****

**《程序设计课程实践》**

**综合项目实验报告**

# 项目名称：校园外来人员进出监控与管理系统

**团队成员：（学号 姓名）（组长放第1个位置）**

**19063140 郑凯心**

**19063110 蔡思林**

**19071322 柯瑞阳**

完成时间 2021 年 6 月

# 校园外来人员进出监控与管理系统项目实验报告

## 团队成员组成及分工

|  |  |  |
| --- | --- | --- |
| **学号** | **姓名** | **详细任务分工** |
| **19063140** | **郑凯心** | 代码 |
| **19063110** | **蔡思林** | 代码 |
| **19071322** | **柯瑞阳** | 报告 |

## 2．开发背景

校园外来人员进出监控与管理系统基于调研新冠疫情期间校园人员进出管理流程，设计得到合理可行的软件系统实现人员进出登记与管理功能，能实现记录人员进出申请和进出登记等信息，能按人员信息、日期、时间段等信息查询人员进出校园记录。人员登记基本信息包括：姓名、性别、联系电话、身份证号、单位信息、车牌号、进校事由、担保人、担保人电话、健康码、14天内是否去过疫区、是否有咳嗽发热等症状、申请进入时间、申请离开时间、实际进入时间、实际离开时间等信息。

## 3.系统功能设计

### 3.1系统功能模块设计

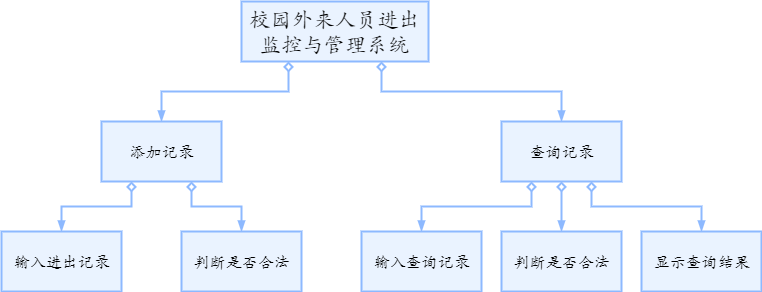
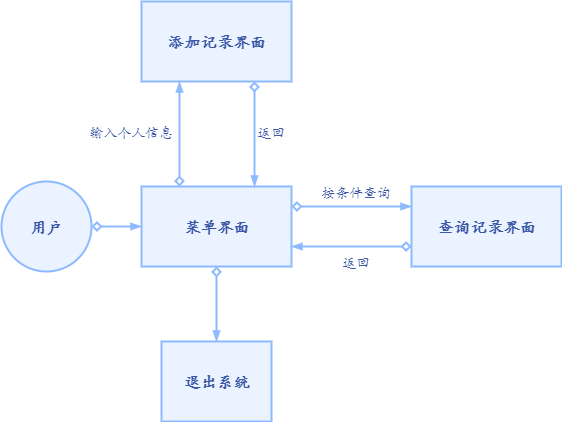


图1 系统功能模块图

### 3.2 校园外来人员进出监控与管理系统执行流程图



## 4.项目创建

### 4.1系统开发环境要求

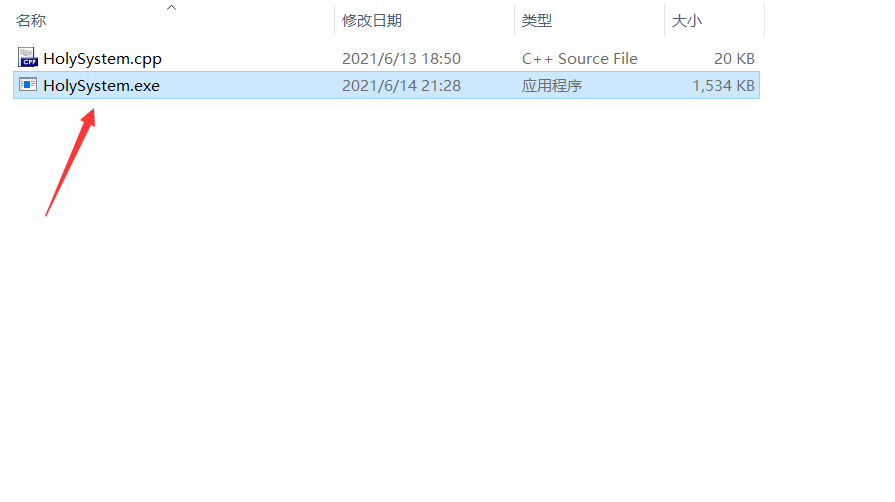
本项目的开发及运行环境要求：Windows 10 + Visual Studio Code + C++

操作系统：Windows 10

开发工具：Visual Studio Code

开发语言：C++

### 4.2 项目创建过程



双击HolySystem.exe运行程序

## 5.预处理模块设计

文档完成人：柯瑞阳

### 5.1 文件引用

#include <bits/stdc++.h> //万能头文件，用于调用各类库函数

#include <tchar.h> //用于在控制台输出宽字符

### 5.2定义全局变量

string buf; //全局字符缓冲区

### 5.3函数声明

1. string trim(string str)

{

str.erase(0, str.find\_first\_not\_of(\_T(" \n\r\t")));

str.erase(str.find\_last\_not\_of(\_T(" \n\r\t")) + 1);

return str;

}

输入参数：string str 字符串

输出参数：string

实现功能：去除首尾空字符

1. template <class T>

void input(istream &is, T &obj)

{

for (bool ok = false; !ok; )

{

getline(is, buf);

try

{

ok = obj.check();

}

catch (const char\* msg)

{

cerr << "错误:" << msg << "请重新输入:" << endl;

}

}

}

输入参数：istream &is 输入流， T &obj 模板类

输出参数：void

实现功能：输入函数,如果输入不符合要求就一直输入直到正确为止

### 5.4类声明

**性别类：**

class Gender

{

public:

friend istream& operator >>(istream &is, Gender &g)

{

cout << "男|女|第三性别 格式:(male|female|third)" << endl;

input(is, g);

return is;

}

friend ostream& operator <<(ostream &os, const Gender &g)

{

if (g.gender == 0) os << "男";

if (g.gender == 1) os << "女";

if (g.gender == 2) os << "第三性";

return os;

}

friend bool operator ==(const Gender &lhs, const Gender &rhs)

{

return lhs.gender == rhs.gender;

}

bool check()

{

buf = trim(buf);

if (buf.empty()) return throw "输入不能为空", false;

if (buf == "male") return gender = 0, true;

if (buf == "female") return gender = 1, true;

if (buf == "third") return gender = 2, true;

throw "输入了奇奇怪怪的性别";

return false;

}

private:

int gender; // 0 男 1 女 2 第三性别

};

**身份证号类：**

class IdNumber

{

public:

friend istream& operator >>(istream &is, IdNumber &in)

{

cout << endl;

input(is, in);

return is;

}

friend ostream& operator <<(ostream &os, const IdNumber &in)

{

return os << in.id;

}

friend bool operator ==(const IdNumber &lhs, const IdNumber &rhs)

{

return lhs.id == rhs.id;

}

bool check()

{

id = trim(buf);

if (id.empty()) return throw "输入不能为空", false;

if (id.size() != 15u && id.size() != 18u) return throw "位数不对", false;

if (regex\_match(id, regex("^[1-9]\\d{5}\\d{2}((0[1-9])|(10|11|12))(([0-2][1-9])|10|20|30|31)\\d{3}$")) ||

regex\_match(id, regex("^[1-9]\\d{5}(18|19|([23]\\d))\\d{2}((0[1-9])|(10|11|12))(([0-2][1-9])|10|20|30|31)\\d{3}[0-9Xx]$")))

return true;

throw "输入错误";

return false;

}

private:

string id;

};

**电话号码类：**

class PhoneNumber

{

public:

friend istream& operator >>(istream &is, PhoneNumber &pn)

{

cout << endl;

input(is, pn);

return is;

}

friend ostream& operator <<(ostream &os, const PhoneNumber &pn)

{

return os << pn.number;

}

friend bool operator ==(const PhoneNumber &lhs, const PhoneNumber &rhs)

{

return lhs.number == rhs.number;

}

bool check()

{

number = trim(buf);

if (number.empty()) return throw "输入不能为空", false;

// if (number.size() != 11u) return throw "位数不对", false;

if (regex\_match(number, regex("^(0\\d{2}-\\d{8}(-\\d{1,4})?)|(0\\d{3}-\\d{7,8}(-\\d{1,4})?)$")) || // telephone

regex\_match(number, regex("^((13[0-9])|(14[5|7])|(15([0-3]|[5-9]))|(18[0,5-9]))\\d{8}$"))) // cellphone

return true;

throw "输入错误";

return false;

}

private:

string number;

};

**车牌号类：**

class NumberPlate

{

public:

friend istream& operator >>(istream &is, NumberPlate &np)

{

cout << endl;

input(is, np);

return is;

}

friend ostream& operator <<(ostream &os, const NumberPlate &np)

{

return os << np.number;

}

friend bool operator ==(const NumberPlate &lhs, const NumberPlate &rhs)

{

return lhs.number == rhs.number;

}

bool check()

{

number = trim(buf);

if (number.empty()) return throw "输入不能为空", false;

// if (regex\_match(number, regex("^[京津沪渝冀豫云辽黑湘皖鲁新苏浙赣鄂桂甘晋蒙陕吉闽贵粤青藏川宁琼使领A-Z]{1}[A-Z]{1}[A-Z0-9]{4}[A-Z0-9挂学警港澳]{1}$")))

if (regex\_match(number, regex("^[A-Z]{1}[A-Z0-9]{4}[A-Z0-9]{1}$")))

return true;

throw "输入错误";

return false;

}

private:

string number;

};

**健康码类：**

class HealthCode

{

public:

friend istream& operator >>(istream &is, HealthCode &hc)

{

cout << "绿|黄|红 格式(g|y|r)" << endl;

input(is, hc);

return is;

}

friend ostream& operator <<(ostream &os, const HealthCode &hc)

{

if (hc.color == 0) cout << "绿";

if (hc.color == 1) cout << "黄";

if (hc.color == 2) cout << "红";

return os << "码";

}

friend bool operator ==(const HealthCode &lhs, const HealthCode &rhs)

{

return lhs.color == rhs.color;

}

bool check()

{

buf = trim(buf);

if (buf.empty()) return throw "输入不能为空", false;

if (buf[0] == 'g' || buf[0] == 'G') return color = 0, true;

if (buf[0] == 'y' || buf[0] == 'Y') return color = 1, true;

if (buf[0] == 'r' || buf[0] == 'R') return color = 2, true;

throw "输入错误";

return false;

}

private:

int color; // 0 绿 1 黄 2 红

};

**字符串类：**

class String

{

public:

friend istream& operator >>(istream &is, String &s)

{

cout << endl;

input(is, s);

return is;

}

friend ostream& operator <<(ostream &os, const String &s)

{

return os << s.str;

}

friend bool operator ==(const String &lhs, const String &rhs)

{

return lhs.str == rhs.str;

}

bool check()

{

str = trim(buf);

if (str.empty()) return throw "输入不能为空", false;

return true;

}

private:

string str;

};

## 6.系统逻辑设计

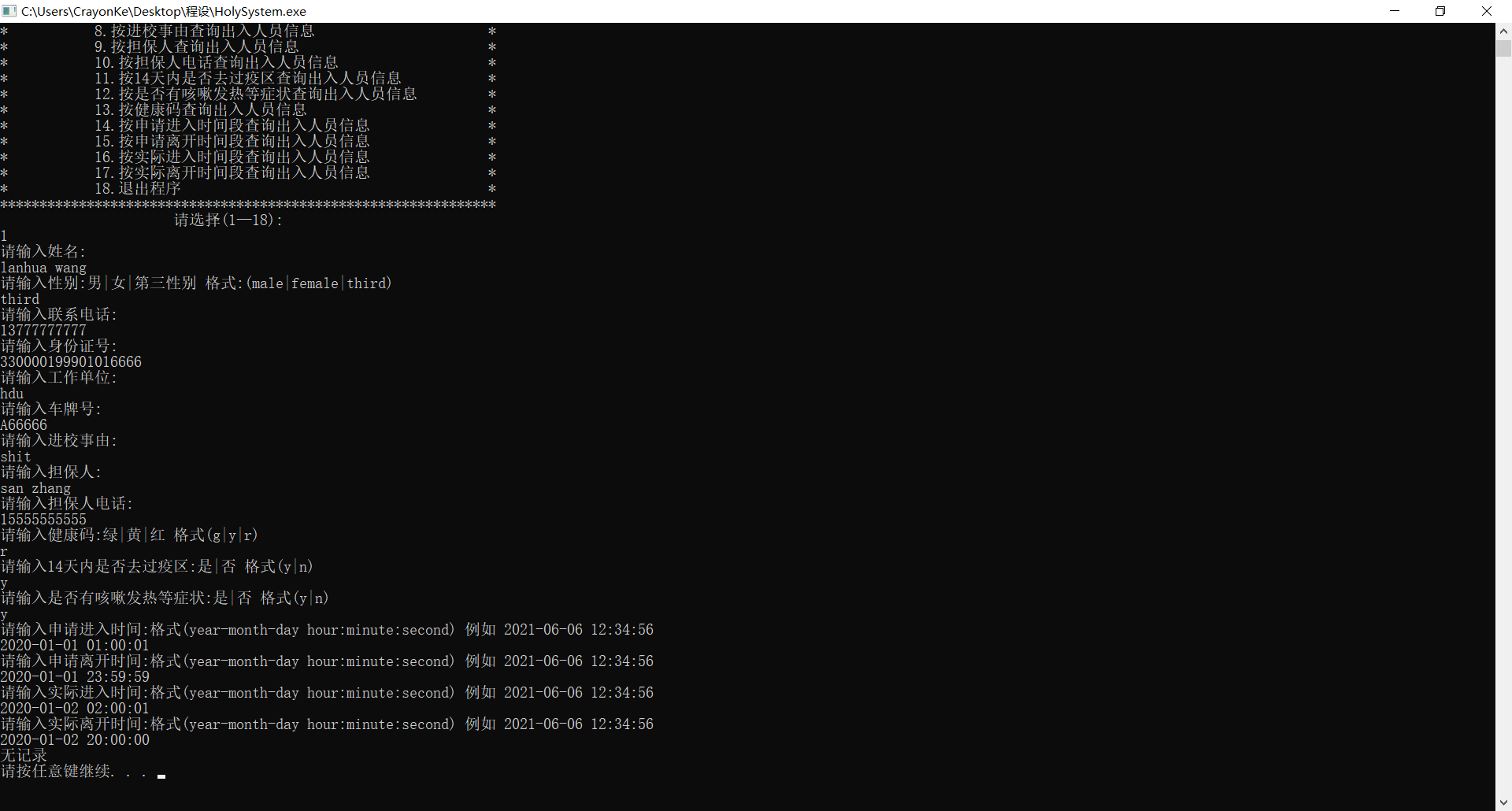
菜单界面输入1后进入添加记录界面，根据提示输入各条信息后，返回菜单；

菜单界面输入2~17后进入各种条件查询记录界面，根据提示输入各条信息后，返回菜单；

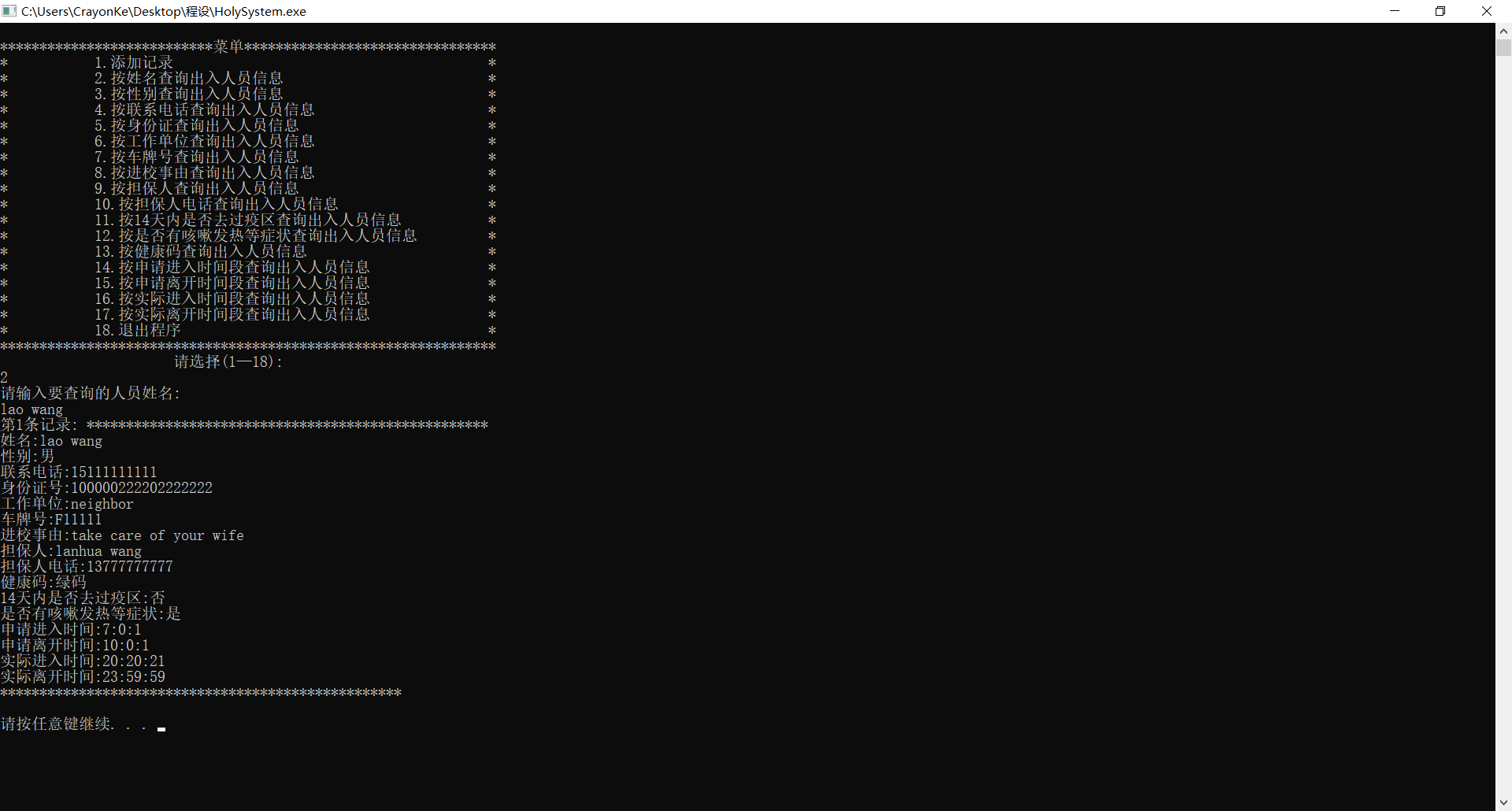
菜单界面输入18后退出程序。

## 7.项目运行效果

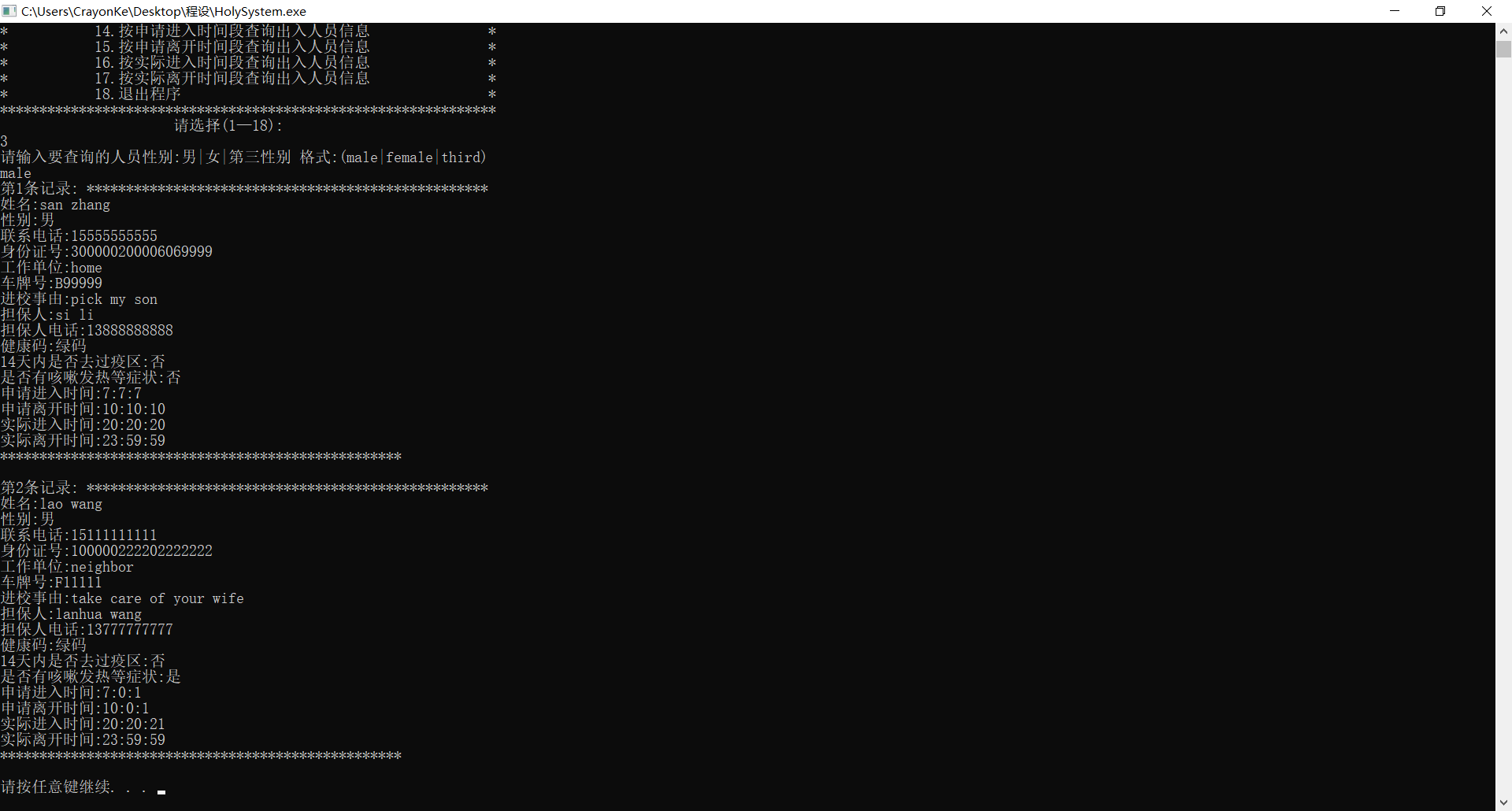
1. 添加记录



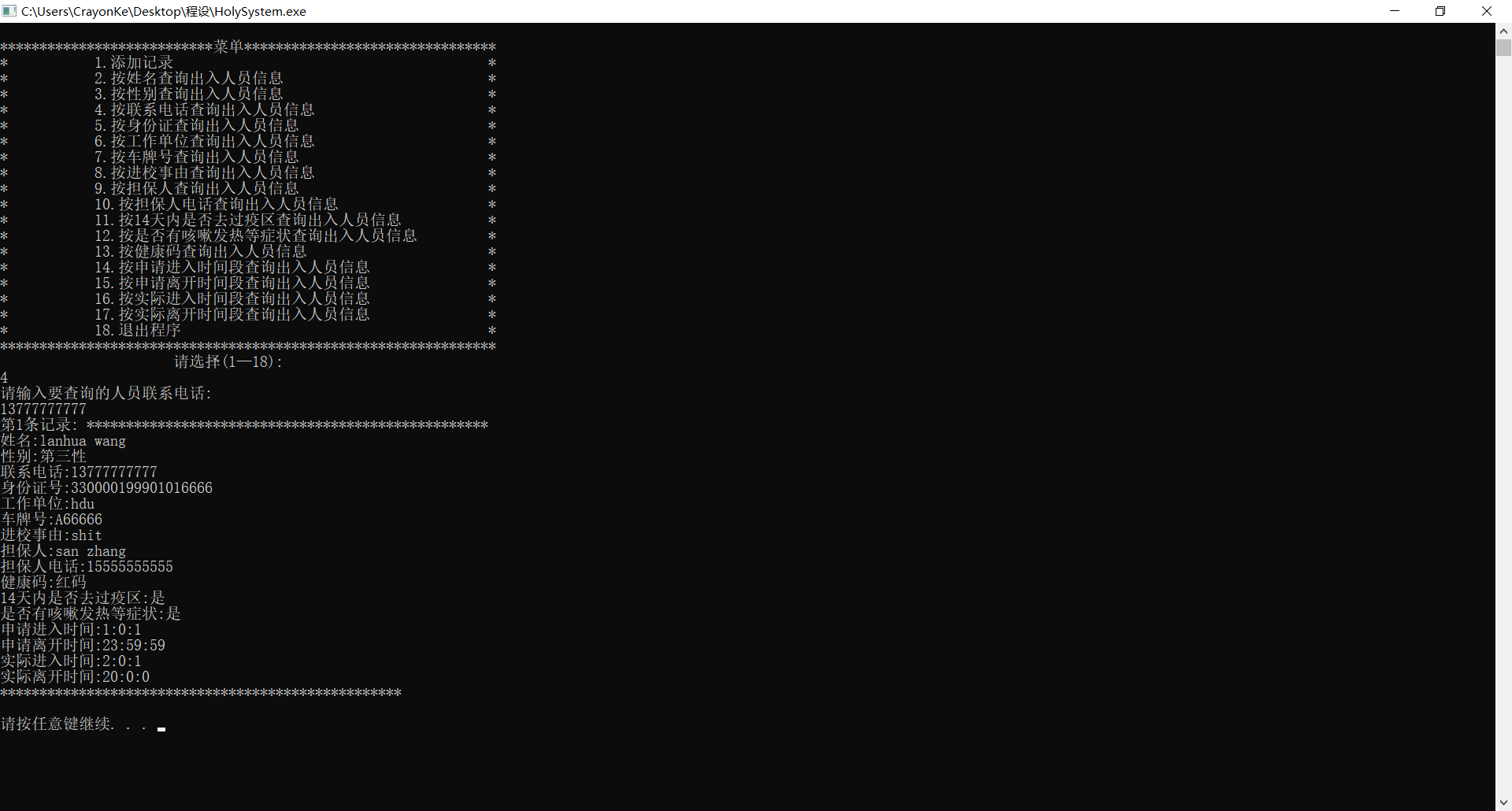
1. 按姓名查询出入人员信息



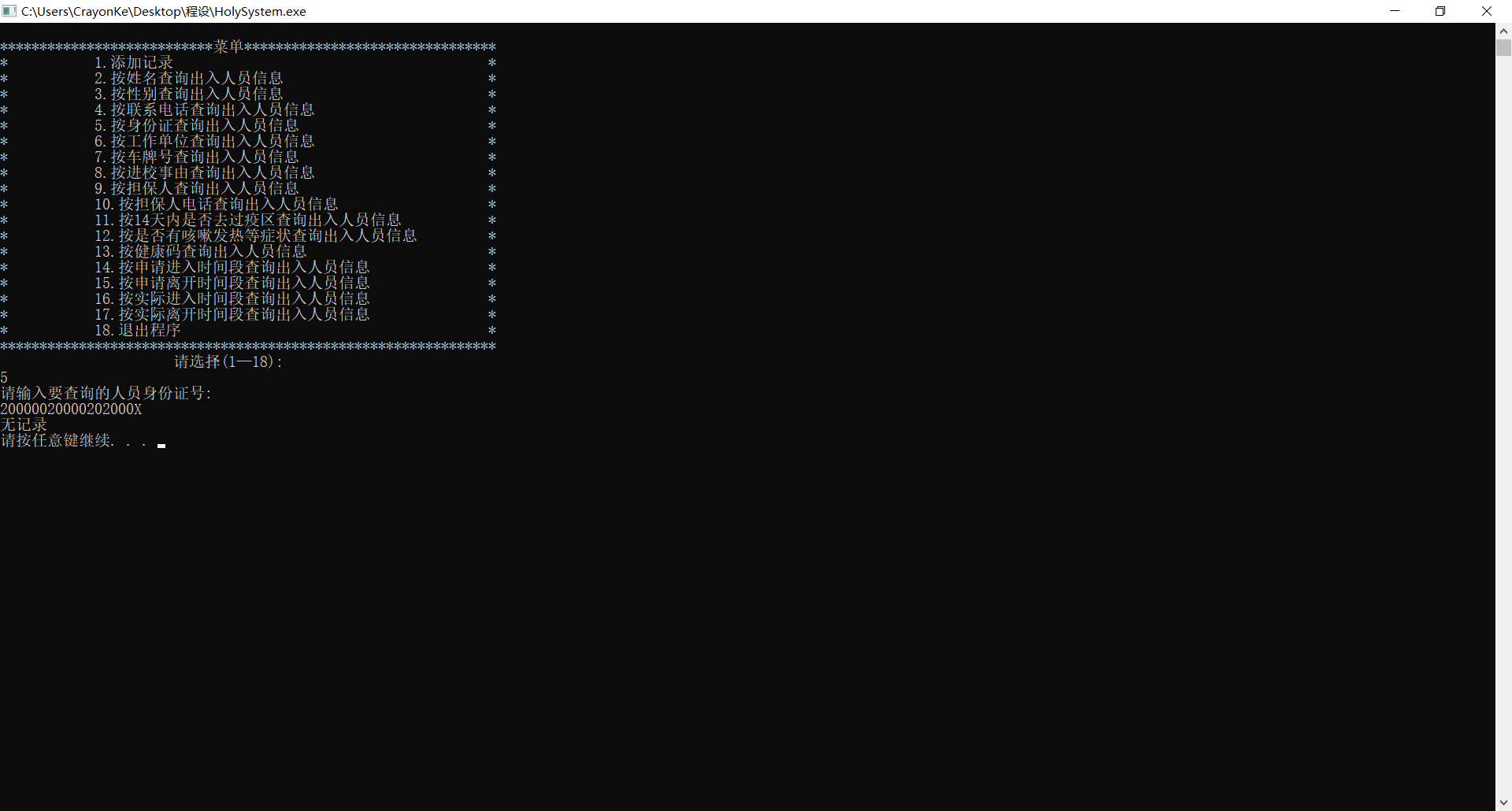
1. 按性别查询出入人员信息



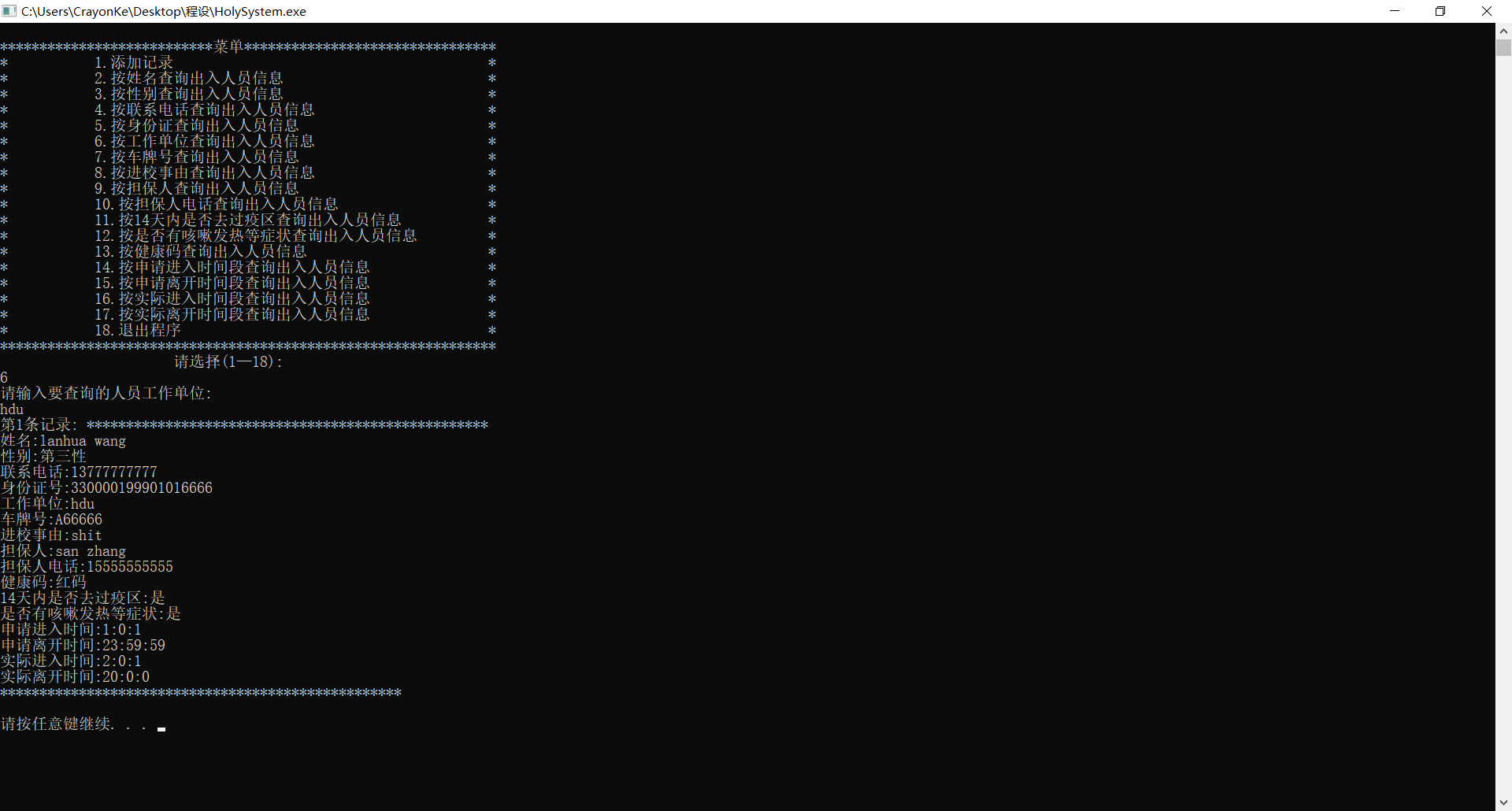
1. 按联系电话查询出入人员信息



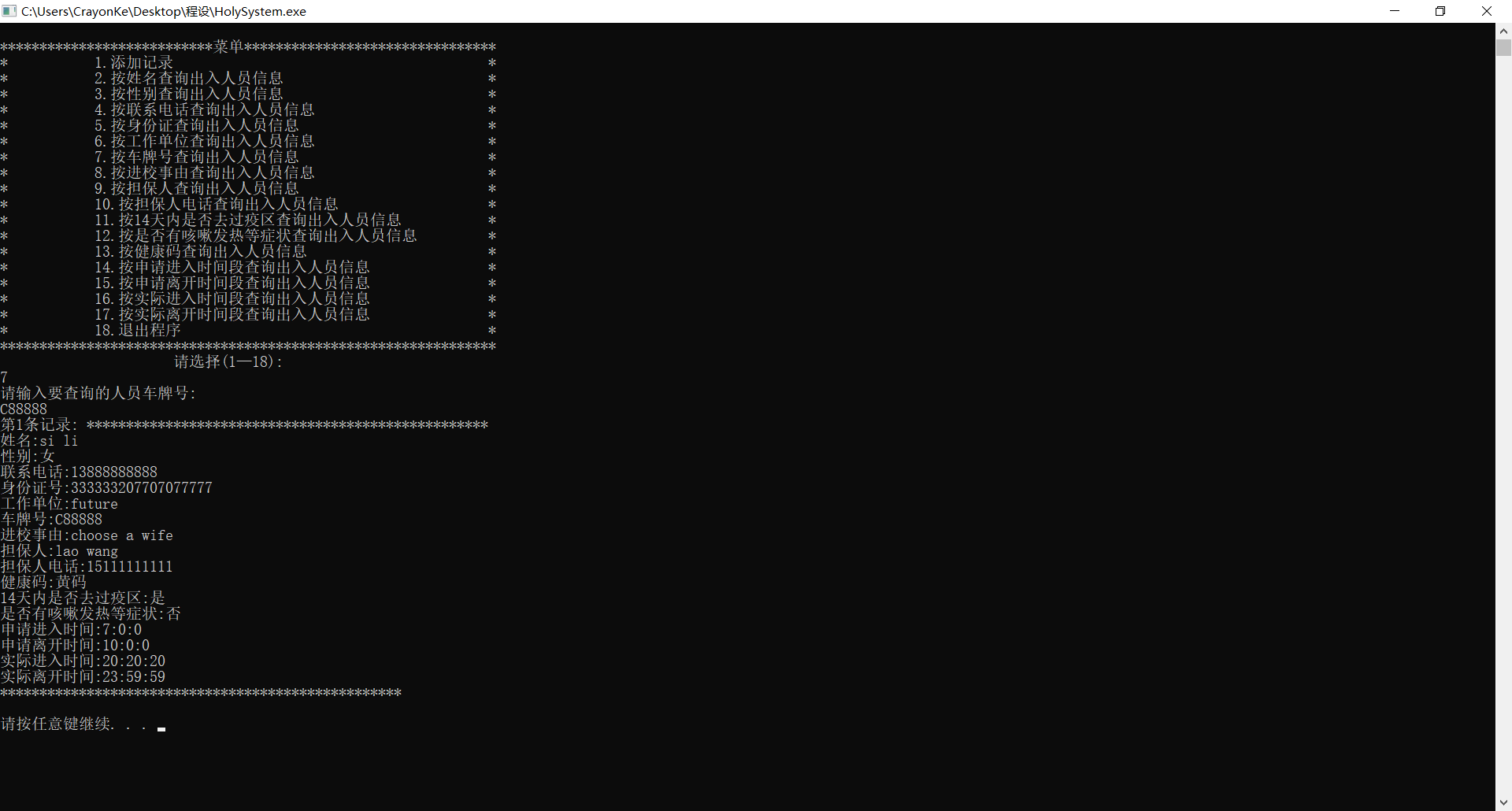
1. 按身份证查询出入人员信息



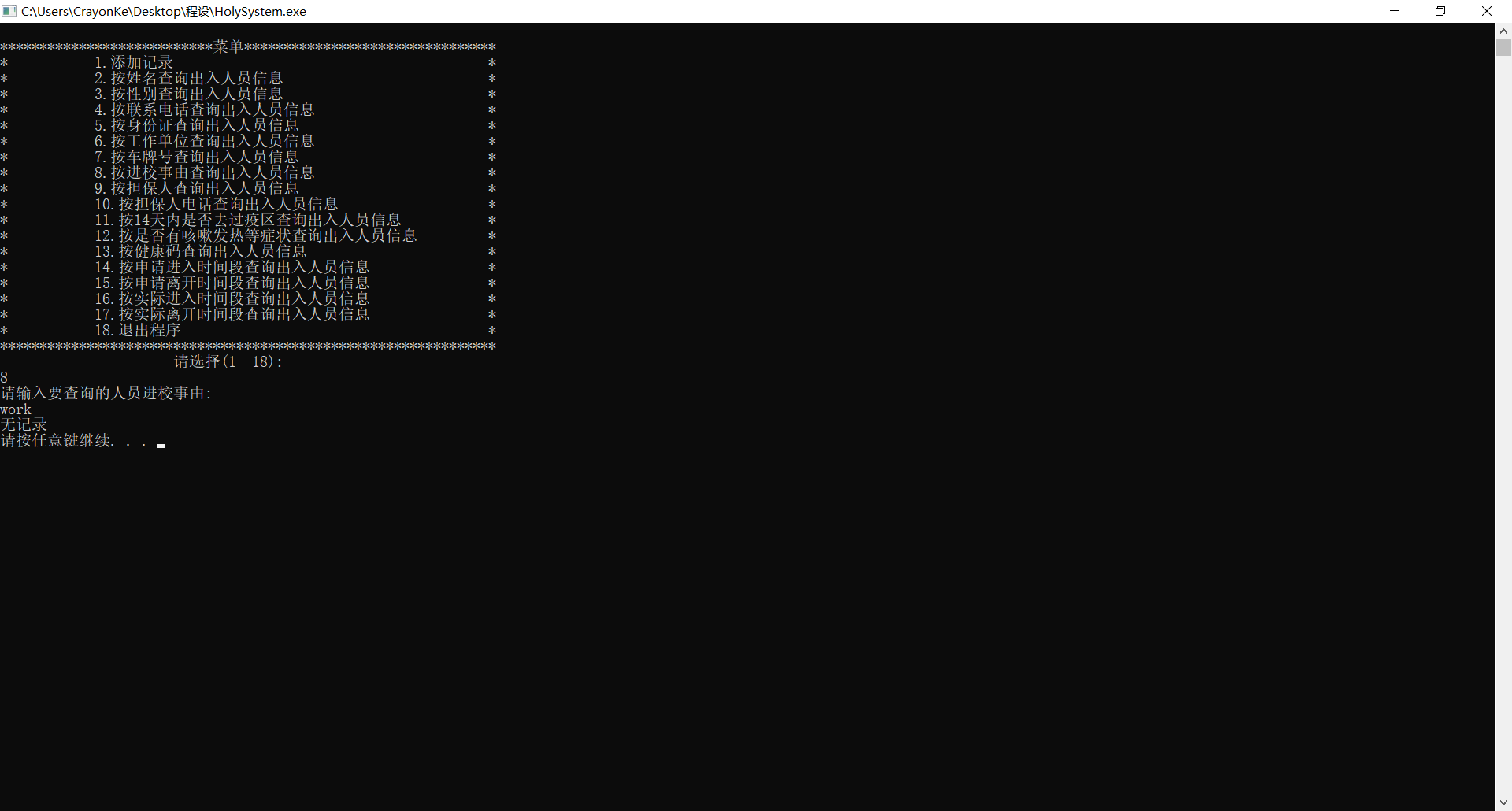
1. 按工作单位查询出入人员信息



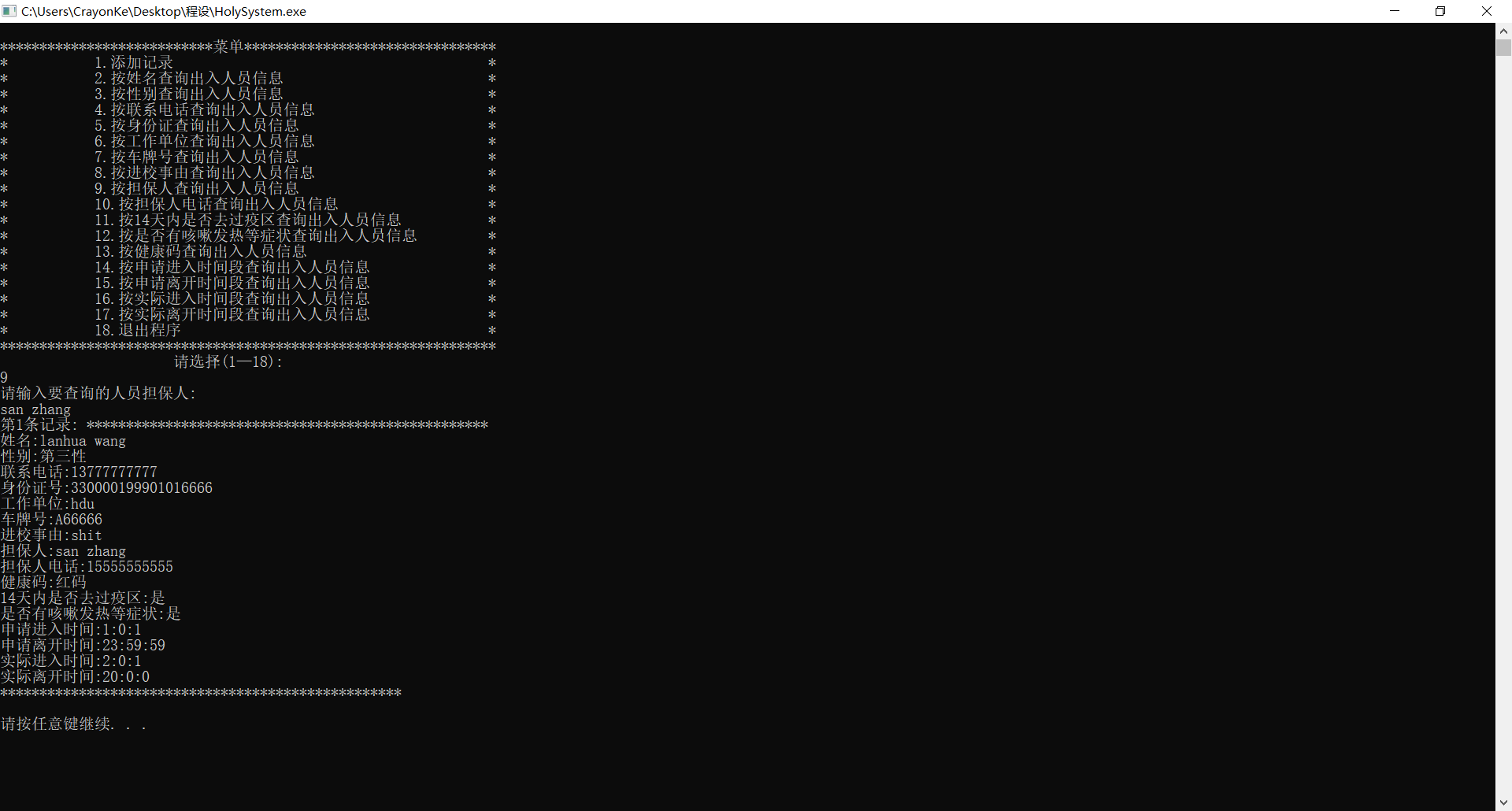
1. 按车牌号查询出入人员信息



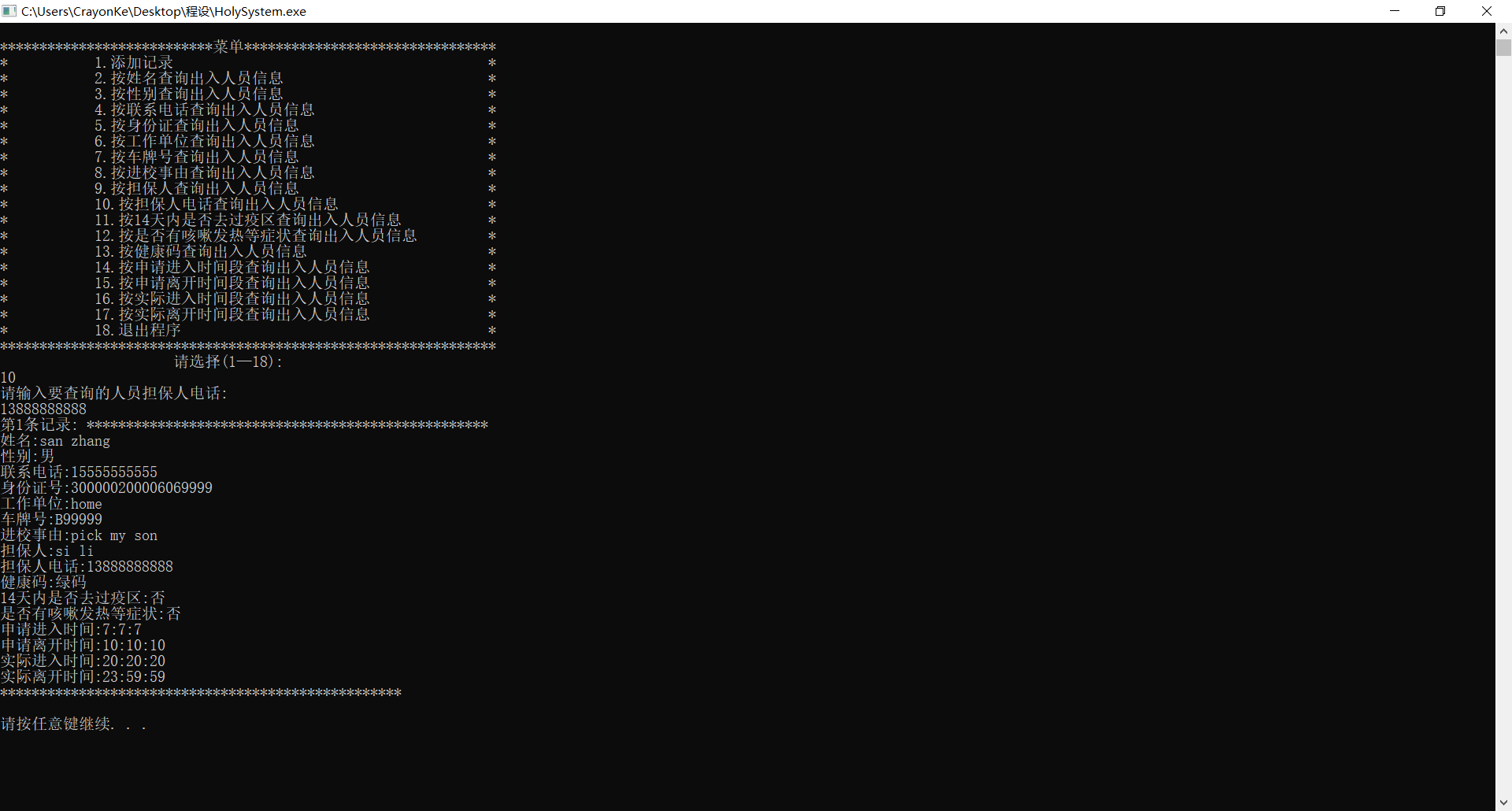
1. 按进校事由查询出入人员信息



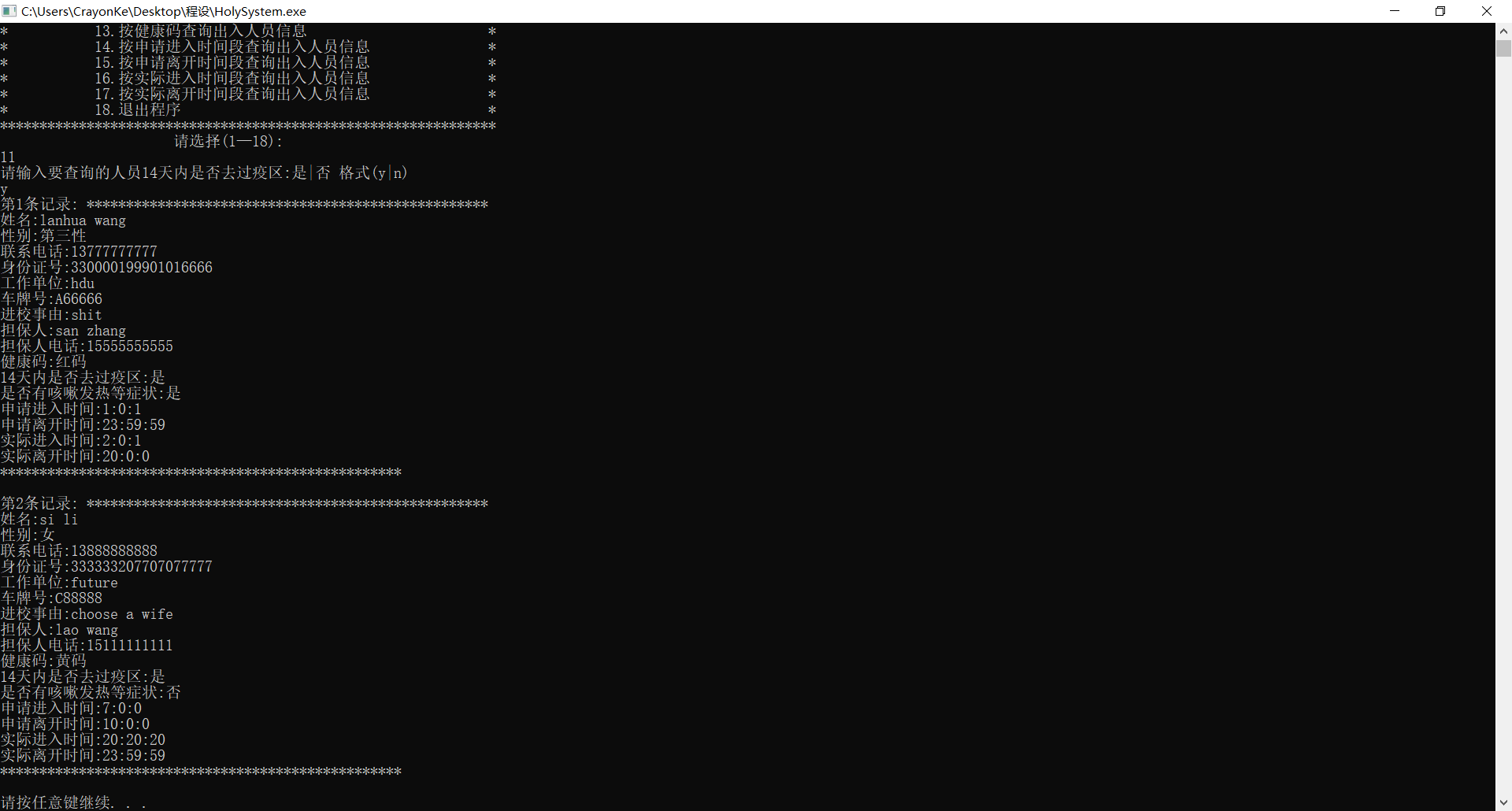
1. 按担保人查询出入人员信息



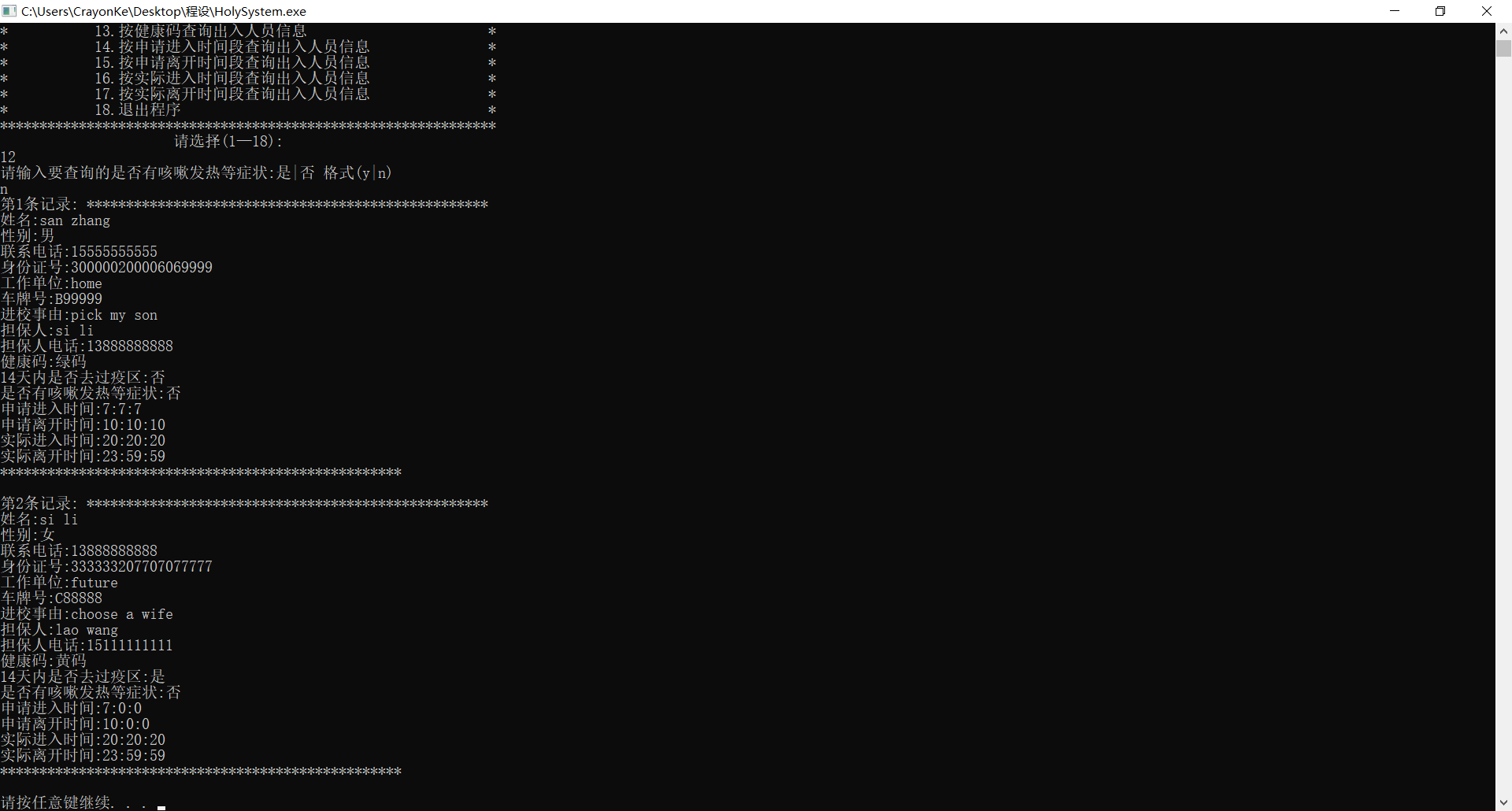
1. 按担保人电话查询出入人员信息



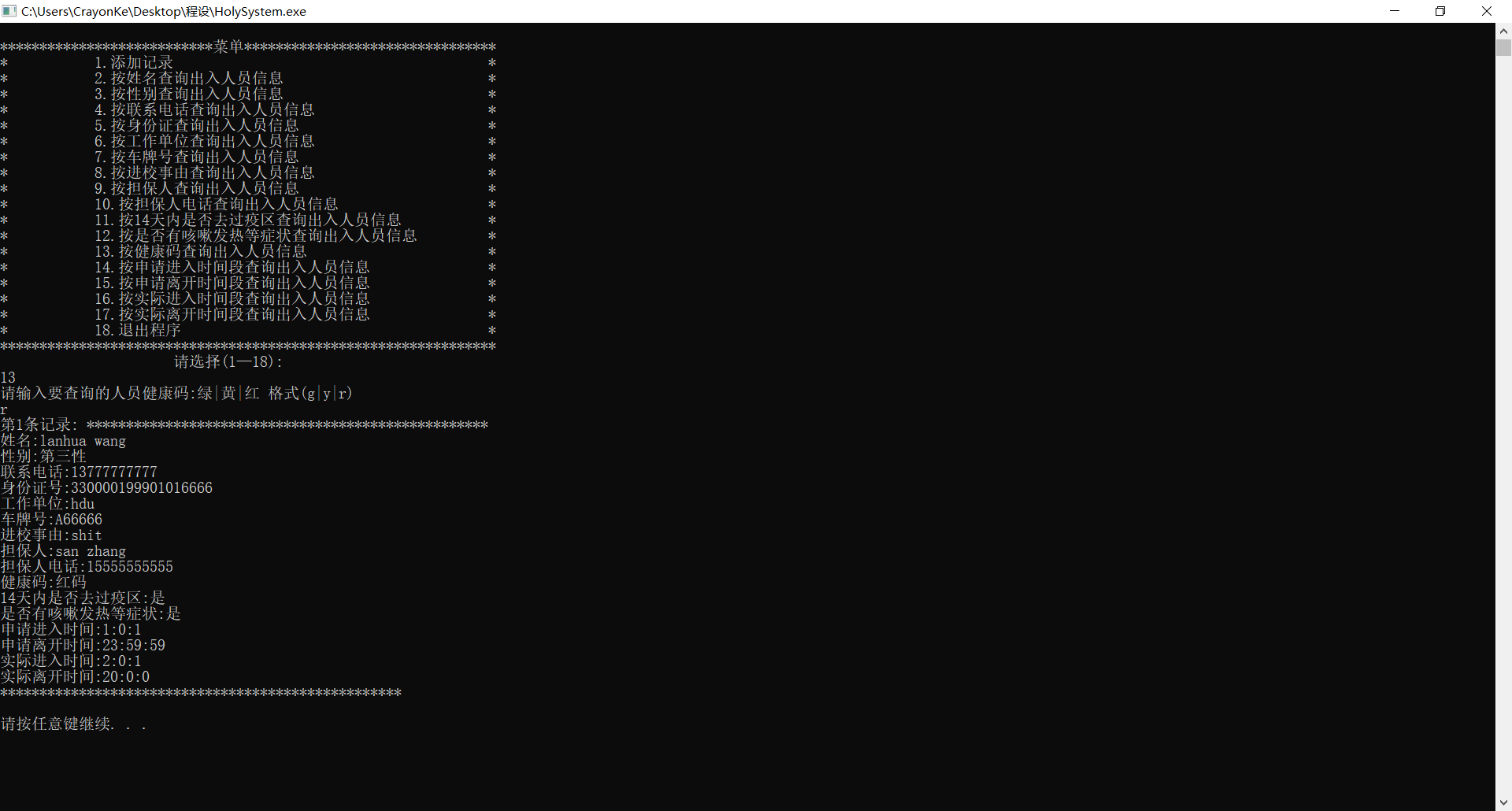
1. 按14天内是否去过疫区查询出入人员信息



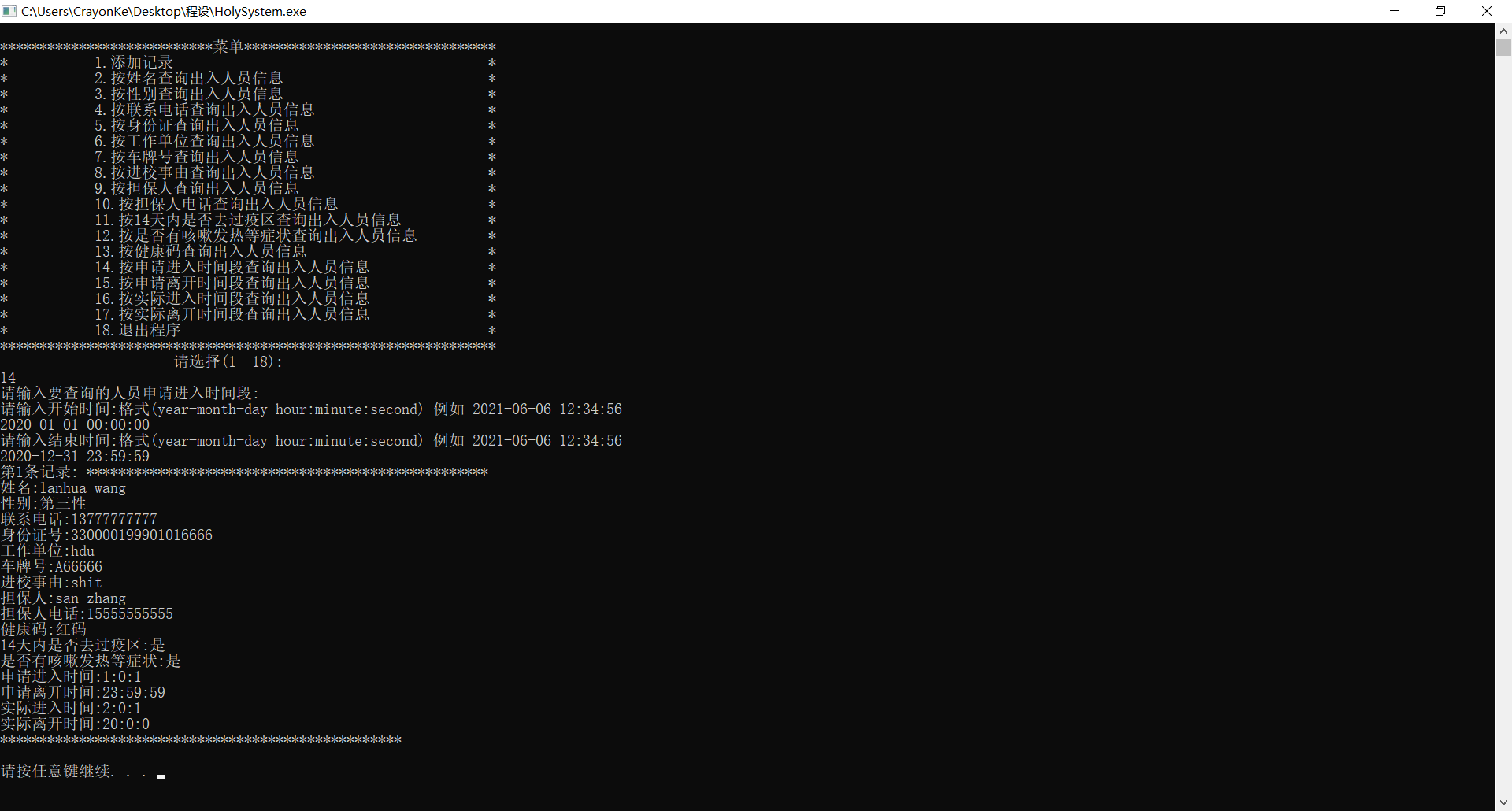
1. 按是否有咳嗽发热等症状查询出入人员信息



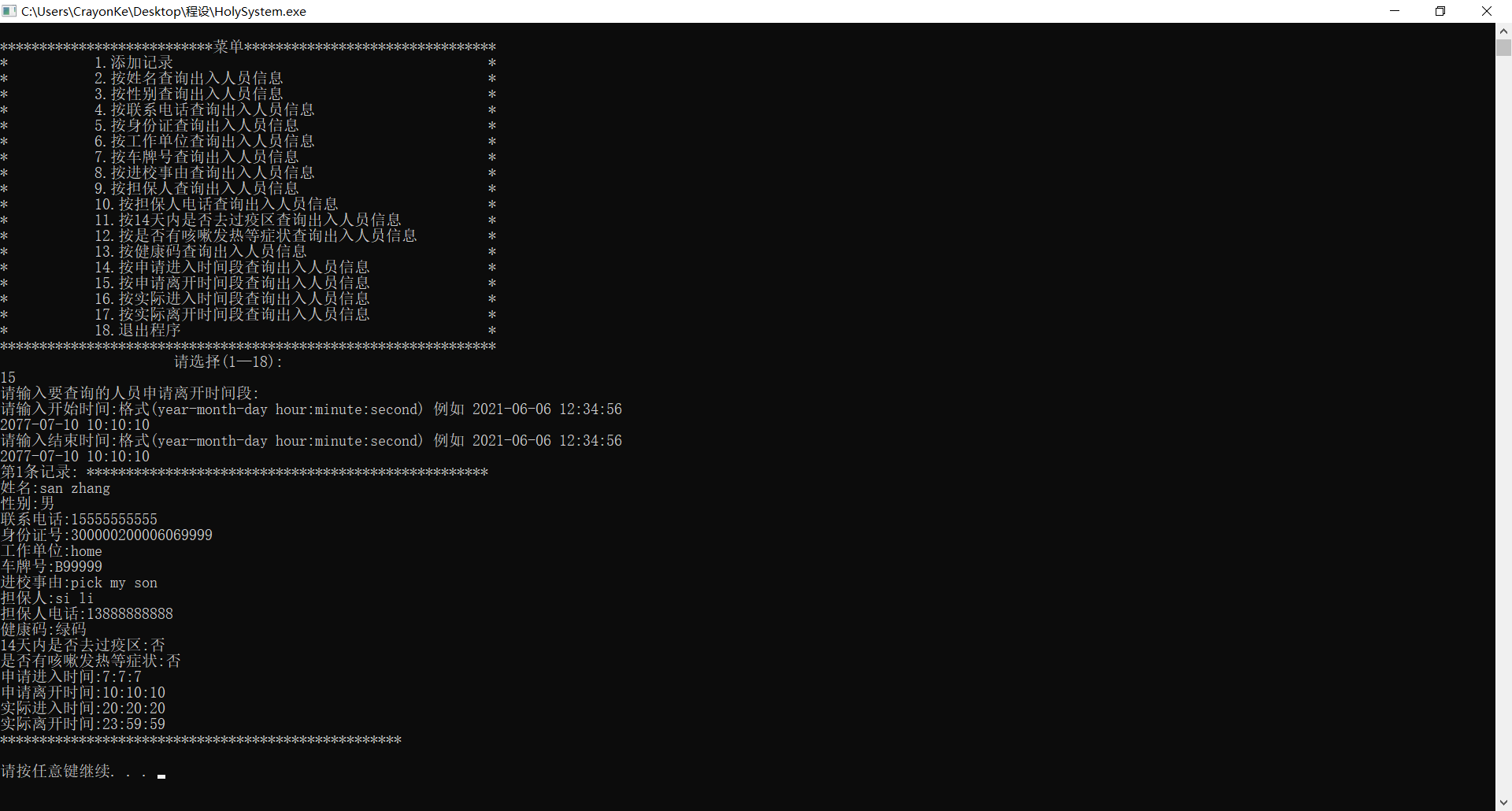
1. 按健康码查询出入人员信息



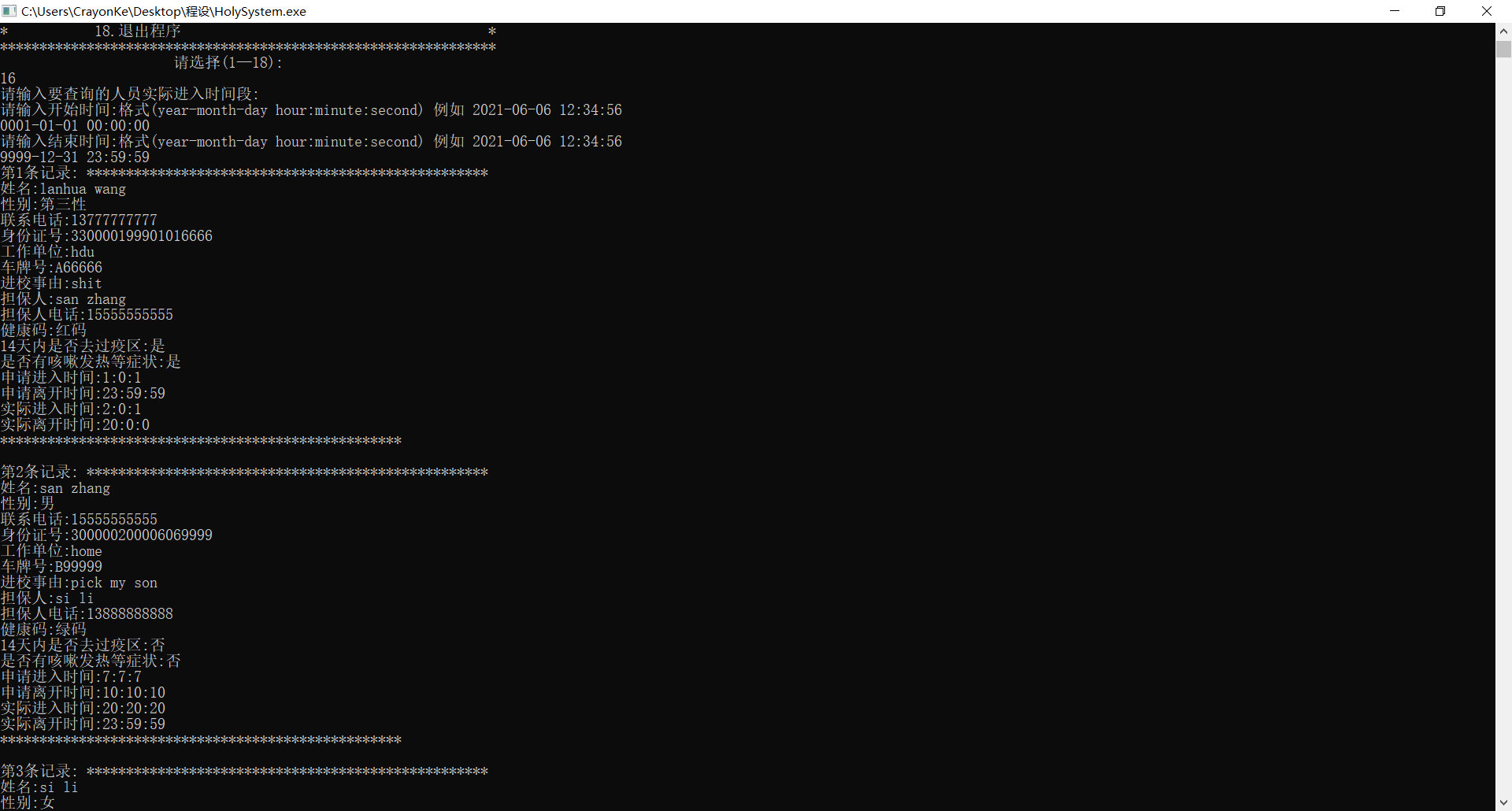
1. 按申请进入时间段查询出入人员信息



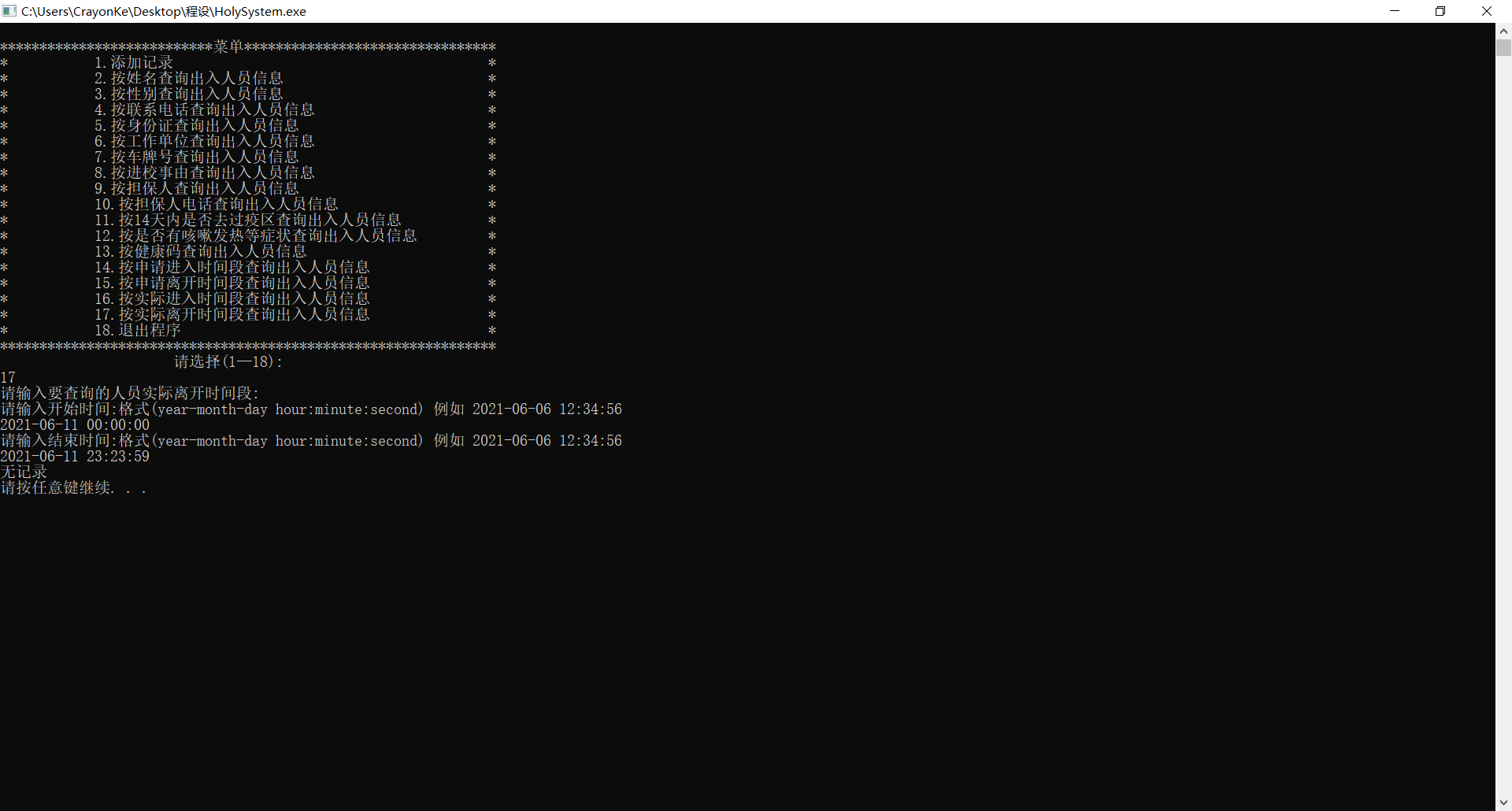
1. 按申请离开时间段查询出入人员信息



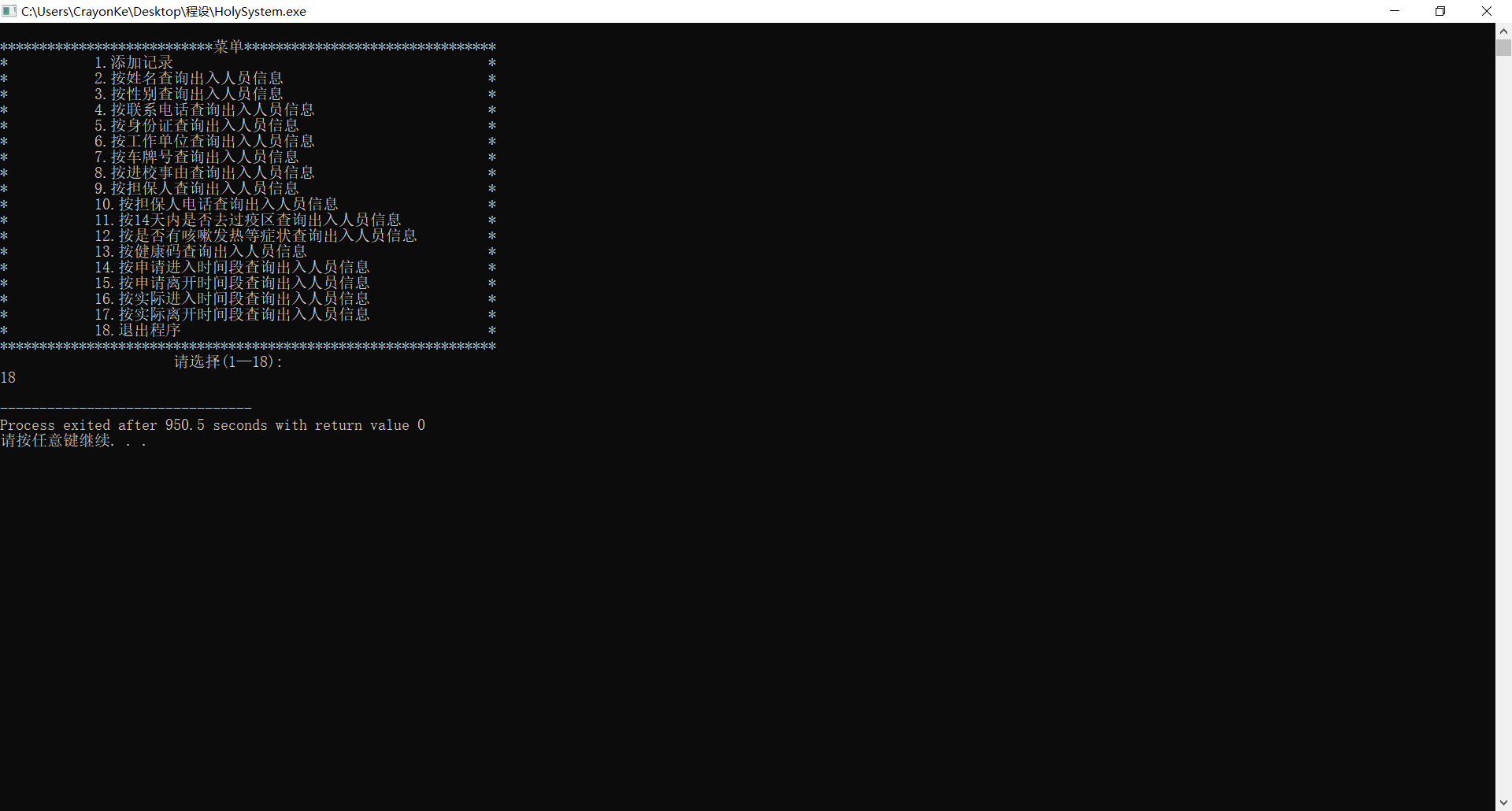
1. 按实际进入时间段查询出入人员信息



1. 按实际离开时间段查询出入人员信息



1. 退出程序



## 8.项目创新点

输入输出有接口函数

将不同的个人信息封装成类，代码实现更加简洁、清晰

## 9.收获和建议

组长 郑凯心：这次实践使我认识到了C程序设计是一门实践性很强的学科，我们学校C不仅仅是为了学习而学习，还能锻炼我们的创新能力，合作精神等等，我们不仅要按照要求完成作业，还要在设计时加上自己的想法及理解，开始是以为做着会很简单，但真正做的时候才发现真的很不容易，要完成这项工作真的要耐心，细心的去分析，设计。 我们小组根据题目经过研讨会后明确的进行了分工。不得不说这一题很繁琐，因为需要很多信息的输入与查询，但是在做的过程中则发现其实规律性很大，慢慢的就会感到轻松，不在会觉得枯燥，但是在编程之后的调试又遇到了问题，那么长的程序不可避免的会遇到错误，在改得过程中则不再有规律性可言，要慢慢的分析，有的地方实在理解不了要去查书，问同学，一处处的去修改，经过无数次的调试及修改，我终于完成了这次编程。我深刻的体会到编程的艰辛与困难，确实，实践是检验真理的唯一标准。之前学习C时，大部分时间都用在听课和看书上，即使上机也是一些简单的程序设计，所以缺少了亲自去动手编写程序的经历，对C程序设计并不熟练，对一些基础知识也只是机械式的记忆，并没有什么深刻的印象。但是在这次编写程序过程中，出现的许多错误，令我对书本有了更加深刻的印象，那么多的错误有的是因为自己的不小心，少了标点，设置错误的变量类型，但是通过自己的反复检查，这些简单的错误在后来的编程中较少出现，可见C的实践可以加深对书本的理解及记忆，对于错误的修改也让我体会到细节的重要性。

这么大型的程序我是第一次编写，但是完成之后我明白了，其实大型的程序并没有想象中的那么复杂，它们也是由一个个的小程序组合而成的看到自己的劳动成果真的很高兴，之前的烦躁也变得是那么值得，这次的编程对我的耐心真的是一次很大的锻炼，，平时我们学C程序设计语言几乎都是为了考试或者做一些比较常规的题目，而这次程序设计实践却给我们的将所学知识运用于解决实际问题的机会，让人更有激情，这就是编写软件的雏形，也让我们感受到了编写软件程序的乐趣。但是不仅仅是如此，更为重要的是在运用所学知识的过程中，记住这些知识，并能够广泛的运用！在过程中有好几次都想放弃，但最后都坚持了下来，当最后结束工作时，欣喜不只是以为C作业的完成，还因为这次编程让我学到了很多，对自己更加有信心也不再觉得C离我们那么遥远，其实它就在我们生活中的方方面面。

成员1 蔡思林：程序设计实践这门课程是锻炼我们学生动手能力而设的。我在课程内学习并设计了链表、栈、队列、排序、线性回归、聚类及BP神经网络。其中在学习完堆栈、队列后可以利用深度优先搜索和广度优先搜索解决迷宫问题和马踏棋盘问题，也了解了贪吃蛇的设计原理，增加了趣味性，让我对程序设计更感兴趣了。学习BP神经网络也是充满挑战，要推数学公式，在一次又一次失败之后看到最后模型训练成功体现了程序的魅力，让我加深理解了全连接层底层的工作。从基础来讲，实践中发现了与理解的偏差，学习程序语言不能停留在词法、语法的层次，而是运用这些工具解决问题；从提升方面说，实践课帮助我们把问题归类，用抽象的数据结构或类去解决具体的问题。程序设计要摒弃一些固有的思维，试着用计算机的思维去看问题，传统的思路表现在程序上往往是复杂且难以实现的。学习各种算法能很快地提升解决问题的能力。自己在设计编程时还会遇到各种各样的bug，这些小错误让我纠正了以前的错误认知，并更加了解了一些算法在细节上的巧妙。还有排序那章让我重新整理了各种排序方法，计算各种算法的复杂度，让我在以后运用排序时更能针对不同问题能选择更优的排序算法。总的来说，本学期的程序设计课程实践难易适中，即有巩固又有提升，也为后续学习更深更专业的知识推开了一道门缝，让我收获颇丰。

成员2 柯瑞阳：在这几周的时间，通过我们小组各成员之间的相互讨论和合作，我们完成了校园外来人员进出监控与管理系统的程序设计。虽然我只负责其中的时间报告部分，但我感受到我对C语言有了更深一步的了解，掌握了更多的技巧和技能。

在这次课程设计中，我们首先对系统的整体功能进行了构思，然后用结构化分析方法进行分析，将整个系统清楚的划分为几个模块，再根据每个模块的功能编写代码。而且尽可能的将模块细分，最后在进行函数的调用。这次设计中，我还学会了用流程图来表达自己的想法，并根据流程图来逐步实现程序的功能。开始的时候，我画流程图很是困难，需要比较长的时间才能清楚的根据自己的想法画出图来，后来画多了，就更加了解它的功能，十分得心应手，能够比较快而准确地画出来。

C语言是计算机程序设计的重要理论基础，在我们以后的学习和工作中都有着十分重要的地位。要学好这种语言，仅仅学习课本上的知识是不够的，还要经常自己动手，有较强的实践能力。只有多动手，经常编写程序，才能发现我们学习上的漏洞和自己的不足，并在实践中解决这些问题，不断提高自己转化知识的能力。

## 附：源代码清单（每个部分写明是哪位同学完成）

<HolySystem.cpp>（郑凯心、蔡思林）