#ifndef HEAD1\_H

#define HEAD1\_H

#define MAX 40     //用于限制字符数组大小的常量 #define N 4        //普通服务柜台数

/\*队列结点\*/ typedef struct qq {

 int n;         //用户取号  int service;   //对应的服务窗口  struct qq \*next;//指针域 }QNode;

/\*队列\*/ typedef struct Q {

 QNode \*front;  //头指针  QNode \*rear;   //尾指针 }Queue;

/\*全局变量\*/

static Queue \*q;           //队列 static int people;         //总排队人数 static int current\_people; //当前排队人数 static int common;         //普通用户人数 static int vip;            //VIP用户办理人数

/\*-----------------------队列操作的定义---------------------\*/

void Initial();                  //初始化队列 int Is\_Empty();                  //队列是否为空 void In\_Queue(QNode \* &s);       //入队 void Out\_Queue(QNode \* &t);      //出队

/\*------------------------函数模块声明----------------------\*/

void System();                          //主体函数

void Get\_Time(struct tm \* &t);          //获取时间

char Get\_Choice(char,char);             //获取正确的选择

char Start();                           //开始界面同时返回用户选择

int Is\_Vip(char [],char []);            //判断是不是vip客户

void Check\_Queue();                     //查看排队情况(针对于用户)

void Grade(int);                        //评分

void Common\_User(int,char [],char []);  //普通用户界面

void Vip\_User();                        //VIP用户界面

void User();                            //用户主界面

.h> #include<windows.h> #include"Head.h"

/\*-----------------------------------队列操作-----------------------------------\*/

void Initial()               //初始化队列 {

 q=(Queue \*)malloc(sizeof(Queue));  q->front = q->rear = NULL; }

int Is\_Empty()               //队列是否为空 {

 return(q->rear == NULL); }

void In\_Queue(QNode \* &s)    //入队 {

 QNode \* p= q->front;

 s=(QNode \*)malloc(sizeof(QNode));

 while(p)   p=p->next;

 people++;

 s->n = people-vip;              //获取排队编号

 if(Is\_Empty())   q->front = q->rear = s;  else

 {   q->rear->next = s;   q->rear = s;

 }

 s->next = NULL;  common++;  current\_people++; }

void Out\_Queue(QNode \* &t)//出队 {

 if(Is\_Empty())  {   printf("\t\t暂时没人排队,因此无法提供相应信息^\_^!\n");   return;

 }

 t = q->front;

 if(q->front == q->rear)   q->front = q->rear =NULL;  else

  q->front = t->next;  current\_people--;

}

/\*-------------------------------------主体函数----------------------------------\*/

void System()

{  while(1)  {

 system("cls");

 system("color 2f");

  char ch=Start();

 switch(ch)   {

case '1':

   User();    break;

  case '2':

   system("cls");

   printf("\t--------------------------------------------------------------\n"

"\t                         中国银行                         \n"

"\t--------------------------------------------------------------\n\n\n");

   printf("欢迎下次光临,再见^\_^\n\n");    system("pause");

  }

   if(ch == '2')

break;

system("pause");

 } }

/\*------------------------------开始界面&&获取用户的选择---------------------------\*/ char Start() {

 struct tm \*t;  //时间结构  char ch;       //选择

 Get\_Time(t);   //获取系统时间

 system("color 2f");

 printf("      \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/欢迎光临中国银行\\\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"

"      \*                                                                  \*\n"

 "      \*                                                                  \*\n"

"      \*                                                                  \*\n"

 "      \*                                                                  \*\n"

"      \*                         1  客户                                 \*\n"

  "      \*                                                                  \*\n"

 "      \*                                                                  \*\n"

 "      \*                         2  退出                                 \*\n"

 "      \*                                                                  \*\n"

 "      \*                                                                  \*\n"

 "      \*                                                     %2d:%2d:%2d     \*\n"

 "      \*                                                    %4d/%2d/%2d    \*\n"

 "      \*                                                                  \*\n"

 "      \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/欢迎光临中国银行

\\\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n\n",

  t->tm\_hour,t->tm\_min,t->tm\_sec,t->tm\_year + 1990,t->tm\_mon + 1,t->tm\_wday);

 printf("\t\t尊敬的用户,");

 if(t->tm\_hour < 12)   printf("上午好^\_^\n");

 else if(t->tm\_hour < 18)   printf("下午好^\_^\n");   else

printf("晚上好^\_^\n");

 printf("\t\t请输入您的选择(1~2):");  ch=Get\_Choice('1','2');

 return ch;           //返回选择 }

/\*------------------------------------用户界面------------------------------------\*/

void User() {

 char ch;          //用户的菜单选择

 while(1)  {   system("cls");   system("color 3f");

   printf("\t--------------------------------------------------------------\n"    "\t                     欢迎光临中国银行                         \n"    "\t--------------------------------------------------------------\n\n\n");   printf("\t\t\t\t1  取号排队\n"

   "\t\t\t\t2  查看排队情况\n"    "\t\t\t\t3  叫号情况\n"

   "\t\t\t\t4  办理业务(VIP可以直接办理业务^\_^)\n"

"\t\t\t\t5  返回主菜单\n\n\n");   printf("\t\t输入您的选择:");   ch=Get\_Choice('1','5');

   switch(ch)   {   case '1':    {     FILE \*fp;

char card\_num[MAX];

     if((fp=fopen("vip.txt","r"))==NULL)     {      printf("\t\tvip.txt文档操作失败!  :(\n");      exit(1);

}

     printf("\t\t您是VIP用户吗?(Y/N)\n"

"\t\tVIP用用户的账号有:\n");

    while(fgets(card\_num,MAX,fp)!=NULL)     {      int i=0;      printf("\t\t");      for(;i<6;i++)

putchar(card\_num[i]);      putchar('\n');

    }

     printf("\t\t");     if(getchar()=='Y')     {      printf("\t\t好的,您可以直接去办理业务,无需排队了^\_^\n");      break;     }

     QNode \*s;     In\_Queue(s);

    printf("\t取号成功,您的排队编号为:%d(温馨提醒:请记住编号,这是排队的依据^\_^)\n",s->n);

    break;

   }   case '2':

   {     Check\_Queue();//查看排队情况     break;

   }

  case '3':

   {                 //查看叫号情况     if(Is\_Empty())     {      printf("\t\t暂时没人排队,因此无法提供相应信息^\_^!\n");      break;

    }

    printf("\t\t排队序号为%d的用户可以办理业务了^\_^\n",q->front->n);     break;

   }   case '4':

   {     char c\_n[MAX];  //用户输入的账号     char c[MAX];    //用户输入的密码     int is\_vip;     //判断用户是不是VIP

is\_vip=Is\_Vip(c\_n,c);

     switch(is\_vip)     {     case 0:      {

QNode \*t;

       if(Is\_Empty())       {        printf("\t\t暂时没人排队,因此无法提供相应信息^\_^!\n");        break;

}

       printf("\t\t您的排队序号是%d吗?\n"        "(本着中华民族的诚信优良传统,请不要插别人的队^\_^)\n"        "(输入Y代表'是',输入N或其它代表'不是'):",q->front->n);       if(getchar() != 'Y')

break;

system("pause");

Out\_Queue(t);

       t->service = (t->n - 1)%N + 1;//获取普通客服柜台       Common\_User(t->service,c\_n,c);       free(t);       break;

     }     case 1:

     {       printf("\t\t您是VIP用户,可以直接到VIP客户区办理业务!\n");       system("pause");       people++;       current\_people++;       vip++;       Vip\_User();       break;

     }

    }     break;

   }   case '5':

   break;

  }

  if(ch == '5')

break;   system("pause");

 } }

/\*---------------------------------普通用户界面-------------------------------\*/ void Common\_User(int s,char c\_n[],char c[]) {  char ch;

 time\_t start=time(NULL);//获取用户到达时间  while(1)  {   int money;   system("cls");   system("color 5f");

  printf("\t==================尊敬的用户你好,%d号柜台为您服务==================\n\n\n"

   "\t\t\t\t1  取钱\n"    "\t\t\t\t2  存钱\n"    "\t\t\t\t3  成为VIP\n"

"\t\t\t\t4  退出\n\n",s);

  printf("\t\t请输入您的选择:");   ch=Get\_Choice('1','4');

   switch(ch)   {   case '1':    printf("\t\t请输入您想取出的金额数(￥):");    scanf("%d",&money);    Sleep(3000);

   printf("\n\t\t您的%d元已经取出,请好好保存^\_^\n",money);    break;

  case '2':

   printf("\t\t请输入您想存入的金额数(￥):");    scanf("%d",&money);    Sleep(3000);

   printf("\n\t\t您的%d元已经成功储存到您的账户上^\_^\n",money);    break;   case '3':

   {     FILE \*fp;

char Ch[MAX];

     if((fp=fopen("vip.txt","a+"))==NULL)     {      printf("\t\t数据库打开失败:( !\n");      exit(1);

}

     while(fgets(Ch,MAX,fp)!=NULL)

continue;

    fprintf(fp,"%s %s\n",c\_n,c);     Sleep(3000);

    printf("\n\t\t恭喜您,成为了VIP(以后就可以直接到VIP窗口办理业务了^\_^)\n");     fclose(fp);     break;

   }   case '4':

   break;

  }

   if(ch == '4')   {

time\_t end;        //用户离开时间

Grade(s); //评分     end=time(NULL);    //获取用户离开时间

   printf("您总共用时%.0f秒\n",difftime(end,start));    break;   }

  system("pause");

 } }

/\*---------------------------------VIP用户界面----------------------------------\*/ void Vip\_User() {  char ch;  int money;

 time\_t start=time(NULL);//VIP用户到达时间  while(1)  {   system("cls");   system("color 4f");

printf("\t\t==================尊敬的VIP用户您好:)====================\n\n\n"

   "请问有什么可以能帮助您的呢?\n"    "(以下是业务菜单^\_^)\n"    "\t\t\t\t1  取钱\n"    "\t\t\t\t2  存钱\n"    "\t\t\t\t3  退出\n"

"\t\t输入您想办理的业务所对应的编号:");

  ch=Get\_Choice('1','3');    switch(ch)   {   case '1':    printf("\t\t请输入您想取出的金额数(￥):");    scanf("%d",&money);    Sleep(3000);

   printf("\n\t\t您的%d元已经取出,请好好保存^\_^\n",money);    break;

  case '2':

   printf("\t\t请输入您想存入的金额数(￥):");    scanf("%d",&money);    Sleep(3000);

   printf("\n\t\t您的%d元已经成功储存到您的账户上^\_^\n",money);    break;   case '3':

   {     FILE \*fp;     double avr=0;     int score,s,i=0;

     if((fp=fopen("win\_vip.txt","a+"))==NULL)     {      printf("\t\twin\_vip.txt操作失败! :(\n");      exit(1);

    }

    printf("\t\t请您为本次的服务评分(^\_^):\n"      "\t\t1分很不满意\n"      "\t\t2分不满意\n"      "\t\t3分一般满意\n"      "\t\t4分比较满意\n"

"\t\t5分很满意\n\n\t\t");

    scanf("%d",&score);

    while(fscanf(fp,"%d",&s)!=EOF)     {      i++;

avr+=1.0 \* s;

    }

     fprintf(fp,"%d\n",score);     avr+=score\*1.0;

    printf("\t\t本客户获得的平均得分为:%.1f\n"

"\t\t非常感谢您的配合^\_^\n",avr/(i+1));     break;

   }

  }

  if(ch == '3')   {    time\_t end=time(NULL);//用户离开时间

   printf("您总共用时%.0f秒\n",difftime(end,start));    current\_people--;    break;

  }

   system("pause");

 } }

/\*--------------------------------------获取当前时间--------------------------------\*/ void Get\_Time(struct tm \* &t) {

 time\_t timer;

 timer=time(NULL);  t=localtime(&timer); }

/\*------------------------------------获取正确的选择-------------------------------\*/ char Get\_Choice(char s,char e) {  char ch;

 fflush(stdin);       //清空输入缓冲区

 while(ch=getchar())  {   fflush(stdin);   //清空输入缓冲区   if(ch>=s && ch<=e)

break;

  printf("\t\t输入错误,重新输入(%c~%c):",s,e);

 }

 return ch; }

/\*------------------------------------判断是不是VIP--------------------------------\*/ int Is\_Vip(char card\_number[],char code[]) {

 FILE \*fp;               //文件指针  char c\_n[MAX];          //校验卡号  char c[MAX];            //校验密码  int i;                  //循环变量

 if((fp=fopen("vip.txt","a+"))==NULL)  {   printf("\t\tvip.txt文件不存在!\n");   exit(1);

 }

 printf("\t\t输入您的卡号(6位):");  for(i=0;i<6;i++)  {   card\_number[i]=getch();   printf("%c",card\_number[i]);  }

 card\_number[i]='\0';

 Sleep(1000);

 printf("\n\t\t密码(6位):");  for(i=0;i<6;i++)  {   code[i]=getch();   printf("\*");

 }

 code[i]='\0';  printf("\n");  Sleep(1000);

 while(fscanf(fp,"%s %s",c\_n,c)!=EOF)  {   if(!strcmp(card\_number,c\_n) && !strcmp(code,c))//如果卡号和密码都正确 那么就是VIP   {    fclose(fp);    return 1;

  }

 }

 fclose(fp);  return 0; }

/\*----------------------------------查看排队情况----------------------------------\*/ void Check\_Queue() {

 int n;//用户输入的编号  QNode \*p=q->front;  int count=0;

 if(Is\_Empty())  {   printf("\t\t没有用户排队,请先取号吧^\_^\n");   return;

 }

 printf("\t\t请输入您的排队编号:");  scanf("%d",&n);

 while(p && (p->n != n))  {   count++;   p=p->next;

 }   if(p)  {   if(count)    printf("\t\t您前面还有%d位用户,请耐心等待^\_^\n",count);   else

printf("\t\t您前面没有用户了,你可以办理业务了^\_^\n");  }  else   printf("\t\t不存在此编号,也许您输入有误、或者您还没有取号^\_^\n"); }

/\*----------------------------------评分----------------------------------\*/ void Grade(int service) {

 FILE \*fp;

 double avr=0;  //平均评分  int score;   //评分数  int i=0,s;

 printf("\t\t请您为本次的服务评分(^\_^):\n"   "\t\t1分很不满意\n"   "\t\t2分不满意\n"   "\t\t3分一般满意\n"   "\t\t4分比较满意\n"   "\t\t5分很满意\n\n\t\t");

 scanf("%d",&score);

 switch(service)  {  case 1:   {    if((fp=fopen("win1.txt","a+"))==NULL)    {     printf("\t\twin1.txt文档操作失败:(\n");     exit(1);

}

    while(fscanf(fp,"%d",&s)!=EOF)    {     avr+=s\*1.0;     i++;     continue;

}

    fprintf(fp,"%d\n",score);    avr+=score;

   printf("\t\t本窗口的平均评分为:%.1f\n"

"\t\t非常感谢您的评分(^\_^)\n\n",avr/(i+1));    break;   }  case 2:

  {    if((fp=fopen("win2.txt","a+"))==NULL)

{

    printf("\t\twin2.txt文档操作失败:(\n");     exit(1);

   }

    while(fscanf(fp,"%d",&s)!=EOF)    {     avr+=s\*1.0;     i++;     continue;

   }

    fprintf(fp,"%d\n",score);    avr+=score;

   printf("\t\t本窗口的平均评分为:%.1f\n"

"\t\t非常感谢您的评分(^\_^)\n\n",avr/(i+1));    break;

  }  case 3:

  {    if((fp=fopen("win3.txt","a+"))==NULL)    {     printf("\t\twin3.txt文档操作失败:(\n");     exit(1);

   }

    while(fscanf(fp,"%d",&s)!=EOF)    {     avr+=s\*1.0;     i++;     continue;

   }

    fprintf(fp,"%d\n",score);    avr+=score;

   printf("\t\t本窗口的平均评分为:%.1f\n"

"\t\t非常感谢您的评分(^\_^)\n\n",avr/(i+1));    break;

  }  case 4:

  {    if((fp=fopen("win4.txt","a+"))==NULL)    {     printf("\t\twin4.txt文档操作失败:(\n");

exit(1);

   }

    while(fscanf(fp,"%d",&s)!=EOF)    {     avr+=s\*1.0;     i++;     continue;

   }

    fprintf(fp,"%d\n",score);    avr+=score;

   printf("\t\t本窗口的平均评分为:%.1f\n"

"\t\t非常感谢您的评分(^\_^)\n\n",avr/(i+1));    break;

  }

 }

 fclose(fp); }

/\*--------------------------银行排队系统-------------------------\*/ /\*-------------------------( Bank\_System )-------------------------\*/

#include<stdio.h> #include"Head.h"

/\*----------------------------main函数---------------------------\*/ int main() {

 Initial();            //初始化队列  System();  return 0; }