

# 2014211319-I 小组-模拟银行系统-详细设计报告

## ( 命令行 )

版本号：I-Design-20150530

编制时间：2015-05-30

编制人员：刘含，祝翎桓，范子晗

### —模块 1<main.c>

#### ➤ 函数 1

■ 函数名称：int main()

■ 函数参数说明：无

■ 函数功能：统一管理各个子模块，实现系统框架

■ 函数局部变量设计：无

■ 函数全局变量设计：char ch;//保存读入的字符

tongji sta[8];//保存每个窗口统计结果

time\_t cstart,start[8],end[8];//end[8]保存每个客户服务时长

int MaxSeqLen=0,MinSec=0,MaxSec=0,MinRestSec=0,MaxRestSec=0,

VIPSERVLen=0;//保存从配置文件中读入的信息

int vis[]={1,0,1,1,1,0,0,0}; //保存当前窗口"上班"或"暂停"状态

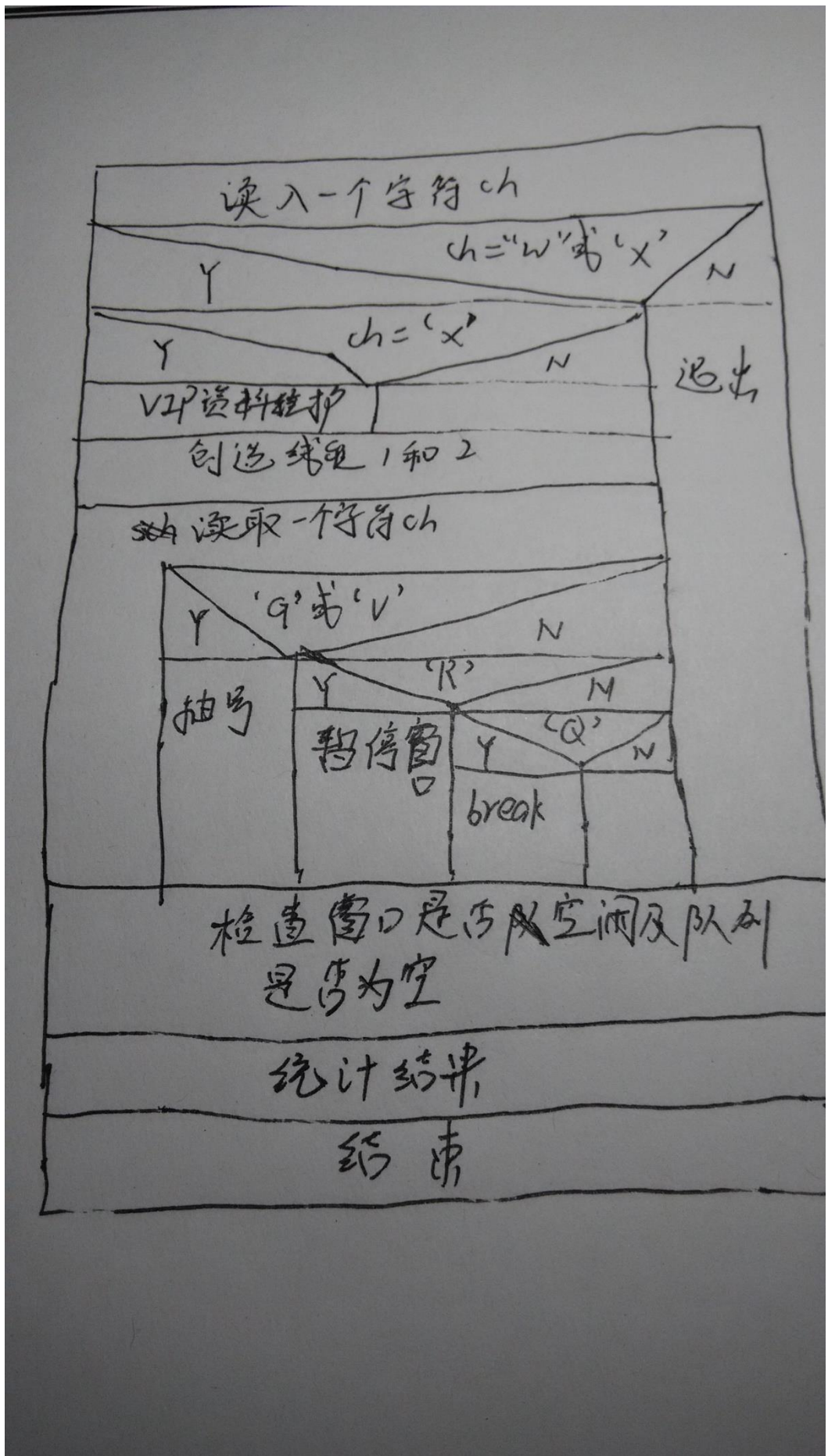
int cur[]={-1,-1,-1,-1,-1,-1,-1,-1};//保存当前窗口服务客户编号

int

maxnumber=0,counts=0,vmaxnumber=0,vcounts=0,current=0,vcurrent=0,is

wait=0,flag=1,tactic=1;//maxnumber 保存最大普通客户 , counts 保存普通客户排队人数 ,vcounts 保存 vip 客户排队人数 ,current 保存当前普通队首客户编号 , vcurrent 保存 vip 队首客户编号 , iswait 保存当前是否等待下班状态 , tactic 保存当前实行的是 vip 窗口顺便服务策略还是 vip 窗口定向服务策略。

■ 函数算法逻辑：



➤ 函数 2

■ 函数名称：reprint

■ 函数参数说明：无

■ 函数功能：调用 print.c 模块每隔 10s 打印一次银行运行情况。

■ 函数局部变量设计：无

■ 函数算法逻辑：

每隔 10s 调用一次 print 模块打印当前情况。

—模块 2 <cconfig.c>

➤ 函数 1

■ 函数名称：int cconfig()

■ 函数参数说明：无

■ 函数功能：实现从 config.ini 文件中读取相关参数

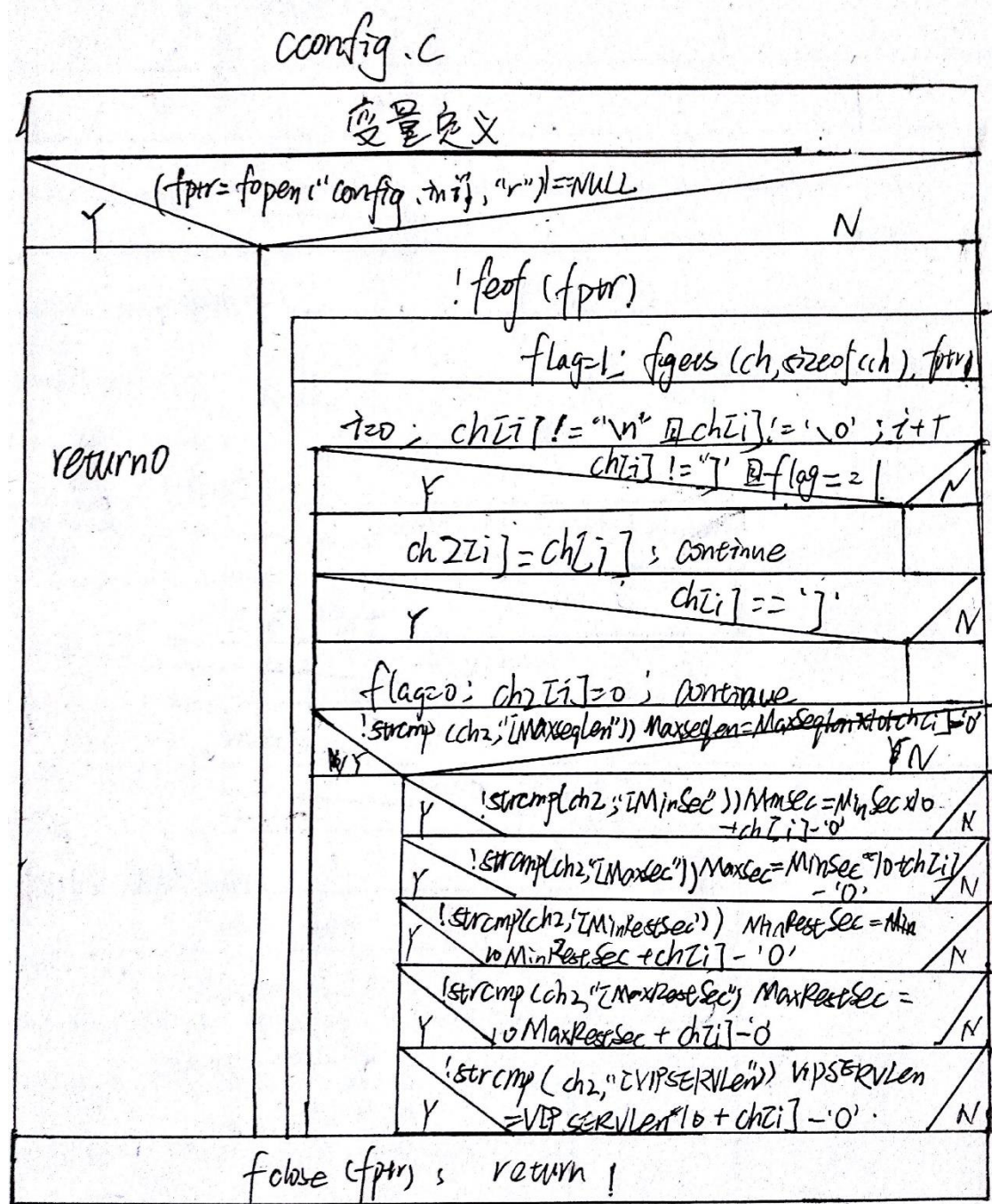
■ 函数外部变量设计：extern int

MaxSeqLen, Minsec, Maxsec, MaxRecSec, MinRestSec, VIPSERVLen

■ 函数局部变量设计：int i, flag=1; FILE \*fptr;

char ch[20], ch2[20];

■ 函数算法逻辑设计：



—模块 3 <vip.c>

➤ 函数 1

■ 函数名称：int vip()

■ 函数参数说明：无

■ 函数功能：银行营业之前更新 vip 客户资料

■ 函数局部变量设计：FILE \*fptr; int i; char c,d;

customer a,temp[10000],b;//保存从文件中读取出来的结构体和需要写入文件的  
结构体

■ 函数算法逻辑：

```
int vip()
{
    打印提示信息;
    while(读取字符 c 字符不为 E){
        读取字符 d;
        打开文件"vip.dat"
        if 文件打开失败{
            打印"File can not open.\n";
            返回 0;
        }
        if (字符 d 为'+'){
            打印"Please input the customer`s name,ID and deposit you want to add:";
            从键盘读取 a.name, a.ID, a.account;
            将指针定位文件末尾;
            写入用户信息;
            关闭文件;
            打印"-----written successfully!-----";
        }
        else if (字符 d 为'-'){
            打印"Please input the customer`s name you want to delete:";
            从键盘读取用户名 a.name;
            将文件指针指向文件头;
            查找并删除用户;
            关闭文件;
            打印("-----delete successfully!-----\n\n");
        }
        else if(字符 d 为'/'){
            打印"Please input the customer`s name you want to update:";
            从键盘读取用户名 a.name;
            打印"Please input the new ID and new deposit:";
            从文件读取用户 ID a.ID,余额 a.account);
            将文件指针指向文件头;
            查找并更新用户资料;
            关闭文件;
        }
        打印"-----update successfully!-----";
    }
}
```

```

    }
    else 打印"-----illegal input!-----Please input again: ";
    }
    打印"-----VIP data update finished.-----";
    返回 1;
}

```

## —模块 4<pnumber.c>

### ➤ 函数 1

■ 函数名称：void pnumber()

■ 函数参数说明：无

■ 函数功能：实现新客户到达抽号服务

■ 函数外部变量设计：extern int maxnumber, counts, vmaxnumber,

vcounts, current, vcurrent;//排队的队列相关参数

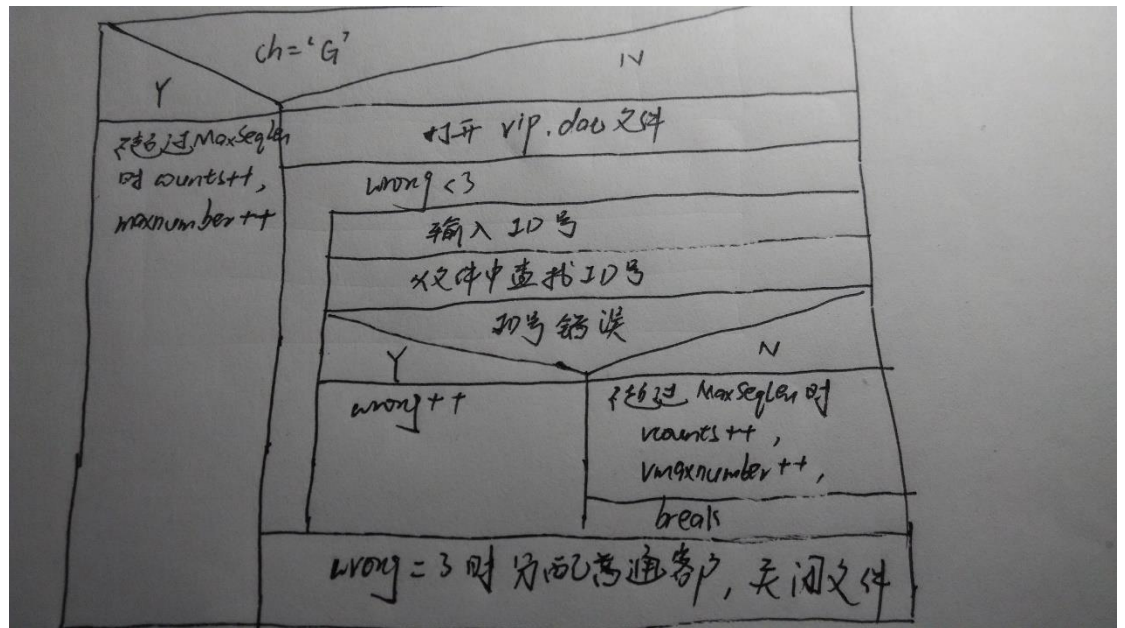
extern char ch;//之前读取的字符

■ 函数局部变量设计：FILE \*fptr; customer a;

int wrong=0;//vip 客户 id 输入错误次数

■ 函数算法逻辑：





## —模块 5 <print.c>

### ➤ 函数 1

■ 函数名称：void print()

■ 函数参数说明：无

■ 函数功能：打印当前银行运行情况

■ 函数外部变量设计：extern int vis[8], cur[8]; extern int maxnumber, counts, vmaxnumber, vcounts, current, vcurrent, iswait, tactic;

■ 函数局部变量设计：int i;

■ 函数算法逻辑：

```

void print()//
{
    打印"-----The bank's business information-----";
    打印普通客户人数和 VIP 客户人数;
    打印"The maxnumber of customer is:          ";
    if (maxnumber != -1)
        打印普通客户编号;
    else 打印"common:NULL";
    if (vmaxnumber != -1)
        打印 VIP 客户编号;
}
  
```



```

else 打印"VIP:NULL";
打印"The state of all windows is:";
打印窗口名;
打印"State:  ";
for(i←0;i<8;i←i+1)
{
    if(cur[i]≠-1||vis[i]=1) 打印"work";
    else if(vis[i]=-1) 打印"pause";
    else 打印"close\t";
}
打印"current: ";
for(i←0;i<8;i←i+1)
{
    if(cur[i]≠-1){
        打印当前窗口客户编号;
    }
    else 打印"NULL";
}
打印"Is waiting for closed?";
if(iswait)
打印"Yes.";
else 打印"No.";
fileprint();
}

void fileprint()
{
    打开文件
    int i;
    time_t timer;
    time(&timer);
    在文件中写入"-----The bank's business log----- ";
    在文件中写入当前时间;
    在文件中写入当前普通客户和 VIP 客户数量;
    在文件中写入最大客户数量;
    if(maxnumber≠-1) 在文件中写入普通客户编号;
    else 在文件中写入"common:NULL";
    if(vmaxnumber≠-1) 在文件中写入 VIP 用户编号;
    else 在文件中写入"VIP:NULL";
    在文件中写入窗口状态;
for(i=0;i<8;i++)
{
    if(cur[i]≠-1||vis[i]=1) 在文件中写入"work";
    else if(vis[i]=-1) 在文件中写入"pause";
}
}

```

```

        else fprintf(fptr,"close\t");
    }
    在文件中写入"current: ";
    for(i←0;i<8;i←i+1)
    {
        if(cur[i]≠-1){
            在文件中写入当前窗口客户编号;
        }
        else 在文件中写入"NULL";
    }
    在文件中写入"Is waiting for closed?";
    if(iswait) 在文件中写入"Yes.";
    else 在文件中写入"No.";
    关闭文件;
}
}

```

## —模块 6<check.c>

### ➤ 函数 1

■ 函数名称：void check()

■ 函数参数说明：

■ 函数功能：动态调整营业窗口数量，并修改实行策略。

■ 函数外部变量设计：extern int

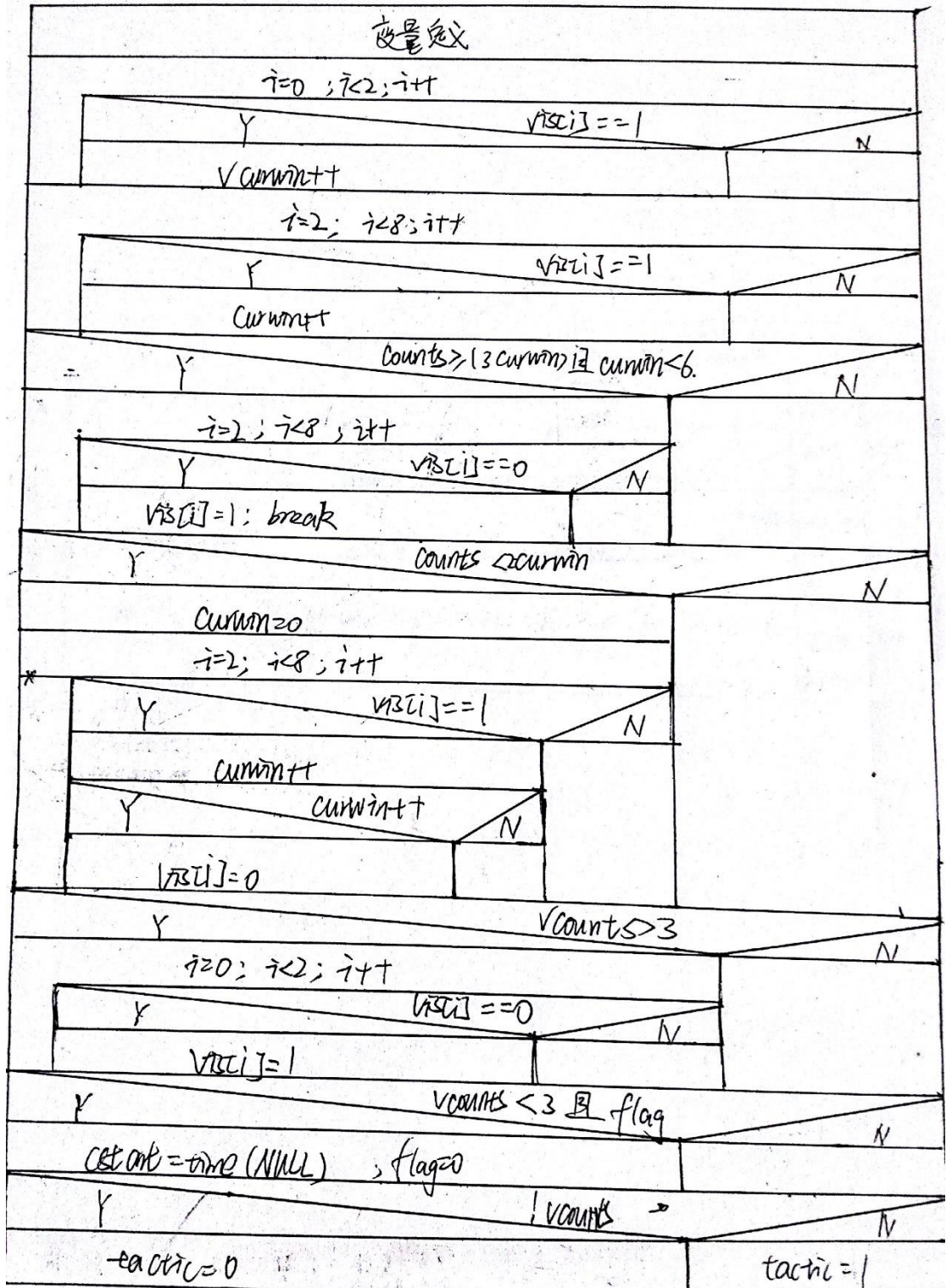
maxnumber,vmaxnumber,counts,vcounts,current,vcurrent,flag,cstart;

extern int vis[8],cur[8],tactic;

■ 函数局部变量设计：int curwin=0,vcurwin=0,i;//保存当前营业窗口数量

■ 函数算法逻辑：

check.c



## —模块 7<work.c>

### ➤ 函数 1

■ 函数名称：void work()

■ 函数参数说明：无

■ 函数功能：调用 check.c 动态调整服务窗口并根据不同服务策略进行叫号办理业务，并且在办理业务后统计个窗口办理业务人数和办理业务总时长。

■ 函数外部变量设计：extern int

maxnumber,vmaxnumber,counts,vcounts,current,vcurrent,iswait,VIPSERVLen,fl

ag,tactic;

extern int MaxSeqLen,MinSec,MaxSec,MinRestSec,MaxRestSec,VIPSERVLen;

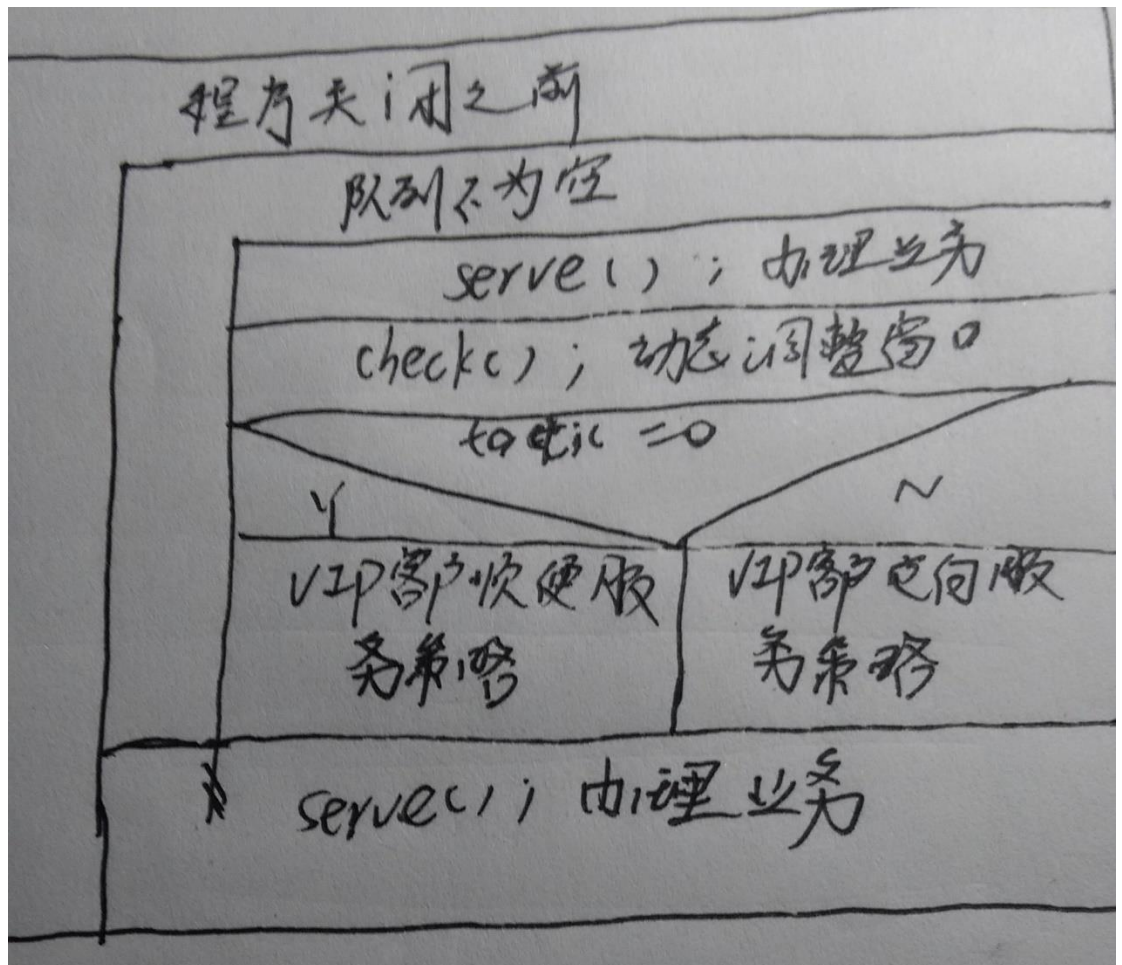
extern int vis[8],cur[8];

extern tongji sta[8];

extern time\_t cstart,start[8],end[8];

■ 函数局部变量设计：time\_t curtime;//保存当前程序运行时间

■ 函数算法逻辑：



## —模块 8<statices.c>

### ➤ 函数 1

- 函数名称：void statices()
- 函数参数说明：无
- 函数功能：实现程序结束时将统计结果写入指定文件中
- 函数外部变量设计：extern tongji sta[8];
- 函数局部变量设计：FILE \*fptr
- 函数算法逻辑：

```

void statices()
{
    打开文件;
    if (文件打开失败)

```

```

        打印("File could not be opened.");
    else {
        在文件中写入时间;
        在文件中写入窗口号、总客户数、总时长、平均时长;
    }
    在文件中写入普通客户总数;
    在文件中写入 VIP 用户总数;
    关闭文件;
}
}

```

## —模块 9<pause.c>

### ➤ 函数 1

■ 函数名称：void pause()

■ 函数参数说明：无

■ 函数功能：实现窗口暂停一定时间

■ 函数外部变量设计: extern time\_t rest[8],rstart[8];

extern int vis[8],MinRestSec,MaxRestSec;//time\_t 类型变量表示存储休息时及开始休息时间，vis[]表示当前窗口状态；

■ 函数局部变量设计：int curwin=0,vcurwin=0,i,restnum;curwin 表示窗口数量，restnum 为申请休息窗口号；

■ 函数算法逻辑：



pause.C.

Y	i=0; i<2; i++ v73[i]=1 vcurwin++	N
Y	i=2; i<8; i++ v73[i]=2 curwin++	N
Y	scanf ("%d" &resnum); resnum>=1 & resnum<=2 & vcurwin==2    resnum>=3 & vcurwin>3	N
Y	cur[resnum-1] != 1 & <del>v73[resnum-1] != 2</del>	N
v73[resnum-1] < 2	v73[resnum-1] = -1; start[resnum-1] = time(NULL);	printf("Sorry, the current window is too less.
res[resnum-1] = (rand() % (MaxResec - MinResec + 1)) / 100;	MinResec MaxResec	too less.
printf (" The number %d window will rise for %d seconds")	Please application later"	