****

**Tongji University**

**离散数学实验报告**

**002-**命题逻辑推理

专 业： 软 件 工 程

指导教师： 唐 剑 锋

学 号： 2 1 5 3 0 6 1

姓 名： 谢 嘉 麒

1. **命题逻辑推理**
   1. **题目内容**

本次题目以具体实例出发，通过程序模拟命题逻辑推理，对实际问题进行解答，并通过代码逻辑和计算体现命题逻辑推理的过程。

* 1. **实验用例**

本次实验共给出五个命题，通过命题逻辑推理的方法判断作案者，具体过程由C++程序实现。命题汇总如下：

（1）营业员 A 或 B 偷了手表；

（2）若 A 作案，则作案不在营业时间；

（3）若 B 提供的证据正确，则货柜未上锁；

（4）若 B 提供的证据不正确，则作案发生在营业时间；

（5）货柜上了锁。

* 1. **实验目的**

加深对命题逻辑推理的理解，通过命题逻辑推理方法解决实际问题，并实现对实际问题的程序化操作。

1. **整体设计思路**
   1. **实验设计原理**
2. 首先将所有命题符号化，方便其为计算机所识别和处理，并作为命题变元参与主程序运算。
3. 将以上命题组成复合命题，并用C++语言中的运算符及函数替换逻辑联结词，用变量替换相应的命题变元，组成程序语言中的表达式
4. 通过不断改变命题变元的初始值进行尝试，最终得到逻辑运算的正确结果。
   1. **程序设计思想**

本次程序设计的特点主要体现在命题变元变量化、运用程序语言实现逻辑联结词和运用嵌套循环进行逻辑推理三个思想。

* + 1. 命题变元变量化

根据C++语言程序设计需要，可以将命题变元设计为整数类型（int）或者布尔型数（bool），本次实验将命题变元用int型数进行代换，并采用A-E进行辨别和操作。

1. int A; //"营业员 A 偷了手表"
2. int B; //"营业员 B 偷了手表"
3. int C; //"作案不在营业时间"
4. int D; //"提供的证据正确"
5. int E; //"货柜未上锁"
   * 1. 运用程序语言实现逻辑联结词

即将抽象逻辑联结词用C++语言的方式在程序中实现，可实现方式为基本运算符和函数，本次实验利用“||”“！”“&&”等运算符及其组合对逻辑关系进行程序化。

* + 1. 运用嵌套循环进行逻辑推理

本次实验需要通过对命题变元赋值TRUE(1)或者FALSE(2)进行不断尝试，这一需求则可通过嵌套for循环进行实现。每个for循环对应一个命题变元，设定初始值为0（命题变元为假），当未得出正确结果时则回溯使命题变元加一（此时命题变元为真），进行尝试，即实现命题逻辑推理的模拟。

1. **数据结构运用**

本次解题需要将命题变元符号化，可采用的数据结构较多，主要包含整数类型和布尔类型。实验中笔者使用整数类型（int型）进行操作。

1. **核心算法**

本次实验设计核心算法主要是将逻辑联结词程序化，并通过嵌套for循环进行命题逻辑推理的模拟。

1. 将命题逻辑代码化后作为嵌套循环的终止条件（即若命题变元的真假符合此式，就代表命题逻辑推理成功），得到的表达式如下：

(A || B) && (! A || C) && (! D || E) && (D ||! C) &&! E

1. 嵌套for循环如下：

for (A = 0; A <= 1; A++)

for (B = 0; B <= 1; B++)

for (C = 0; C <= 1; C++)

for (D = 0; D <= 1; D++)

for (E = 0; E <= 1; E++)

1. **调试过程碰到的问题**

本次实验主要问题在于通过抽象逻辑得到嵌套循环的终止条件，以及利用合理方式对命题变元的真假赋值进行不断尝试。

通过笔者的尝试和分析，发现终止条件实际上可以由已给出的命题变元进行代码化得到；而如何不断进行真假赋值则是笔者尝试过for循环、嵌套for循环、switch-case语句后，得到嵌套for循环应为较优解的结果。

1. **心得体会**

通过本次作业，笔者加深了对命题逻辑推理的理解，也学会了将抽象数学中的逻辑进行程序化，代码化。

1. **附件：源程序**
2. int main()
3. {
4. int A; //"营业员 A 偷了手表"
5. int B; //"营业员 B 偷了手表"
6. int C; //"作案不在营业时间"
7. int D; //"提供的证据正确"
8. int E; //"货柜未上锁"
9. for (A = 0; A <= 1; A++)
10. for (B = 0; B <= 1; B++)
11. for (C = 0; C <= 1; C++)
12. for (D = 0; D <= 1; D++)
13. for (E = 0; E <= 1; E++)
14. if ((A || B) && (!A || C) && (!D || E) && (D || !C) && !E)
15. cout << "A = " << A << ", B = " << B << endl;
16. return 0;
17. }
18. **实验结果**

由程序结果得到是B偷了手表

