

思路

1. 在 `pass1` 中

1. 读入汇编代码的某一行，判断是否为注释行，空行，若不是，则分解代码。在分解代码的同时判断引号 `""` 的输入，认为第一个引号和最后一个引号之间为引号中间部分
2. 然后根据第一个单词是否是语言限定符（如ADD），决定是否加入查找表SymbolTable，如果在查找表中已经有了，则不能加入，不能有两个相同的label，提示相应信息，pass1不过。如果没有则加入。
3. 对于任何一条指令，都会将其行号，PC记录下来放到Code结构体中，然后这个结构体加入代码向量中
4. 另外还需要判断 `ORIG` 和 `END` 是否存在，如果不存在，则pass1不过，提示相应信息

2. 在 `pass2` 中

1. 根据不同的指令，进入相应的函数进行判断，产生相应的机器码以及相应的报错信息
2. 某一些指令的判断类似，可以进入相同的函数进行判断、产生机器码
3. 对于每行的代码，采用Code结构体，结构体中包含
 1. codeparts，代码的每一部分字符串，不包括标签
 2. PC，指令的PC
 3. LC，指令原本所处的行

功能

输入程序的指令可大写可小写。

立即数的前标需要是小写，如x, b

可以判断偏移量是否超出范围

判断是否有重复的LABEL

在使用LABEL时判断是否有这个LABEL

对于错误，会报道相应的行数

examples中的a.asm的第一个程序中几乎包含了所有指令，没有实际意义，其他asm均为前几个实验，测试可行