LAB 5

1. INTRODUCTION

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

According to the requirements, the user will input the information and the name, then the program should compare the name with the input information, and output both the full-names and the office room numbers.

2. ALGORITHM

To finish the tasks, the algorithm should be like:

1. Get the input;

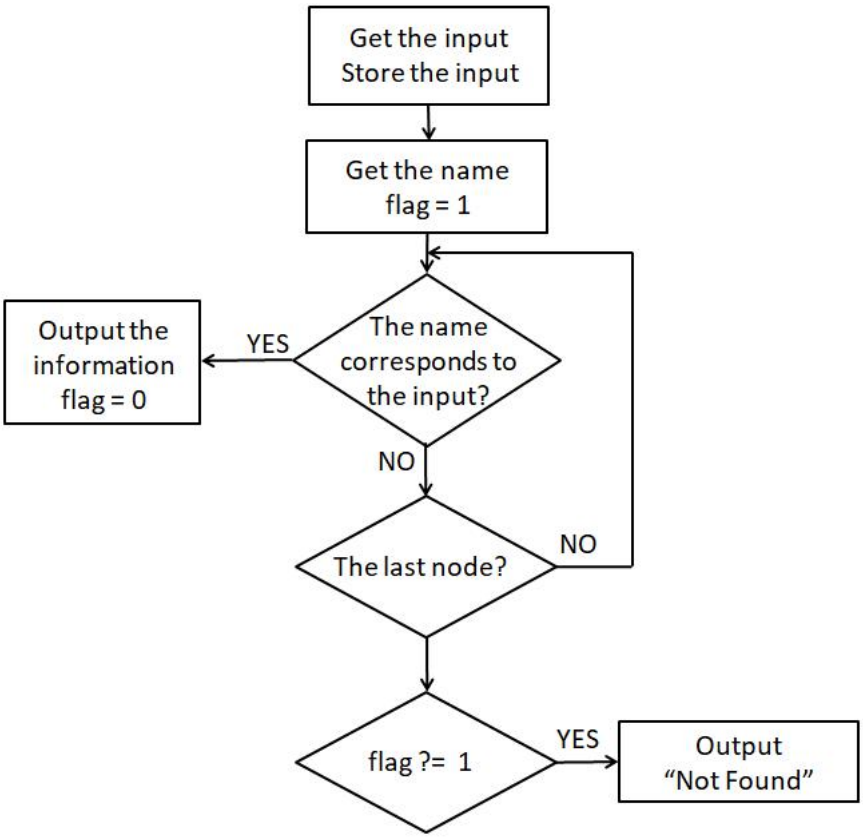
2. Create a linked list to store the information;

3. Search for the corresponding information;

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

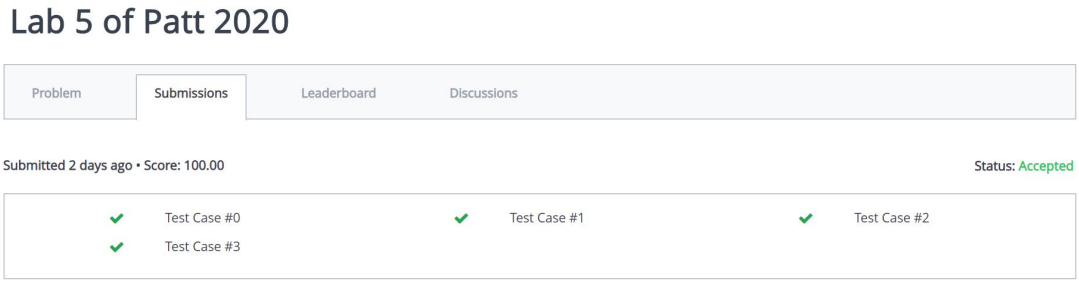
5. If there is no corresponding information and the program has come to the last node, display “Not Found”;

The diagram is shown as follow:



3. Testing Result

The program passes all the test cases on HackerRank.



4. Discussion and Experience

Similar to the linked list in Lab 2, the linked list has a dummy head, which makes the iterative simpler.

I deeply agree that a high-level language is much more preferable than an assembly language.

APPENDIX: SOURCE CODE

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
typedef struct Node
{
   char first[20];
   char last[20];
   char room[20];
   struct Node* next;
}Node;     /*definition of a node*/

int main()
{
   int n, i=0, flag=1;
   char input[20];
   Node* head=(Node* )malloc(sizeof(Node));
   Node* ptr=head;
   scanf("%d", &n);
   for (i=0; i<n; i++)    /*get the input*/
   {
      ptr->next=(Node* )malloc(sizeof(Node));
      ptr=ptr->next;
      scanf("%s", ptr->first);
      scanf("%s", ptr->last);
      scanf("%s\n", ptr->room);
      ptr->next=NULL;
   }
   scanf("%s", input);
   ptr=head;
   for (i=0; i<n; i++)    /*compare*/
   {
      ptr=ptr->next;
      if (strcmp(input, ptr->first)==0||strcmp(input, ptr->last)==0)
      {
         printf("%s %s %s\n", ptr->first, ptr->last, ptr->room);
         flag=0;
      }
   }
```

```
   if (flag)
   {
      printf("Not Found\n");
   }
   return 0;
}
```