At first, I wrote more case ‘/’ return tk\_div, which resulted in some failures. After checking, I found that it needs to be compared with eol\_comment to pass, but this problem has been corrected later. And the case ‘\*’ was initially set, only to find that there was no separate ‘\*’, only ‘\*=’ (tk\_mult), which was later modified.In the later stage of correct\_spelling, there is no clue about scientific on how to correct the errors.

The required tokens of the milestone were completed, and problems were encountered in the process. The previously written parse\_number cannot directly use if, and the parse\_integer function must be called first to pass t03 and t04. Generally speaking, there is no big problem in this part, and the subsequent tokens need to be perfected with the correct\_spelling function to pass.

Completed the functions of symbol, eol\_comment and hash\_comment. But I encountered a little problem at first.I used if(next\_char\_isa(cg\_symbol)) directly in parse\_symbol, but I couldn’t parse the second'/', resulting not pass in a few errors.but the errors of tokens are finally passed after correction.

sorry，The system reported an error, so comment out parse\_symbol and resubmit.

This submission completed the four functions of parse\_identifier, parse\_integer, parse\_number, and parse\_string, and then improved the above four functions in parse\_token.

There are questions about parse\_symbol's symbols such as'<<<'. I don't know whether we should use if loop one by one, so this function is not perfect for the time being.

Complete the cg\_wspace, cg\_letter, cg\_integer, cg\_digit, cg\_digit19, cg\_eee, cg\_sign, etc. in char\_isa() according to the rules of tokeniser.h