/\*头文件声明开始\*/

#include <stdio.h> //包含标准输入输出库 ；

#include <stdlib.h> //包含标准库函数 ；

#include <time.h> //包含日期和时间的头文件，用于需要时间方面的函数 ；

#include <windows.h> //包含定义了关于创建窗口，消息循环等函数，如果要用API来做有界面的程序的话，必须要用到这个头文件；

/\*头文件声明结束\*/

/\*定义一个结构体\*/

typedef struct card

{

const char \*number; //定义字符型指针变量number ；

const char \*color; //定义字符型指针变量color ；

} Card; //定义了一个card的结构体 ；

/\*结构体定义结束\*/

/\*函数声明开始\*/

//声明将字符串放入Card结构体的函数

void FillDeck( Card \* wDeck, const char \* wNumber[], const char \* wColor[] );

//声明洗牌的函数

void Shuffle( Card \* wDeck );

//声明发牌的函数

void Deal(Card \* wDeck );

//声明写入文件的函数

void Write\_File(Card \* wDeck);

//声明读出文件的函数

void Read\_File();

//声明打印函数

void Print();

void park();

/\*函数声明结束\*/

/\*main函数开始\*/

int main()

{

Print(); // 打印首页面

Card deck[ 52 ]; // 定义Card型数组deck[]，用于存储52张牌；

const char \*number[] = {"A", "2", "3", "4", "5", "6", "7", "8", "9", "10", "J", "Q", "K" }; // 指针数组的初始化，表示13张牌

const char \*color[] = { "3","4","5","6" }; //指针数组的初始化，表示不同花色；

FillDeck( deck, number, color ); //将字符串放入Card结构体

Shuffle( deck ); //洗牌

Deal( deck ); //发牌

Write\_File(deck); //写入文件

Read\_File(); //读出文件

return 0;

}

/\*main函数结束\*/

/\*Print函数开始\*/

void Print() //打印首页面

{

printf("\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_系统正在加载中，请稍等\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\t\n\n");

for(int i=0;i<20;i++)

{

printf(".....");

Sleep(200); //延时打印 ；

}

system("CLS"); //清屏 ；

system("color 5F"); //颜色属性由两个十六进制数字指定 ， 5 = 紫色（背景色） F = 亮白色（前景色）；

printf(" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

Sleep(100);

printf(" \* A%c \* | \* A%c \* | \* A%c \* | \* A%c \*\n",3,4,5,6); //按照ASCII码表,十进制的3，用字符表示一个桃心；

Sleep(200);

printf(" \* \* | \* \* | \* \* | \* \*\n");

printf(" \* \* | \* \* | \* \* | \* \*\n");

Sleep(250);

printf(" \* \* | \* \* | \* \* | \* \*\n");

Sleep(100);

printf(" \* \* | \* \* | \* \* | \* \*\n");

printf(" \* A%c\* | \* A%c\* | \* A%c\* | \* A%c\*\n",3,4,5,6);

Sleep(100);

printf(" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

Sleep(100);

printf(" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*模拟发牌\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\n");

printf("\n");

printf("\n");

}

/\*Print函数结束\*/

**void FillDeck( Card \* wDeck, const char \* wNumber[], const char \* wColor[] )**

{

}

**void Shuffle( Card \* wDeck )**

{

}

**void Deal( Card \* wDeck )**

{

}

**void Write\_File(Card \* wDeck){**

FILE \*fp = NULL; //定义一个文件指针

fp= fopen("data.dat","wb"); //写入一个dat文件，二进制文件

if(fp == NULL) //判断是否可以打开

{

printf("error!");

return;

}

????? //你的代码：按照格式写入

fclose(fp); //关闭文件

printf("\n\n=================================存 储 成 功 !=================================\n\n\n");

}

**void Read\_File()**

{

char reply; //定义一个字符变量用来表示是否继续游戏;

int i,a; //定义局部变量i(变量循环),a(选项);

do{

Card wDeck[52];

Card \*pointer; //定义一个 Card型的指针pointer；

pointer=wDeck; //让 pointer指向 wDeck；

FILE \*fp = NULL; //定义文件指针；

fp = fopen("data.dat","rb"); //打开文件，默认地址是执行文件下 ；

if(fp == NULL)

{

printf("error!");

return ;

}

??? //你的代码：读出文件内容流；

fclose(fp); //关闭指针；

printf("\n");

printf("请问您想看第几副牌（1--4）：");

scanf("%d",&a);

//以下写入你的代码：根据a的值输出牌面信息

？？？？

}while(reply=='Y'||reply=='y'); //当输入y继续循环

}

void park()

{

int i,a=0;

printf(" 华丽丽的洗牌中! \n");

while(a<15) //循环15次

{

//每个for循环代表每个颜色持续的时间

for(i=0;i<9;i++);

system("color a"); //A = 淡绿色

for(i=0;i<9;i++);

system("color b"); //B = 淡浅绿色

for(i=0;i<9;i++);

system("color c"); //C = 淡红色

for(i=0;i<9;i++);

system("color d"); //D = 淡紫色

for(i=0;i<9;i++);

system("color e"); //E = 淡黄色

for(i=0;i<9;i++);

system("color f"); //F = 亮白

a++;

}

printf("\n\n\n 洗牌结束! \n\n");

}