



**学年论文（课程设计）**

题目**： 三门问题**

学 院 数学与信息科学学院

学科门类 理学

专 业 数学与应用数学

学 号 20201501083

姓 名 李瑞

指导教师 高相辉

2022年 6月23日

河北大学学年论文（课程设计）任务书

（指导教师用表）

装

订

线

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 学 生 姓 名 | | 李瑞 | 指 导 教 师 | | 高相辉 |
| 论文（设计）题目 | | 三门问题 | | | |
| 主要研究  （设计）内容 | 使用C++编程解决三门问题，利用程序模拟随机实验，通过统计的方法揭示三门问题后面的概率问题。 | | | | |
| 研究方法 | 结合所学课程，研究随机实验算法，给出算法思想，并在此基础上实现设计算法。按照学校课程设计报告模板格式要求，编写文档。编程实验，程序代码编写、调试、测试，完成程序各功能的实现。 | | | | |
| 主要任务  及目标 | 研究三门问题所隐含的概率统计问题,给出C++语言模拟随机实验的设计和相应算法。通过实践，使学生进一步掌握程序设计的方法，及运用程序设计语言解决实际问题的能力。 | | | | |
| 主要参  考文献 | [1] Visual C++ 编程、开发相关参考书  [2] C程序设计(教科书) | | | | |
| 进度安排 | 论文（设计）各阶段名称 | | | 日期 | |
| 数据结构、功能结构设计 | | | 第十五周 | |
| 程序设计规划、主要部分实验 | | | 第十五周 | |
| 编程、撰写论文 | | | 第十六周 | |
| 撰写论文 | | | 第十七周 | |

指导教师签字:

系主任签字：

主管教学院长签字：

河北大学学年论文（课程设计）成绩评定表

学院： 数学与信息科学学院

装

订

线

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 学生姓名 | | 李瑞 | | 专业 年级 | 数学与应用数学 20级 |
| 论文（设计）题目 | | | 三门问题 | | |
| 论文（设计）内容提要 | 使用C/C++来模拟三门问题，通过10000次的模拟给出了：在主持人知道门后情况和主持人不知道门后两种情况下，参赛者在选择一扇门后更改决策与不更改决策能获得汽车的概率。可分为三种情况。  第一种情况：主持人知情并且主持人会打开非汽车的门，讨论参赛者换门与不换门能获得汽车的概率，通过程序模拟得到，参赛者不换门获胜的概率是1/3，而换门获胜的概率是2/3。  第二种情况：主持人不知门后情况，会随机打开两扇门中的一扇，通过程序模拟得到，参赛者换门与不换门能得到汽车的概率都是1/2  第三种情况：参赛者不知道主持人是否知道门后情况，并且主持人诚实的回答了参赛者的问题。  并且在此基础上扩展延申到四门问题，百门问题。 | | | | |
| 指导教师评语 |  | | | | |
| 成绩： 指导教师（签名）： 年 月 日 | | | | |

三门问题

# 摘　　要

　　三门问题，是一个博弈论的数学游戏问题。参赛者前面有三扇门，门后分别藏有两只羊和一辆车。此问题有三种不同的情况。情况一是主持人知道每扇门后分别是什么，首先参赛者选择一扇门，主持人会打开剩下两扇中有羊的一扇门，然后询问参赛者是否改变一开始的选择——换成剩下的一扇门。参赛者是否改变选择，会影响他中奖的概率。情况二是主持人不知道每扇门后是什么，主持人打开剩下两扇门中的任意一扇门，其他步骤同情况一。情况三是主持人知道每扇门后的情况，但参赛者不知道主持人是否知道每扇门后的情况，他询问主持人，主持人诚实的回答他：“知道。”其他步骤同情况一。这三种情况基于主持人是否知道门后情况，要求用代码及分析来说明参赛者改变或不改变两种情况下中奖的概率。三门问题也可以进行扩展，变成四门问题。四门问题是指，在参赛者进行最初选择后，问主持人打开一扇门和两扇门两种情况下，参赛者是否改变的中奖率。

关键词：三门问题；选择；中奖率

**The three-door problem**

**ABSTRACT**

The three-door problem is a mathematical game problem of game theory.There were three doors in front of the contestants, with two sheep and a car behind them.There are three different situations in this problem.First, the host knows what is behind each door. First, the contestant chooses a door, and the host will open the remaining two doors with sheep, and then ask the contestant whether to change the initial choice of —— to the remaining door.Whether a contestant changes his choice will affect his probability of winning the lottery.Case two is that the host does not know what is behind each door, the host opens any one of the remaining two doors, and the other steps are the same as situation one.Situation third is that the host knows the situation behind each door, but the participants do not know whether the host knows the situation behind each door, he asked the host, the host honestly answered him honestly: "Ｉknow."The other steps are the same case.These three cases are based on whether the host knows the situation behind the door, and the code and analysis are required to explain the probability of winning in the two cases.The three-door problem can also be expanded into four-door problems.Four-door questions refer to the winning rate changed by the contestant after the contestant makes s initial selection.

**Key words**:The three-door problem；choice；winning rate

目录

[三门问题 4](#_Toc107095714)

[摘　　要 4](#_Toc107095715)

[1 功能需求 7](#_Toc107095716)

[1.1功能任务 7](#_Toc107095717)

[1.2使用的开发工具 7](#_Toc107095718)

[1.3设计界面 7](#_Toc107095719)

[1.4预期指标说明 7](#_Toc107095720)

[2 详细设计 8](#_Toc107095721)

[2.1功能结构 8](#_Toc107095722)

[2.2实现方法 8](#_Toc107095723)

[2.3思想 9](#_Toc107095724)

[2.3.1三门问题 9](#_Toc107095725)

[2.3.2四门问题 9](#_Toc107095726)

[2.3.3百门问题 9](#_Toc107095727)

[2.4算法描述 9](#_Toc107095728)

[2.5数据组织结构 9](#_Toc107095729)

[3 程序、算法流程图 10](#_Toc107095730)

[3.1三门问题 10](#_Toc107095731)

[3.1.1主持人知道汽车的位置 10](#_Toc107095732)

[3.1.2主持人不知道汽车的位置 10](#_Toc107095733)

[4 程序实现 11](#_Toc107095734)

[4.1三门问题 11](#_Toc107095735)

[4.1.1主持人知道门后情况的获奖概率 11](#_Toc107095736)

[4.1.2主持人不知道门后情况的获奖概率 12](#_Toc107095737)

[4.2四门问题 14](#_Toc107095738)

[4.2.1主持人知道门后情况的获奖概率 14](#_Toc107095739)

[4.3百门问题 17](#_Toc107095740)

[5 程序测试 19](#_Toc107095741)

[5.1三门问题 19](#_Toc107095742)

[5.1.1主持人知道门后情况 19](#_Toc107095743)

[5.1.2主持人不知道门后情况 19](#_Toc107095744)

[5.2四门问题 19](#_Toc107095745)

[5.3百门问题 20](#_Toc107095746)

[6 　扩展思考题 22](#_Toc107095747)

[７ 总结个人心得体会及建议 23](#_Toc107095748)

[谢 辞 25](#_Toc107095749)

1 功能需求

1.1功能任务

基于三门问题的主持人是否知道门后情况，用C语言进行编程给出对应情况的参赛者的中奖率，并进行分析总结，给出具体的程序流程、功能结构以及主要代码说明。

1.2使用的开发工具

Visual Studio 2013

1.3设计界面

代码的输出界面是直接输出相应情况下对应的获奖概率。

1.4预期指标说明

由程序实现4.1.1得到在主持人知道门后情况并且打开非汽车的门，参赛者换门与不换门能获得汽车的概率。

由程序实现4.1.2得到在主持人不知道门后情况，随机打开剩下两扇门中的一扇，参赛者换门与不换门能获得汽车的概率。

由程序实现4.2延申讨论四门问题。

由程序实现4.3讨论百门问题。

2 详细设计

2.1功能结构

1. 三门问题主持人知道情况：
2. 四门问题中主持人知情并打开剩下两扇是羊的门：
3. 百门问题：

这三种情况代码几乎相同，功能结构分析如下：

首先自定义两个函数void function(int num,bool chose)，bool playGame(bool chose)。

其中playGame(bool chose)用于模拟游戏,用rand()随机数,模拟汽车所在的门和参赛者猜测的门,经过逻辑语句会返回true或者false。返回true表明参赛者获得汽车,返回false表明参赛者没有获得汽车。

在function(int num,bool chose)里通过for循环执行调用10000次playGame()函数,统计playGame()返回true的次数，从而实现统计概率。

4）三门问题主持人不知道情况：

首先自定义两个函数void hostNotKnowChange(int doors[])，void hostNotKnowNotChange(int doors[])。

其中void hostNotKnowChange(int doors[])用于主持人不知道门后情况时，计算参赛者改变决策的中奖概率，用srand和rand组合使得主持人和参赛者的选择随机化，运用逻辑语句使得选择合理化且符合题目，并且统计获奖次数和概率。

void hostNotKnowNotChange(int doors[])功能同理，用于主持人不知道门后情况时，计算参赛者不改变决策的中奖概率。

5）四门问题主持人知道情况，打开剩下的一扇是羊的门：

首先让用户判断第二次选择是否换门，用for循环，模拟TOTAL\_COUNT次实验，参赛者选择的门是随机的，主持人选择的门须符合实际条件。用if-else语句分成换门与不换门两种情况，并讨论符合实际的情况下，换门与不换门的获奖次数，最终统计获奖概率。

2.2实现方法

使用程序模拟实验，通过返回确定的true/false实现统计参赛者获得汽车的次数。使用自定义函数实现模拟，统计计数的功能，通过rand()随机数实现猜测门号。使用for循环while循环和if-else控制逻辑语句，实现游戏模拟。通过%f格式化输出获得汽车的概率。

2.3思想

2.3.1三门问题

1）主持人知道门后情况，主持人帮忙排除一个选项，参赛者换门的获奖概率是2/3，不换门的获奖概率是1/3。

2）而当主持人不知道门后情况时，随机开启的一扇门可能是汽车，可能是山羊，无论开启的门是哪扇，参赛者换与不换都是1/2。

３）当参赛者不知道主持人知道的信息时，但主持人诚实回答参赛者，参赛者换门的获奖概率会提高，最多提高1/6。

2.3.2四门问题

1）主持人知道门后情况。主持人帮忙排除一个选项，即打开剩下三门中一扇是羊的门，根据程序的运行结果，可知参赛者换门的获奖概率是3/8，不换门的获奖概率是1/4。主持人帮忙排除两个选项，即打开剩下三门中两扇是羊的门，根据程序的运行结果，可知参赛者换门的获奖概率是约1/4，不换门的获奖概率是3/4。

2）主持人不知道门后情况。参赛者与主持人的选择都是随机的，待选项被选的可能性相等，所以和三门问题一样，参赛者改变或不改变决策，获奖概率都是1/2。

2.3.3百门问题

1. 主持人知道门后情况。假设有100扇门,参赛者选了一扇能获奖的概率是1/100，当主持人开启了98扇门且门后都是羊,参赛者应该更换选择,因为主持人已经排除了剩下99个选择中98个选择，不换门获得汽车的概率是1/100,而换门获得汽车得概率是99/100。
2. 主持人不知道门后情况。参赛者换门与不换门的概率都是1/2。

2.4算法描述

首先规定测试次数，再来分为两种情况：参赛者在第二次选择的过程中是否改变决策，并通过相应的代码使得每次测试的每种情况合理化，最终通过获奖次数／总测试次数来输出两种情况下的获奖概率。

2.5数据组织结构

代码中涉及到的类型定义有：整型数据，整型数组，字符数组，浮点型数据。

一维数组表示游戏门的个数

3 程序、算法流程图

3.1三门问题

3.1.1主持人知道汽车的位置

换门

获奖

选择羊

不换门

不获奖

换门

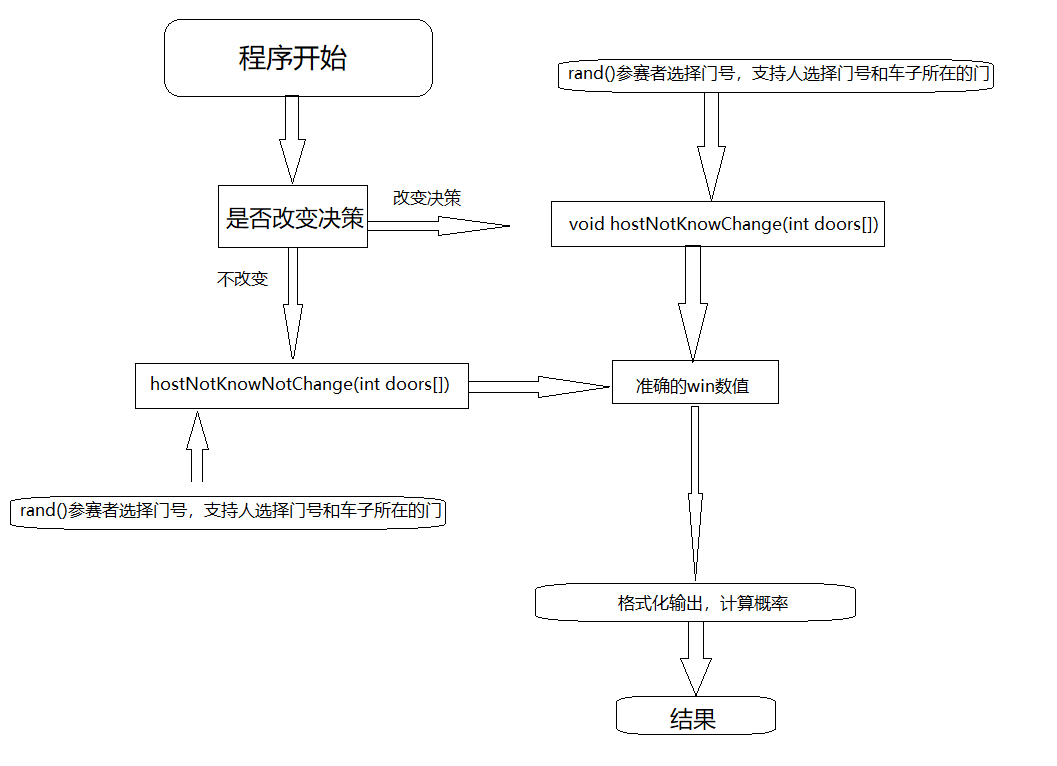
不获奖

选择汽车

不换门

获奖

3.1.2主持人不知道汽车的位置



4 程序实现

4.1三门问题

4.1.1主持人知道门后情况的获奖概率

#include<iostream> //使用C++语言及C语言进行编程

#include<cstdio>

#include<cmath>

#include<cstdlib>

#include<algorithm>

#include<stdbool.h> //C语言中的bool类型头文件

#define N 100000 //N为总的测试次数

using namespace std; //std命名空间内所有的标识符都已经声明并且存在

void function(int num, bool chose);

bool playGame(bool chose);

int main()

{

printf("选择不换门时的获奖概率为：\n");

function(N, false); //选择不换门

printf("选择换门时的获奖概率为：\n");

function(N, true); //选择换门

return 0;

}

//函数功能：统计总共在num次测试中，参赛者的获胜次数及获胜概率

void function(int num, bool chose)

//num为总测试次数，chose为参赛者的选择(两种选择:true代表参赛者选中汽车，false 代表参赛者没选中)

{

int wins = 0; //赢的次数

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

{

if (playGame(chose))

{

wins++; //参赛者最终选中了汽车所在的门

}

}

printf("\t获胜次数为：%d\n", wins);

printf("\t获胜概率为：%f\n", 1.0 \* wins / num);

}

//函数功能：模拟游戏过程

bool playGame(bool chose)

{

int car = rand() % 3 + 1; //汽车所在的那扇门

int guess = rand() % 3 + 1; //参赛者选择的那扇门

if (car == guess) //参赛者选中了汽车所在的门

{

if (chose == true) //参赛者决定换门 -> 怎么都不会得到汽车

{

return false;

}

else //参赛者不换门 -> 能得到汽车

{

return true;

}

}

else //参赛者没有选中汽车所在的门

{

if (chose == true)

{ //参赛者决定换门 -> 主持人(位于上帝视角)开启不是汽车的门 -> 参赛者能得到汽车

return true;

}

else //参赛者不换门 -> (选的是错的还不换门)，肯定得不到汽车

{

return false;

}

}

}

4.1.2主持人不知道门后情况的获奖概率

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#define N 100000 //总测验次数

void hostNotKnowChange(int doors[]);

void hostNotKnowNotChange(int doors[]);

int main()

{

int doors[3]; //doors[1]->门后是汽车，doors[0]->表示门后是羊

hostNotKnowChange(doors);

hostNotKnowNotChange(doors);

while (1);

return 0;

}

//函数功能：在主持人不知道门后是什么的情况下，计算参赛者改变决策的中奖次数和中奖概率

void hostNotKnowChange(int doors[])

{

int win = 0; //所有测试中的获奖次数

int temp, i = 0;

srand((unsigned)time(NULL)); //使得下面的rand()函数每次产生的随机数都不同

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

{

int choice = rand() % 3; //随机选择三扇门中的其中一扇

int b = 0, j = 0;

b = rand() % 1 + 2;

doors[0] = 0; doors[1] = 0; doors[2] = 0;

doors[rand() % 3] = 1;

j = (choice + b) % 3; //主持人选择的那扇门在剩下的两扇门中随机抽取

//下面这个if-else语句是为了确保每一次实验的合理性，即一扇门只能被一个人且只能选择一次

if (doors[j])

--i; //j=1,主持人选中汽车

else

{

if (j + choice == 1)

temp = 2;

else if (j + choice == 2)

temp = 1;

else

temp = 0;

if (doors[temp])

++win; //temp=1,参赛者选中汽车

}

}

printf("主持人不知道门后情况，改变决策后的中奖次数为：%d\n", win);

printf("主持人不知道门后情况，改变决策后的中奖概率为：%f\n", win / (float)N);

}

//函数功能：在主持人不知道门后是什么的情况下，计算参赛者不改变决策的中奖次数和中奖概率

void hostNotKnowNotChange(int doors[])

{

int win = 0; //中奖次数

int temp, i = 0;

srand((unsigned)time(NULL));

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

{

int choice = rand() % 3; //参赛者随机选择其中的一扇门

int b = 0, j = 0;

b = rand() % 1 + 2;

doors[0] = 0; doors[1] = 0; doors[2] = 0;

doors[rand() % 3] = 1;

j = (choice + b) % 3; //主持人选择的门

temp = choice; //参赛者选择的门

if (doors[j]) //j=1,主持人选中了汽车

--i;

if (doors[temp]) //temp=1,参赛者选中了汽车

++win; //获奖次数加1

}

printf("\n主持人不知道门后情况，不改变决策的中奖次数为：%d\n", win);

printf("主持人不知道门后情况，不改变决策的中奖概率为：%f\n", win / (float)N);

}

4.2四门问题

4.2.1主持人知道门后情况的获奖概率

1. 主持人打开剩下三门中的一门，后面是山羊

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#define SHEEP 3

#define CAR 1

#define TOTAL\_COUNT 100000 //总测试次数

#define DOOR\_COUNT 4 //门数

#define RAND\_DOOR\_INDEX rand()%DOOR\_COUNT

int main()

{

int wins = 0;

srand((unsigned int)time(0));

printf("你想要换门吗？ 输入'y'换门，否则不换\n");

char c;

c = getchar();

printf("门数：%d, 测试次数:%d, 策略： %s \n", DOOR\_COUNT, TOTAL\_COUNT, c == 'y' ? "change" : "not change");

for (int i = 0; i < TOTAL\_COUNT; ++i)

{

int car\_door\_index = RAND\_DOOR\_INDEX;//随机模拟背后有车的门

int guest\_first\_selected\_index = RAND\_DOOR\_INDEX;//第一次参赛者选择的门

int host\_selected\_index = RAND\_DOOR\_INDEX;

while (host\_selected\_index == guest\_first\_selected\_index || host\_selected\_index == car\_door\_index)

//主持人选择的门，和参赛者选择的门不能相等，不能选择有车的门，不符合条件就重新选

host\_selected\_index = RAND\_DOOR\_INDEX;

if (c == 'y') //换门

{

int guest\_second\_selected\_index = RAND\_DOOR\_INDEX;

while (guest\_second\_selected\_index == host\_selected\_index || guest\_second\_selected\_index == guest\_first\_selected\_index)

//参赛者第二次选择的门，和第一次选择的门不能相等，不能和主持人的选择相等，不符合条件就重新选

guest\_second\_selected\_index = RAND\_DOOR\_INDEX;

if (guest\_second\_selected\_index == car\_door\_index)

wins += 1; //选中汽车，获奖次数加一

}

else //不换门

{

if (guest\_first\_selected\_index == car\_door\_index)

wins += 1; //选中汽车，获奖次数加一

}

}

printf("获奖次数为:%d，获奖概率为：%f\n", wins, wins / (float)TOTAL\_COUNT);//输出获奖次数和概率

return 0;

}

2）主持人打开剩下三门中的两门，后面都是山羊

#include<iostream> //使用C++语言及C语言进行编程

#include<cstdio>

#include<cmath>

#include<cstdlib>

#include<algorithm>

#include<stdbool.h> //C语言中的bool类型头文件

#define N 100000 //N为总的测试次数

using namespace std; //std命名空间内所有的标识符都已经声明并且存在

void function(int num, bool chose);

bool playGame(bool chose);

int main()

{

printf("选择不换门时的获奖概率为：\n");

function(N, false); //选择不换门

printf("选择换门时的获奖概率为：\n");

function(N, true); //选择换门

return 0;

}

//函数功能：统计总共在num次测试中，参赛者的获胜次数及获胜概率

void function(int num, bool chose)

//num为总测试次数，chose为参赛者的选择(两种选择:true代表参赛者选中汽车，false 代表参赛者没选中)

{

int wins = 0; //赢的次数

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

{

if (playGame(chose))

{

wins++; //参赛者最终选中了汽车所在的门

}

}

printf("\t获胜次数为：%d\n", wins);

printf("\t获胜概率为：%f\n", 1.0 \* wins / num);

}

//函数功能：模拟游戏过程

bool playGame(bool chose)

{

int car = rand() % 4 + 1; //汽车所在的那扇门

int guess = rand() % 4 + 1; //参赛者选择的那扇门

if (car == guess) //参赛者选中了汽车所在的门

{

if (chose == true) //参赛者决定换门 -> 怎么都不会得到汽车

{

return false;

}

else //参赛者不换门 -> 能得到汽车

{

return true;

}

}

else //参赛者没有选中汽车所在的门

{

if (chose == true)

{ //参赛者决定换门 -> 主持人(位于上帝视角)开启不是汽车的门 -> 参赛者能得到汽车

return true;

}

else //参赛者不换门 -> (选的是错的还不换门)，肯定得不到汽车

{

return false;

}

}

}

4.3百门问题

#include<iostream> //使用C++语言及C语言进行编程

#include<cstdio>

#include<cmath>

#include<cstdlib>

#include<algorithm>

#include<stdbool.h> //C语言中的bool类型头文件

#define N 100000 //N为总的测试次数

using namespace std; //std命名空间内所有的标识符都已经声明并且存在

void function(int num, bool chose);

bool playGame(bool chose);

int main()

{

printf("选择不换门时的获奖概率为：\n");

function(N, false); //选择不换门

printf("选择换门时的获奖概率为：\n");

function(N, true); //选择换门

return 0;

}

//函数功能：统计总共在num次测试中，参赛者的获胜次数及获胜概率

void function(int num, bool chose)

//num为总测试次数，chose为参赛者的选择(两种选择:true代表参赛者选中汽车，false 代表参赛者没选中)

{

int wins = 0; //赢的次数

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

{

if (playGame(chose))

{

wins++; //参赛者最终选中了汽车所在的门

}

}

printf("\t获胜次数为：%d\n", wins);

printf("\t获胜概率为：%f\n", 1.0 \* wins / num);

}

//函数功能：模拟游戏过程

bool playGame(bool chose)

{

int car = rand() % 100 + 1; //汽车所在的那扇门

int guess = rand() % 100 + 1; //参赛者选择的那扇门

if (car == guess) //参赛者选中了汽车所在的门

{

if (chose == true) //参赛者决定换门 -> 怎么都不会得到汽车

{

return false;

}

else //参赛者不换门 -> 能得到汽车

{

return true;

}

}

else //参赛者没有选中汽车所在的门

{

if (chose == true)

{ //参赛者决定换门 -> 主持人(位于上帝视角)开启不是汽车的门 -> 参赛者能得到汽车

return true;

}

else //参赛者不换门 -> (选的是错的还不换门)，肯定得不到汽车

{

return false;

}

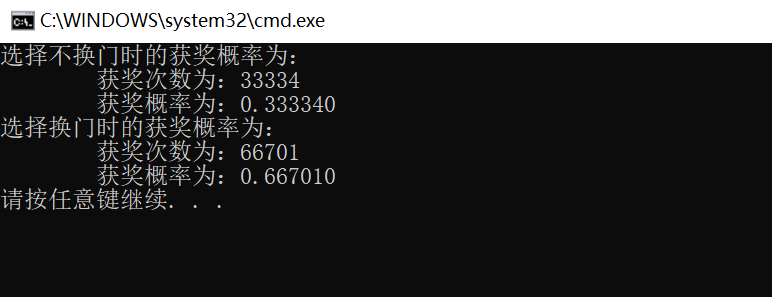
}

}

5 程序测试

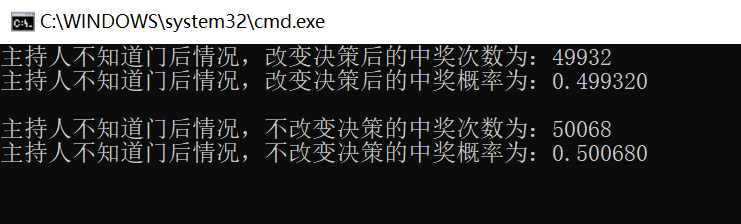
5.1三门问题

5.1.1主持人知道门后情况



由代码的运行结果可知，当主持人知道门后的具体情况时，打开剩下的两扇门中是羊的那扇，参赛者换门的获奖概率为0.333340（约1/3），不换门的获奖概率为0.667010（约2/3）。

5.1.2主持人不知道门后情况

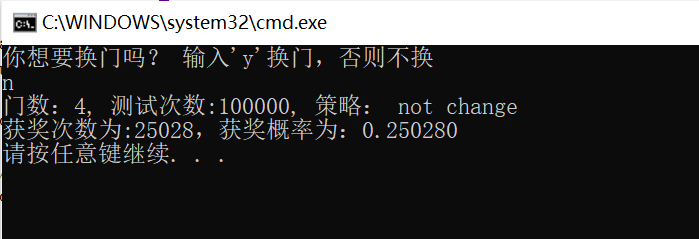
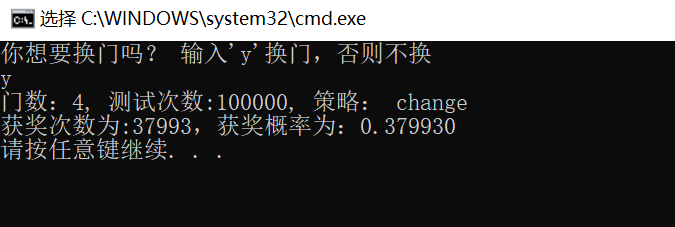


由代码的运行结果可知，当主持人知道门后的具体情况时，打开剩下的两扇门中是羊的那扇，参赛者换门的获奖概率为0.499320（约1/2），不换门的获奖概率为0.500680（约1/2）。

5.2四门问题

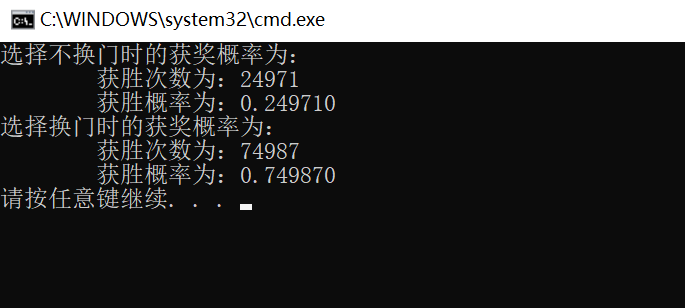
主持人知道门后情况

1. 打开剩下门中的一扇



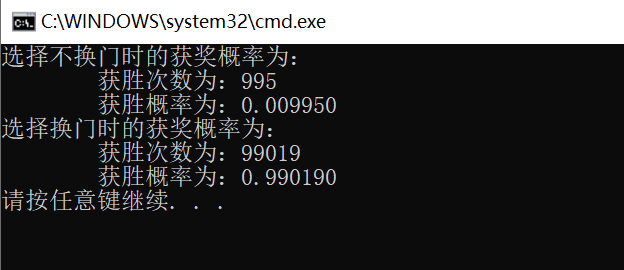
由代码的运行结果可知，当主持人知道门后的具体情况，然后打开剩下三扇门中是羊的一扇，参赛者换门能够获奖的概率为0.379930（约1/8），不换门能够获奖的概率为0.250280（约1/4）。

1. 打开剩下门中的两扇



由代码的运行结果可知，当主持人知道门后的具体情况时，然后打开剩下的三扇门中是羊的两扇，则参赛者换门的获奖概率为0.749870（约3/4），不换门的获奖概率为0.249710（约1/4）。

5.3百门问题



由代码的运行结果可知，当主持人知道门后的具体情况时，然后打开剩下99扇门中是羊的98扇，则参赛者换门的获奖概率为0.990190（约99/100），不换门的获奖概率为0.009950（约1/100）。

6 　扩展思考题

1. 主持人提供的答案含有多大的信息量？

三门问题中，假设没有主持人帮助打开剩余两扇其中的一扇，此时依然有换门与不换门，但是无论怎么换，参赛者能得到汽车的概率都是1/3。而主持人开门这一事件会排除掉一个非汽车的门，相当于泄露信息给参赛者，换门的选项从两个减到了一个，正确率自然就会增加。

1. 是否可以计算出主持人提供的答案价值几何？

三门问题中主持人知道门后情况对比不知道门后情况，参赛者换门获奖的概率最多增加了１/６。但是因为在参赛者比赛时，不知道自己所选的门和剩余两扇门之后是什么，所以对于剩余的两扇门，当主持人知情时，肯定会开启不是汽车的门。而这时参赛者选择的汽车所在的门，在听到主持人回答知道后，参赛着可能会动摇，改变决策或者不改变决策。

但是通过实验模拟可知，在主持人知情的情况下，换门能获得汽车的概率更大。由此参赛则和能否获得汽车完全取决于参赛者对主持人回答的信与不信，所以我认为不能计算出主持人提供的答案价值几何。

3. 如果不能计算出，你需要什么信息才能评估出这条信息的价值？

询问主持人：更改决策能得到汽车的概率大还是不更改决策能得到汽车的概率大

７ 总结个人心得体会及建议

　　在这次学习的过程中，我感觉C语言虽然难，但是在敲代码的过程中，又感觉它很有意思，并且在理解问题并解答出来的时候，我非常感兴趣并且有成就感。通过分析解答三门问题及拓展，我认识到要多看多参考前人的优秀文献，学习其中编写程序的思路。同学之间的讨论也是很有必要的，对于吸收知识有莫大的帮助。我们还要注意学科交叉的内容，如计算机与概率论、数学的知识等。并且我们要以问题为导向，在解决问题的过程中学习，也要不断试错，不要怕犯错，在错误中学习。我们还要找到适合自己的学习方法，并且保持好奇心，再好好努力学精学透，我们就一定会取得进步。

参考文献

1. CSDN，http://t.csdn.cn/79J4n，2021-12-31
2. CSDN，http://t.csdn.cn/GZYj3，2022-04-02
3. 茆诗松，程依明，濮晓龙．概率论与数理统计教程第二版［M］.北京:高等教育出版社，2011.45-49
4. 苏小红,王宇颖,孙志岗等.C 语言程序设计（第 3 版）.北京：高等教育出版社，2019：189~214
5. 何书元．三门问题［Ｊ］．数学通报，２０１５，５４（３）：１－２
6. 淑生．汽车、山羊及其他———一道概率趣题及由此引起 的 思考［Ｊ］．自然杂志，１９９２（１２）：９３５－９３９

谢 辞

贯穿整个17周的C语言学习结束了，虽然整个学期几乎都是上网课，在后期的课程中甚至回到家进行复习和论文的撰写。但是，这一段时间的学习也让我成长了很多。首先感谢我的老师，感谢老师在疫情急迫的时刻能够继续在网上耐心地授课，稳住我们想懈怠的心。我们高年级的同学，一些课程不可避免的会和C语言这门课冲突，因此还要感谢21级的学弟学妹们，及时地告知我们这些高年级的同学一些相关的作业、课堂内容等。我在平时的C语言学习以及在本次论文的撰写中，会遇到大大小小的问题，感谢我的同学不断地解答我的疑问，从而使我们通过交流能够更好的理解这门课。我也要谢谢我自己，在这个学期的过程中始终坚持学习C语言，在放假回家期间没有马马虎虎的对待C语言最后的这篇课程设计的论文。