LAB 2

1. INTRODUCTION

This lab requires to write a LC-3 program to output the full-names and office room numbers corresponding to the input name, based on the information stored in the directory. The information will be stored as a linked list.

According to the requirements, we need to prompt the user to type a name, compare it with the first names and the last names in the directory respectively, and output both the full-names and the office room numbers.

2. ALGORITHM

To finish the tasks, the algorithm should be like:

1. Display the prompt string;

2. Get the name that the user has typed and store the name;

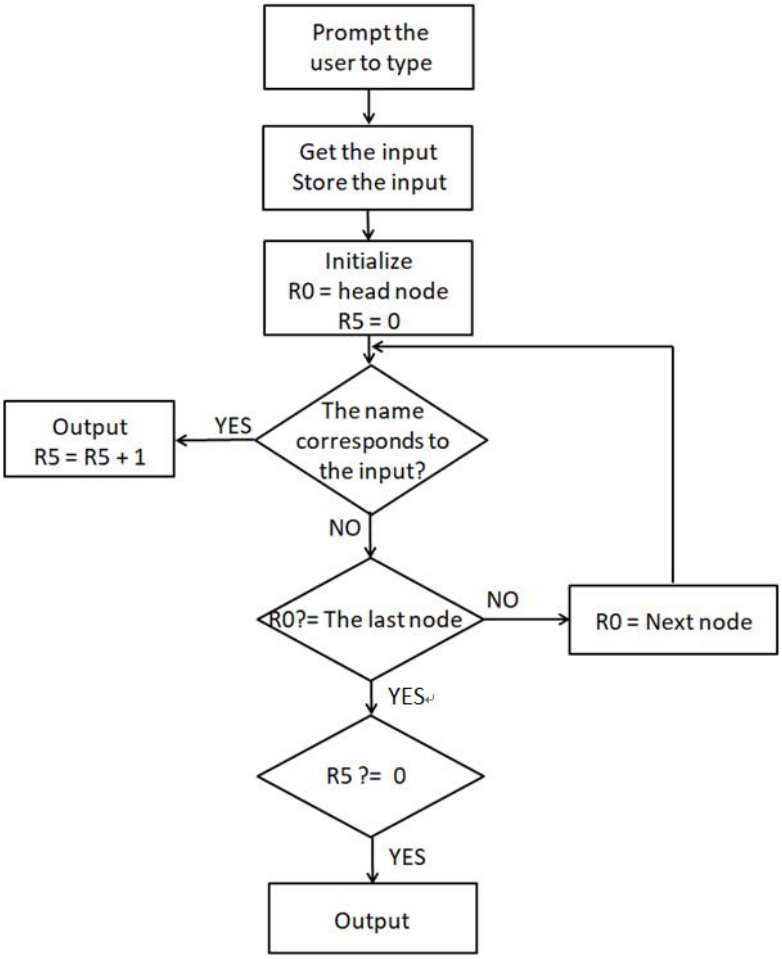
3. Compare the name with the names in the directory one by one;

4. If the information is corresponding, output the corresponding information. If not and there is next node, then go to next node.

5. If there is no corresponding information and the program has come to the last node, display “Not Found”;

R0 acts like a pointer pointing to each node. R5 indicates whether there are any outputs. R1, R2, R3 and R4 are used to compare two strings.

The diagram is shown as follow:



3. Testing Result

The test is based on the directory provided.

Input	Purpose	Expected Result	Output	Status
Yale	search by the first name, and test the first node	Yale Patt 101	Yale Patt 101	pass
Patt	search by the last name	Yale Patt 101	Yale Patt 101	pass
Lin	the last node	Circle Lin 127	Circle Lin 127	pass
(null)	no input	Not Found	Not Found	pass
P	input only one character	Not Found	Not Found	pass
Patttttttttttttt	input 15 characters	Not Found	Not Found	pass
Jiang	when there are two outputs	Jiang Xiaohong 502 Jiang Zengkai 127	Jiang Xiaohong 502 Jiang Zengkai 127	pass

Memory

x3000 | Manage Labels

0x	Label	Hex	Instruction
xFD75		xA22F	LDI R1, xFDA5
xFD76		x202F	LD R0, xFDA6
xFD77		x5040	AND R0, R1, R0
xFD78		xB02C	STI R0, xFDA5
xFD79		x2003	LD R0, xFD7D
xFD7A		x2203	LD R1, xFD7E
xFD7B		x2E03	LD R7, xFD7F
xFD7C		xC1C0	RET
xFD7D		x0000	NOP
xFD7E		x3047	ST R0, xFDC6
xFD7F		x302D	ST R0, xFDAD
xFD80		x000A	NOP

Status

Registers

R0: x7FFF **R1**: xFFFF **R2**: xFD01 **R3**: xC20E
R4: xC1C2 **R5**: x0001 **R6**: x0001 **R7**: xFD75
PC: xFD79 **IR**: xB02C **PSR**: x8001 **CC**: P

Clear R0–R7 | Reset all registers

Step | Next | Finish | Run | Pause | Continue | Unhalt

Follow PC

Console

```
Type a name and press Enter: Yale
Yale Patt 101
----- Halting the processor -----
Type a name and press Enter: Patt
Yale Patt 101
----- Halting the processor -----
Type a name and press Enter: Lin
Circle Lin 127
----- Halting the processor -----
```

Memory

x3000 | Manage Labels

0x	Label	Hex	Instruction
xFD75		xA22F	LDI R1, xFDA5
xFD76		x202F	LD R0, xFDA6
xFD77		x5040	AND R0, R1, R0
xFD78		xB02C	STI R0, xFDA5
xFD79		x2003	LD R0, xFD7D
xFD7A		x2203	LD R1, xFD7E
xFD7B		x2E03	LD R7, xFD7F
xFD7C		xC1C0	RET
xFD7D		x3077	ST R0, xFDF5
xFD7E		x3047	ST R0, xFDC6
xFD7F		x3046	ST R0, xFDC6
xFD80		x000A	NOP

Status

Registers

R0: x7FFF **R1**: xFFFF **R2**: xFD01 **R3**: xC212
R4: xC1C2 **R5**: x0000 **R6**: x0000 **R7**: xFD75
PC: xFD79 **IR**: xB02C **PSR**: x8001 **CC**: P

Clear R0–R7 | Reset all registers

Step | Next | Finish | Run | Pause | Continue | Unhalt

Follow PC

Console

```
Type a name and press Enter:
Not Found
----- Halting the processor -----
Type a name and press Enter: P
Not Found
----- Halting the processor -----
Type a name and press Enter: Patttttttttttttt
Not Found
----- Halting the processor -----
```

Memory

x3000 | Manage Labels

0x	Label	Hex	Instruction
xFD75		xA22F	LDI R1, xFDA5
xFD76		x202F	LD R0, xFDA6
xFD77		x5040	AND R0, R1, R0
xFD78		xB02C	STI R0, xFDA5
xFD79		x2003	LD R0, xFD7D
xFD7A		x2203	LD R1, xFD7E
xFD7B		x2E03	LD R7, xFD7F
xFD7C		xC1C0	RET
xFD7D		x0000	NOP
xFD7E		x3047	ST R0, xFDC6
xFD7F		x302D	ST R0, xFDAD

Status

Registers

R0: x7FFF **R1**: xFFFF **R2**: xFD01 **R3**: xC20C
R4: xC1C2 **R5**: x0002 **R6**: x0002 **R7**: xFD75
PC: xFD79 **IR**: xB02C **PSR**: x8001 **CC**: P

Clear R0–R7 | Reset all registers

Step | Next | Finish | Run | Pause | Continue | Unhalt

Follow PC

Console

```
Type a name and press Enter: Jiang
Jiang Xiaohong 502
Jiang Zengkai 127
----- Halting the processor -----
```

4. Discussion and Experience

At first I found my program could not output "Yale Patt 101" correctly. By looking at the information stored at x4000, I found that it is because there is a dummy head.

When checking my program, I found that if I input 16 characters (which is an illegal input), my program would output nothing after that. By checking the value one by one, I found that the value representing the head of the linked list had been cleared. Although the lab doesn't require to process such input, I think I shall prevent it from making a bug. I can just simply leave a place for the excess next time.

Appendix: Source Code

```
        .ORIG   x3000
; diaplay the prompt string
        LEA     R0, PROM    ; prompt string
        TRAP    x22
; read the string inputted
        LEA     R1, BUFF
READ    TRAP    x20
        ADD     R2, R0, #-10
        BRz     EREAD
        TRAP    x21
        STR     R0, R1, #0
        ADD     R1, R1, #1
        BRnzp   READ
EREAD   STR     R2, R1, #0  ; store 0 to end the string
; search the corresponding information
        AND     R5, R5, #0  ; signals the output
        LD      R0, SENT    ; pointer to the head
SEAR    LDR     R0, R0, #0
        LEA     R1, BUFF    ; store the input
        ADD     R2, R0, #2
        LDR     R2, R2, #0  ; the first name
CMPAF   LDR     R3, R1, #0  ; check the first name
        LDR     R4, R2, #0
        ADD     R6, R3, R4  ; check if both are null
        BRz     FOUND
        ADD     R1, R1, #1
        ADD     R2, R2, #1
        NOT     R4, R4
        ADD     R4, R4, #1
        ADD     R3, R3, R4  ; compare the character
```

```
        BRz     CMPAF
        ADD     R2, R0, #3
        LDR     R2, R2, #0  ; the last name
        LEA     R1, BUFF
CMPAL   LDR     R3, R1, #0  ; check the last name
        LDR     R4, R2, #0
        ADD     R6, R3, R4  ; check if both are null
        BRz     FOUND
        ADD     R1, R1, #1
        ADD     R2, R2, #1
        NOT     R4, R4
        ADD     R4, R4, #1
        ADD     R3, R3, R4  ; compare the character
        BRz     CMPAL
DIV     ADD     R6, R0, R5
        BRz     NFOUND      ; the last node and no results
        ADD     R0, R0, #0
        BRp     SEAR        ; there is next node
        TRAP    x25         ; the last node and outputs
; output
FOUND   LD      R1, SPACE   ; output the corresponding information
        ADD     R2, R0, #0
        LD      R0, ENTER
        TRAP    x21
        ADD     R0, R2, #2
        LDR     R0, R0, #0
        TRAP    x22
        ADD     R0, R1, #0
        TRAP    x21
        ADD     R0, R2, #3
        LDR     R0, R0, #0
        TRAP    x22
        ADD     R0, R1, #0
        TRAP    x21
        ADD     R0, R2, #1
        LDR     R0, R0, #0
        TRAP    x22
        ADD     R5, R5, #1  ; change R5
        ADD     R0, R2, #0  ; restore R0
        BRnzp   DIV
NFOUND  LD      R0, ENTER
        TRAP    x21
        LEA     R0, NMATCH  ; output "Not Found"
        TRAP    x22
```

```
        TRAP    x25
BUFF    .BLKW   x10
SENT    .FILL   x4000
ENTER   .FILL   x000A
SPACE   .FILL   x0020
PROM    .STRINGZ        "Type a name and press Enter: "
NMATCH  .STRINGZ        "Not Found"
        .END
```