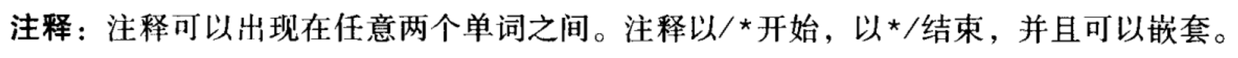
Lab2

How to handle comment

According to the definition in the textbook, comments can occur between any two words, begining with “/*”, ending with “*/”, and it can be nested. So we need another vaviable called comment_level to record.



注释：注释可以出现在任意两个单词之间。注释以/*开始，以*/结束，并且可以嵌套。

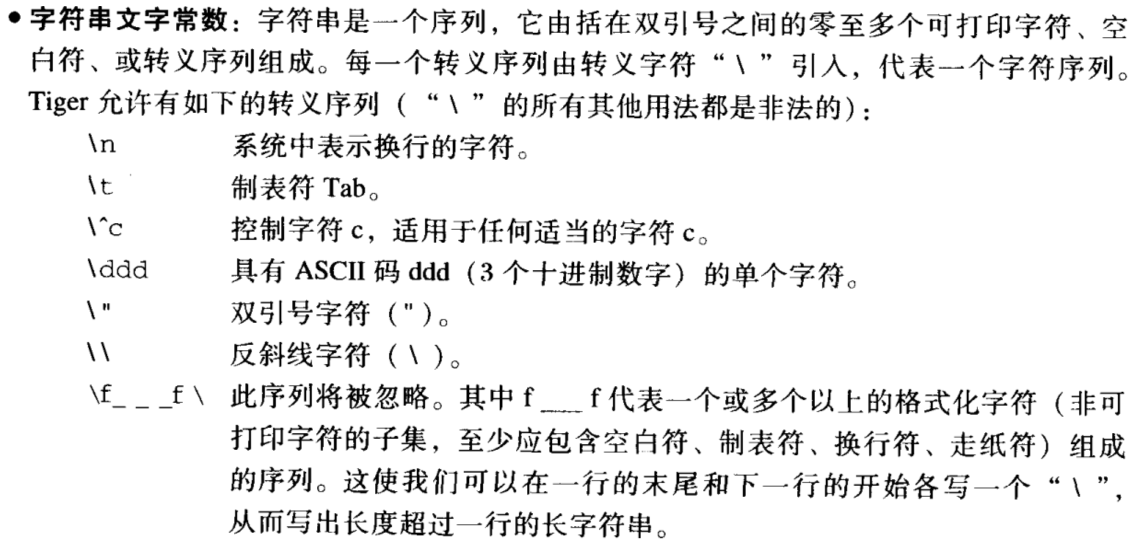
When first time encounter “/*”, enter the COMMENT mode,

Under the COMMENT mode :

- encounter “/*” , add the comment_level by 1
- encounter “*/” , check the comment_level, if equals 0, then comment finished, get back to INITIAL mode
- encouter any other characters, ignore them

How to handle string

String would be a little complicated, with some special escape characters



- **字符串文字常数**：字符串是一个序列，它由括在双引号之间的零至多个可打印字符、空白符、或转义序列组成。每一个转义序列由转义字符“\”引入，代表一个字符序列。Tiger 允许有如下的转义序列（“\”的所有其他用法都是非法的）：

  \n　系统中表示换行的字符。

  \t　制表符 Tab。

  \^c　控制字符 c，适用于任何适当的字符 c。

  \ddd　具有 ASCII 码 ddd（3 个十进制数字）的单个字符。

  \"　双引号字符（"）。

  \\　反斜线字符（\）。

  \f_ _ _f \　此序列将被忽略。其中 f ___ f 代表一个或多个以上的格式化字符（非可打印字符的子集，至少应包含空白符、制表符、换行符、走纸符）组成的序列。这使我们可以在一行的末尾和下一行的开始各写一个“\”，从而写出长度超过一行的长字符串。

When first time encounter `“””` , enter the STRING mode

- encounter `“””`, STRING finished, get back to INITIAL mode
- encounter `“\n”`, add it
- encounter `“\t”`, add it
- encounter `“\^c”`, add the ASCII value corresponding to it
- encounter `“\ddd”`, add the character
- encounter `“\””`, add it
- encounter `“\\”`, add it
- encounter `“\f___f\” `, ignore them

How to handle error

there exist some characters that can not be parsed, moving the tok_pos_to the errormsg_ to indicate the error, ant try not to crash the program

How to handle EOF

Handling End-of-File (EOF) in a lexical analyzer (lexer) involves properly detecting when the input stream has been exhausted and ensuring that any necessary cleanup or state transitions are performed.

When facing the EOF, meaning the program already finished, just return.