Program #4 Grading Criteria - 100 pts total

Code Points (20 pts)
Input strings are read into buffer of size 2,000,001 or larger - **2 pts**

Trie struct stores SUM of word frequencies in its subtrie - **2 pts**

Trie struct stores MAX CHILD FREQ and avoids looping through all 26 children to find this frequency on searches (only award if they do solve the issue, not just if they don't have a loop) - **2 pts**

Trie struct has array of 26 for next letter pointer. - **2 pts**

There is an insert function (can be called anything) for the trie. - **4 pts**

There is a search/query function (can be called anything) for the trie. - **4 pts**

Trie is freed at the end. (Just visually check no need to verify with leak detector.) - **4 pts**

Feel free to award partial credit in each category based on "how well" the program satisfies the criteria presented.

Style Points (10 pts)
Header comment w/name, program, date – 2 pts
Appropriate variable names – 1 pts
Appropriate white space and indenting – 1 pt
Appropriate comments about each function (what it does) - 3 pts
Internal comments in code – 3 pts

Process Document (15 pts)
Planning Phase - 5 pts
Assistance - 0 pts
Debugging Phase - 5 pts
Testing Phase - 5 pts

Be fairly lenient here, but make sure this corresponds to their work. But the quality is way worse than what you expect, please take off some points.

Execution Points (55 pts)
5 pts for doing stdin/stdout

There are 10 test cases, each is worth 5 pts. (One is the sample…)

Alternatively, if you debug and fix it, you may take off points for the actual error depending on how big you think it is.

DO NOT SPEND MORE THAN 5 MINUTES LOOKING FOR AN ERROR. IF YOU CAN'T FIND IT, JUST GRADE BASED ON THE ACTUAL TEST CASES.