Shanghai Jiaotong University
Introduction to Computing Systems

Introduction to Computing Systems Experiment2

Professor: Yale Patt

Name: Hong Yining ID: 515021910453 Date: July 20, 2018

1 How to Use my Program

My program is under the format ".asm". To turn it into an object, you have to click the asm button in LC3edit.
Then, we open LC3 simulate. We load "directory.obj" first, and then "linked-list.obj". We click the "Run Program" button. We can see that what appears in the console window is exactly what the experiment requires us to do.
Note that after we search for a room number and we wanna search for another room number, we have to RELOAD PROGRAM and run program again.

2 How my Program Works

Figure 1 is my program. I will introduce in detail how I implement it.

2.1 Prompt

We save the address of prompt "Type a room and press Enter" in **R0**, so that with **Trap x22**, we can output this sentence.

2.2 Keyboard Input

NUMBER where we store the keyboard inputs. We save the address of NUMBER in **R1**. And we notice that "Enter" in ASCII is x0A, we find the negative "Enter", which is NEGENTER in this program and save it in **R3**. In **AGAIN** of the program, we keep tracking the inputs of the user using **Trap x20**, and Trap x21 displays the inputs. We keep subtracting the NEGENTER from the input so we can know when exactly the user presses Enter and we can then branch to next session "LIST". If enter is not pressed, we keep increment R1 and so that the new character can be saved in the next location.

2.3 LINKLIST

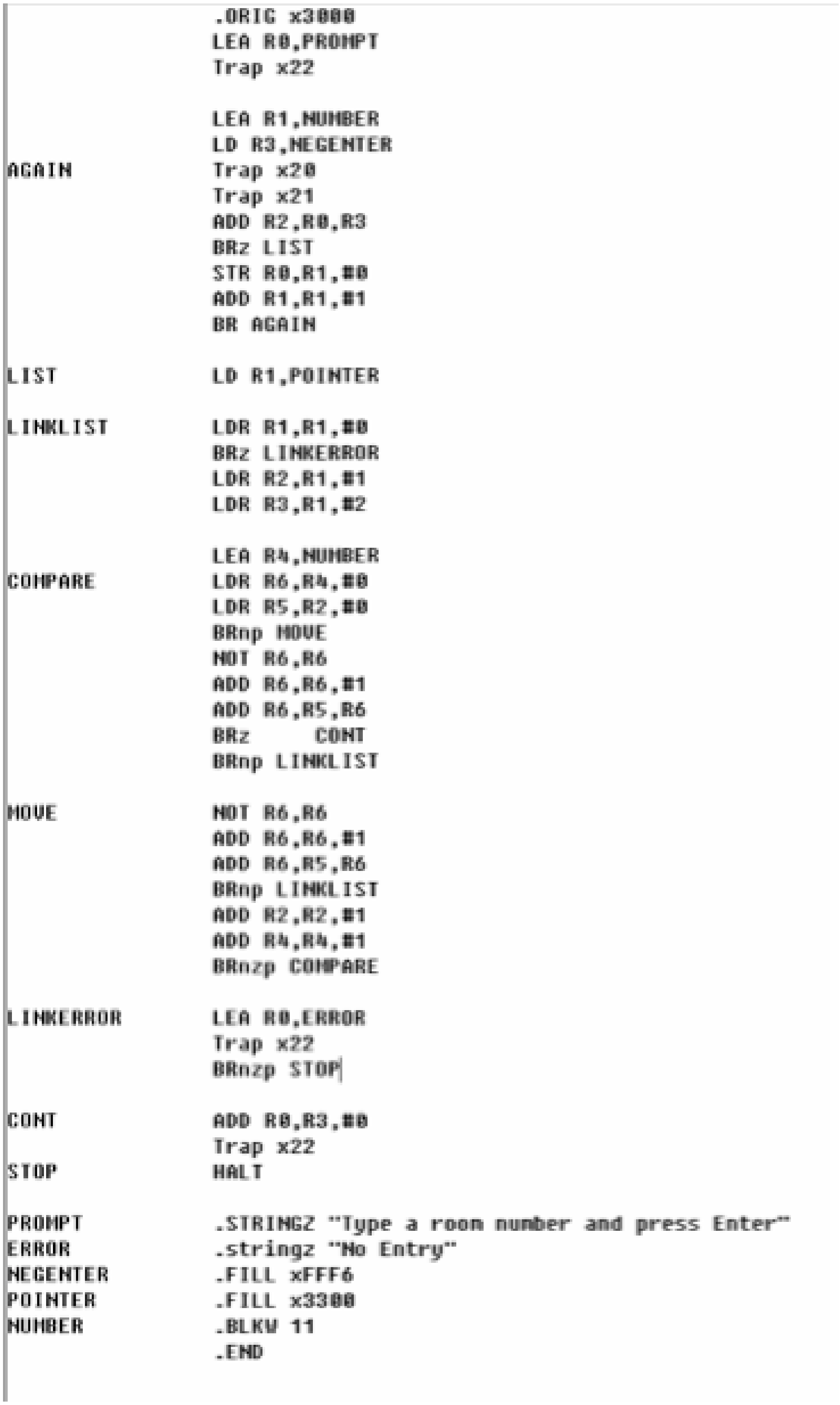
We store the address of the pointer in **R1**. So that we can use LDR to get the content of the pointer(which is the address of next pointer), we can get the room number and name as well using LDR,#1 and LDR,#2, and load them in **R2** and **R3**.

2.4 COMPARE

We compare the input and room numbers by subtracting each character of the string. We load them in **R4** and **R2** and get the values using LDR. We increment **R4** and **R2** in a loop to get the next characters. If we come to the last character of the string and so the next character NULL, which is 0, we branch to "MOVE", which is moving to the next room.

2.5 OUTPUT

If we cannot find a room which matches the room number, we load the address ERROR, which is "No Entry" in **R0**. If not so, we load the value stored in R3, which is the name of the room, in **R0**. And by using **Trap x22**, we can ouput the name of the room



```
                .ORIG x3000
                LEA R0,PROMPT
                Trap x22

                LEA R1,NUMBER
                LD R3,NEGENTER
AGAIN           Trap x20
                Trap x21
                ADD R2,R0,R3
                BRz LIST
                STR R0,R1,#0
                ADD R1,R1,#1
                BR AGAIN

LIST            LD R1,POINTER

LINKLIST        LDR R1,R1,#0
                BRz LINKERROR
                LDR R2,R1,#1
                LDR R3,R1,#2

                LEA R4,NUMBER
COMPARE         LDR R6,R4,#0
                LDR R5,R2,#0
                BRnp MOVE
                NOT R6,R6
                ADD R6,R6,#1
                ADD R6,R5,R6
                BRz     CONT
                BRnp LINKLIST

MOVE            NOT R6,R6
                ADD R6,R6,#1
                ADD R6,R5,R6
                BRnp LINKLIST
                ADD R2,R2,#1
                ADD R4,R4,#1
                BRnzp COMPARE

LINKERROR       LEA R0,ERROR
                Trap x22
                BRnzp STOP

CONT            ADD R0,R3,#0
                Trap x22
STOP            HALT

PROMPT          .STRINGZ "Type a room number and press Enter"
ERROR           .stringz "No Entry"
NEGENTER        .FILL xFFF6
POINTER         .FILL x3300
NUMBER          .BLKW 11
                .END
```

Figure 1: My Program