frd=1)**

- List of unsorted top 20000 English words (and 20000 with the first letter capitalized for ease): **dictionary2.txt** **(https://camino.instructure.com/courses/78578/files/5397017/download?download_frd=1)** Note that this library does not have the popularity classes comment so you must assume the popularity classes yourself. The capitalized versions of the words are in the second half of the document so for ease you need to think of them as in the same popularity class as the lower case ones above.

- **dictionary3.txt** **(https://camino.instructure.com/courses/78578/files/5396991/download?download_frd=1)** is a version of dictionary2 in which the lowercase and caps version of the same word are placed back to back.

Methodology:

- Assume your input document will be in the form of a text file

- Your output should be a series of files, each containing one the possible original documents as well as a console print out (and/or additional file) including the ranking of the outputs.

- You will need to use a proper data structure for storing your English dictionary in memory. This will heavily effect your running time.

- For simplicity you may assume that only the words in the dictionary will show up in your compressed text.

- Your program will need to know how to handle punctuation that separate sentences such as: . ? ! ;

- You may remove words with none-English letters and other problems from the library

- You may also remove all the single letter words except for 'I' , 'A', and 'a' from the library or handle them in another way. If you do not handle this situation your program will create exponentially many outputs.

Testing:

- Construct sentences and paragraphs from the words in the dictionary. These need not be semantically meaningful sentences as your task is to generate all possible outcomes using dynamic programming. This means the ranking of outputs is in part based on the popularity class of the words used -- hint hint :D