Assignment 5

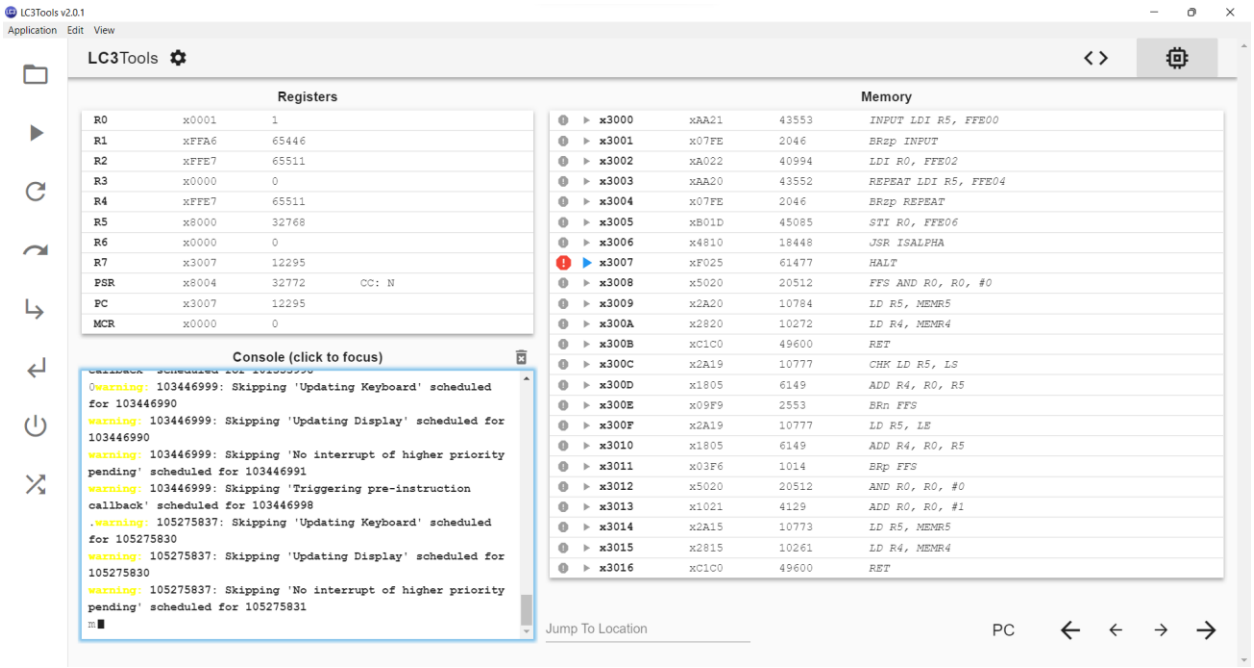
Alphabetic.asm

```
        .ORIG x3000
INPUT   LDI R5, FFE00
        BRzp INPUT
        LDI R0, FFE02
REPEAT  LDI R5, FFE04
        BRzp REPEAT
        STI R0, FFE06
        JSR ISALPHA
        HALT



FFS     AND R0, R0, #0
        LD R5, MEMR5
        LD R4, MEMR4
        RET
CHK     LD R5, LS
        ADD R4, R0, R5
        BRn FFS
        LD R5, LE
        ADD R4, R0, R5
        BRp FFS
        AND R0, R0, #0
        ADD R0, R0, #1
        LD R5, MEMR5
        LD R4, MEMR4
```

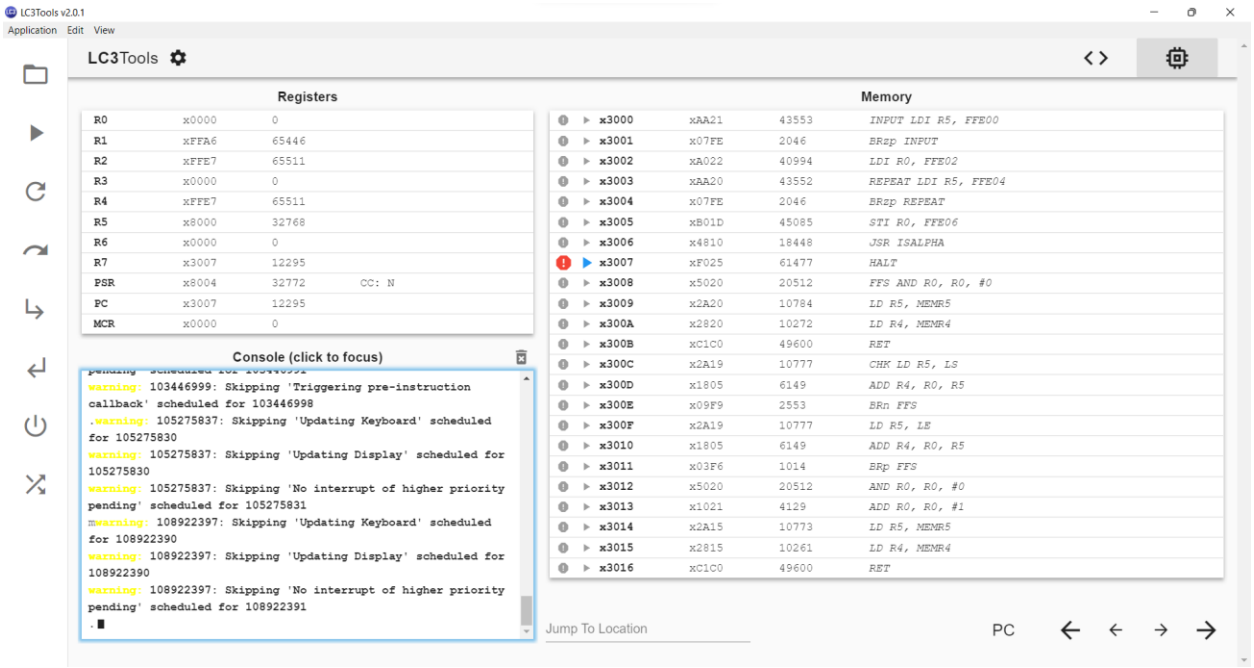
```
        RET
ISALPHA ST R5, MEMR5
        ST R4, MEMR4
        LD R5, US
        ADD R4, R0, R5
        BRn FFS
        LD R5, UE
        ADD R4, R0, R5
        BRp CHK
        AND R0, R0, #0
        ADD R0, R0, #1
        RET



FFE00    .FILL xFE00
FFE06     .FILL xFE06
FFE04     .FILL xFE04
FFE02    .FILL xFE02

LS  .FILL #-97
US  .FILL #-65
UE  .FILL #-90
LE  .FILL #-122

MEMR5  .BLKW 1
MEMR4  .BLKW 1
```

.END

Running when input is a letter (Supposed to be 1 at R0)

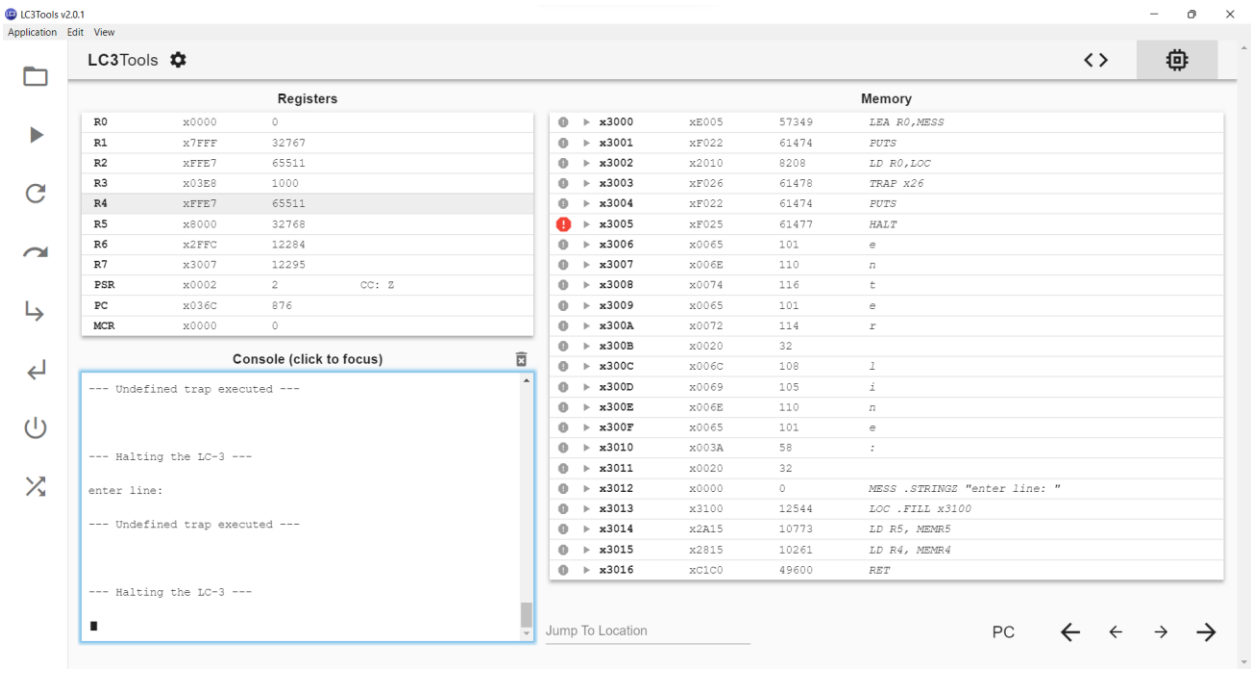


Running when input is not a letter (Supposed to be 0 at R0)



2- .ORIG x3000

```
        LEA R0,MESS
        PUTS
        LD R0,LOC
        TRAP x26 ; string STd at x3100
        PUTS
        HALT
MESS    .STRINGZ "enter line: "
LOC     .FILL x3100
        .END


        .ORIG x028A
        ADD R4, R0, #0
UI   GETC
        OUT
        ADD R3, R0, #-10
        BRnp KP
        RTI

KP   STR R0, R4, #0
        ADD R4, R4, #1
        BR UI
        .END
```



3-

A) TRAP X72
B) Yes, after the trap instructions, RTI will save the program counter

4-

You can infer from x34 the location (x1000) and from x1000 the address at which the trap instruction is set to start