0. 若一行以单个反斜杠字符结尾且反斜杠后无空白，则删去反斜杠，将下一行的所有文本移至本行，并再次进行此判断。结束后，将等量的空行放在本行后。即此操作不改变后续文本的行号。

**替换相关预处理：**

1. #s *deliminator pattern deliminator string deliminator name*

将后续文本中*pattern*的所有匹配替换成*string*，直到*name*相同的ends指令出现为止。

示例：#s/abc/def/name

#s abc def name

*deliminator*为一个任意非标识符字符[^a-zA-Z0-9\_]。

*pattern*与*string*内的特殊字符依照ECMAScript规则（即C++标准库<reg\_ex>中默认的语法规则）替换

*pattern*与*string*按照全部字面意思读取，包括空格与制表符，但不包括换行符。

*pattern*与*string*内不能包含该非标识符字符。

*name*不得包含非标识符字符。

若有两个*name*相同则报错

2. #s *deliminator pattern deliminator string*

将后续文本中*pattern*的所有匹配替换成*string*，直到文件尾。

3. #ends *deliminator name*

结束名为*name*的替换。

（s代表subsitute）

4. #define *identifier* *string*

示例：#define abc defxxx

与C中作用类似。

*identifier*不得包含非标识符字符。

若*string*中不包含斜杠则等价于

#s/\babc\b/defxxx/abc

（总之就这个意思，即使后面包含了斜杠）

5. #define *identifier*(*paraname1*, *paraname2*, ...) *string*

示例：#define abc(d,e,f) (d+e+f)

会将文本中abc(1,3,5)替换成(1+3+5)

*identifier*与*paraname*不得包含非标识符字符

等价于

令str=*string*中将*paraname1*替换成'$01'，将*paraname2*替换成'$02'，以此类推的字符串。在示例中，str就是($01+$02+$03)

#s \babc\(([^,]\*),([^,]\*),([^,]\*)\) ($01+$02+$03) abc

6. #undef *name*

等价于

#ends *name*

**包含相关预处理：**

7. #ecli<*eclfile*>

8. #anim<*anmfile*>

1~8预处理指令均被替换成空行。

**其余预处理：**

9. #rawins(id, difficulty\_mask, post\_pop\_count, param\_count, param\_mask, param1, param2, ...);

替换成以空格分隔的纯数据，以分号结尾。