**详细设计文件**

**姓名：李华**

**学号：2020290005**

**文件及函数组成**

* main.c：main(); //主函数
* elevator.h (头文件)
* input1.c:

void getInput1() （gets input one at a time）

* input2.c

void getInput2()

void append(NODE\*\* head\_ref, int new\_data)

* getInputFILE.C

// gets input from the file

void getInput2FILE(); // 为了选择a

void getInput1FILE() // 为了选择b

* control1.c：

void control1(); //确定当前目标楼层及输出（顺便服务）

int findup(int k);

int finddown(int k);

void time\_count();

* state\_trans.c

void state\_trans();

* control2.c

void control2(); //确定当前目标楼层及输出（先来先服务）

void removeDuplicates(NODEPTR headPtr);

void print\_list(NODEPTR Ptr);

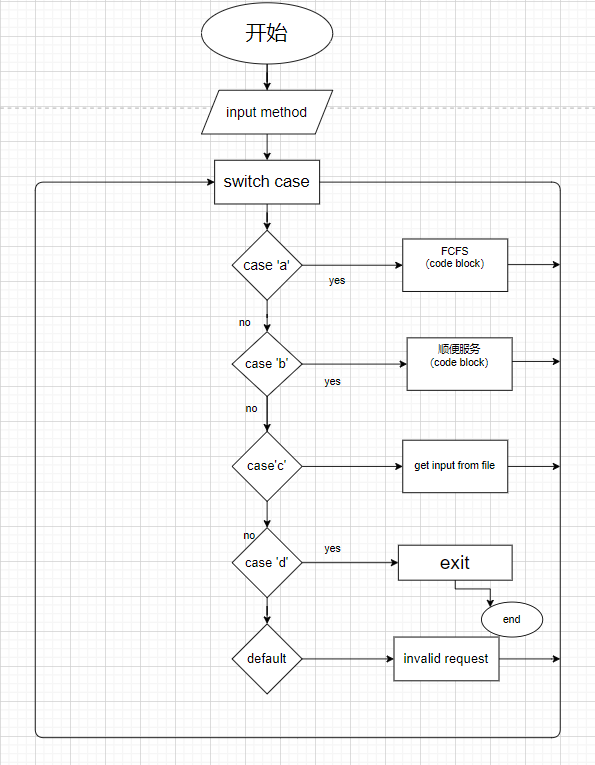
void free\_list(NODEPTR hPtr)

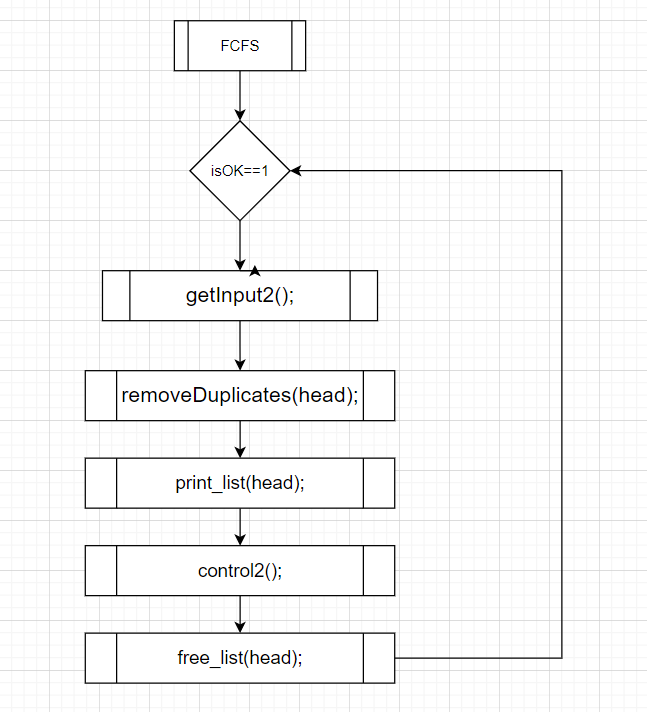
* print\_message.c

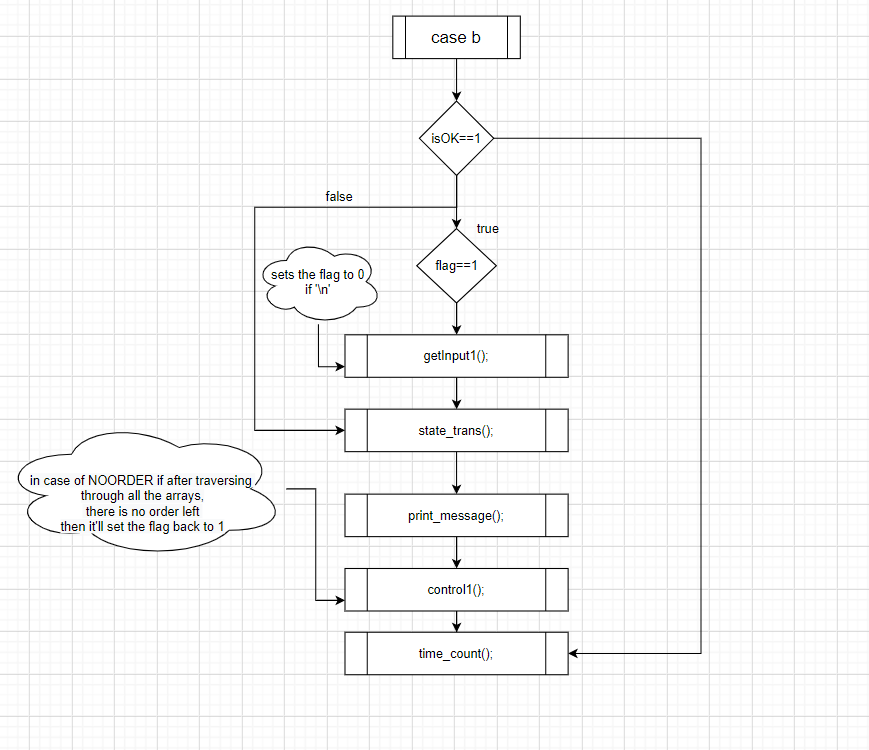
void print\_message(void) // displays output

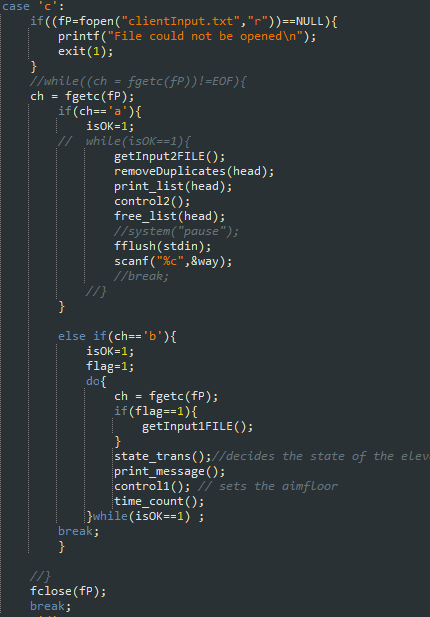
* client2.dat
* clientInput.txt

**Flowchart**



****

****

****

**头文件**

#ifndef TEST\_H

#define TEST\_H

#include<stdio.h>

#include<stdlib.h>

#include<time.h>

#define MAX\_FLOOR 9

#define UP 1

#define DOWN -1

#define NOORDER 0

#define PAUSE 1

#define RUNNING 2

#define STOP 3

struct requestNode{

int floor;

struct requestNode \*nextPtr;

};

typedef struct requestNode NODE;

typedef NODE \*NODEPTR;

extern NODEPTR head; // head pointer to the request list

extern int isOK;

extern char way;

extern int presentfloor;

extern int aimfloor;

extern int flag;

// fcfs

// functions definations

void getInput2(); // gets the input

void control2(); // runs the elevator and print the output to the user

void append(NODE\*\* head\_ref, int new\_data); // apends new node to the list

void removeDuplicates(NODEPTR headPtr); // removes the dublicate nodes from the list

void print\_list(NODEPTR Ptr); // prints the final request list

void free\_list(NODEPTR hPtr); // frees the list

// for shun bian fu wu

extern int state;

extern int direction;

extern int inSide[10];

extern int up[10];

extern int down[10];

// function definations

void getInput1(void); // gets the request

void control1(void); // sets the aim floor for shun bian fu wu

void print\_message(void); //displays the out put

int findup(int k);

int finddown(int k);

void state\_trans(void); // desides the state and present floor

void time\_count(void); // to pause for each floor

void print\_request(int arr[]);

// for input from file

void getInput2FILE(); // gets the input from the file

extern FILE \*fP;

extern char ch; // TO READ CHARACTER FROM THE FILE

void getInput1FILE(); // for option b for input from the file

#endif

**全局变量:**

NODEPTR head=NULL;

int isOK=1;

char way;

int presentfloor = 1;

int aimfloor = 0;

int flag = 1;

int state = PAUSE;

int direction = NOORDER;

// for second algorithm

int inSide[10] = {0};

int up[10] = {0};

int down[10] = {0};

FILE \*fP;

char ch; // TO READ CHARACTER FROM THE FILE

**数据结构的定义**

* **FCFS**

struct requestNode{

int floor;

struct requestNode \*nextPtr;

};

typedef struct requestNode NODE;

typedef NODE \*NODEPTR;

* **顺便服务**

int inSide[10] = {0};

int up[10] = {0};

int down[10] = {0};

Functions（顺便服务）：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 参数 | 返回值 | 函数功能 |
| **void getInput1(void);** | Void | Void | Gets input if the second option is selected. |
| **void control1(void);** | Void | Void | Sets the aim floor.  if no orders are left then sets the flag to 0 in order to take the input. |
| **void print\_message(void);** | void | void | Displays the output according to the state aim floor and present floor. |
| **int findup(int k);** | integer | integer | Gets the next up request if the direction is UP from the present floor |
| **int finddown(int k);** | integer | integer | Gets the next down request if the direction is DOWN from the present floor |
| **void state\_trans(void);** | void | void | Decides the state and resets the orders that has already been served. |
| **void time\_count(void);** | void | void | To pause the elevator after every floor |
| **void print\_request(int arr[]);** | Array of integer | void | Prints the array to check if the orders are being placed. |
|  |  |  |  |

**FCFS:**

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 参数 | 返回值 | 函数功能 |
| **void getInput2();** | void | void | Gets the input all at once and calls the function append every time it take input. |
| **void control2();** | void | void | Sets the aim floor, and traverse through every floor until the destination floor is reached |
| **void append(NODE\*\* head\_ref, int new\_data);** | Pointer to the head pointeer of the list and an integer | void | Appends the new node to the list. |
| **void removeDuplicates(NODEPTR headPtr);** | Head pointer | void | Removes simultaneously occurring same nodes from the list. |
| **void print\_list(NODEPTR Ptr);** | Head pointer to the list | void | Prints the final request list (just to check the final list ) |
| **void free\_list(NODEPTR hPtr);** | Head pointer to the list | void | Frees the list after the elevator has served all the requests |

**Input from file:**

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 参数 | 返回值 | 函数功能 |
| **void getInput2FILE();** | void | void | If the option is a, then it pushes the data into the list. |
| **void getInput1FILE();** | void | void |  |