**LC-3 汇编器**

**PB19051175 梁峻滔**

**1. 设计思路**

该汇编器用C编写，使用一个二维字符数组存储一个字符串形式的汇编程序，此外还定义两个全局二维字符数组，一个作为符号表存储汇编程序出现的Label，另一个存储LC-3汇编语言的关键字，以在创建符号表时辨别某一字符串是否是Label。

采用两边扫描的方案，第一遍先识别程序中的Label并创建符号表，第二遍利用符号表将各条指令翻译成机器码形式。

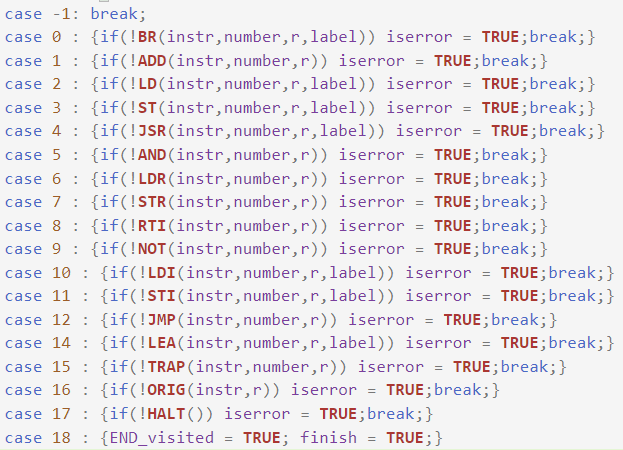
整个汇编器C程序中包含的函数主要有：

① 一个判断指令类型的assemble函数，该函数的功能是传进一行汇编指令，通过字符串匹配辨别该指令是属于哪一种指令，返回对应的编号。



其中in[row][MAX\_NUM]为要翻译的汇编程序，r为行号。

② 各种类型的指令的翻译函数，assemble函数识别某一行汇编指令的类型后，通过返回的编号值，跳转至相应的指令翻译函数来将该行汇编指令翻译成机器码。每个翻译函数的基本操作都是字符串匹配来判别所需要翻译的寄存器、地址偏移等，并返回执行成功/失败信息。



③ 创建关键字表函数



④ 创建符号表函数



⑤ 补码转换函数



⑥ 二进制表示转换函数



**2. 功能**

该汇编器能够识别某些指令的不同的寻址模式；能够创建符号表；能够识别BR、JSR、LD、LDI、ST、STI等指令中的Label，并判断该Label是否是符号表中出现的符号，若是则计算相应的地址偏移量；通过识别.END判断翻译停止。总而言之就是可以将LC-3的15种指令和.ORIG的汇编形式翻译成对应的机器码形式，并通过打印出来的形式显示。

支持的错误检测：

① 指令中的立即数或偏移量是否越界；

② 指令中的Label是否是符号表中已有；

③ 汇编程序中是否缺少.END。

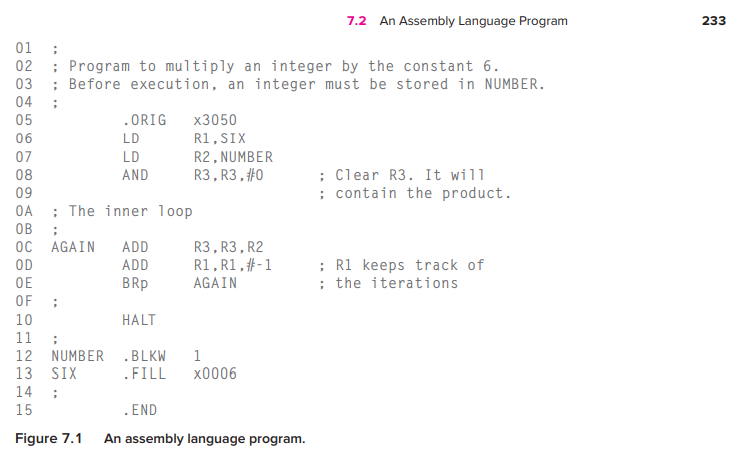
未能完善的部分：

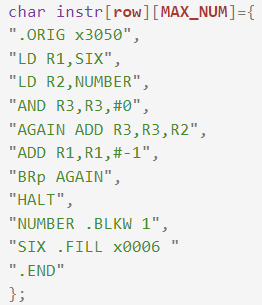
① 存放在二维字符数组中的汇编程序不能有整行注释，即要求每一行都有一条有效指令，这是因为各翻译函数中计算地址偏移量时都使用了该指令的行号来计算；

② 未能实现.FILL、.BLKW、.STRINGZ伪指令的相应翻译，所以要利用翻译出来的机器码在LC-3模拟器上运行的话，程序中.FILL、.BLKW、.STRINGZ的功能需要预先手动设置。

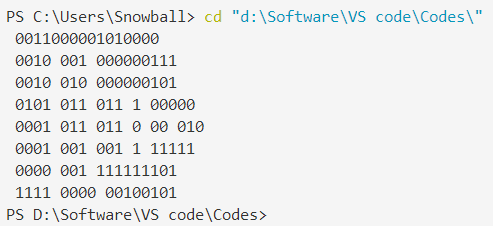
**3. 示例测试**

采用课本的乘6的程序作为测试示例：

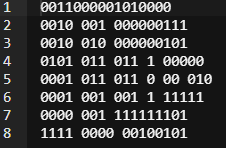


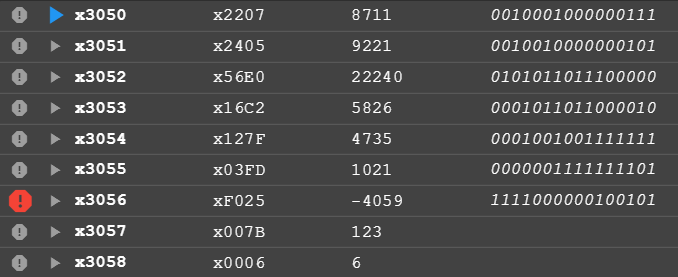


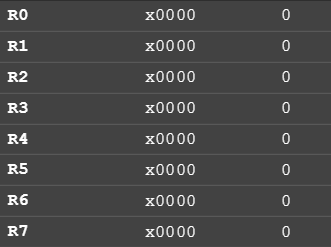
该程序是将最初存放在NUMBER的整数乘以6，与教材一致，假设存放在NUMBER的整数为123，则最后存放在R3的结果应是. 汇编器翻译的机器码如下所示：



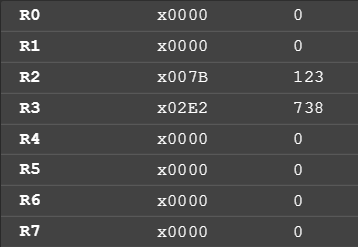
将该翻译结果复制至LC-3模拟器，预先设置NUMBER(x3057)处存放整数123，SIX处存放整数6：







运行后的结果为：



与预期结果相符。

由于个人时间、精力、水平有限，加之面临其他各门课程的期末考试，且汇编器提交时间截止在即，目前的汇编器只能完成最基本的指令翻译功能，很多地方尚未完善，对于可能给助教们带来的检查上的麻烦，我深表歉意，感谢老师在本课程上的教导和助教们的巨大帮助。