

算法与数据结构设计报告

（ 2018 / 2019学年 第 一 学期）

题 目： 实验室安排系统

|  |  |
| --- | --- |
| **专 业** | **计算机科学与技术** |
| **学 生 姓 名** | **成楚凡** |
| **班 级 学 号** | **B16040615** |
| **指 导 教 师** | **黄海平** |
| **指 导 单 位** | **计算机科学与技术系** |
| **日 期** | **2018.10.29-2018.11.11** |

|  |  |
| --- | --- |
| **评价准则** | **计分（每项总分10分）** |
| 1、能够掌握算法与数据结构设计的相关基础知识，并能够针对求解的工程问题，进行合理的分析与设计（3-1） |  |
| 2、能够选择合适的程序设计语言与编程开发平台，对求解的工程问题进行编程实现（3-1） |  |
| 3、能够给出数据结构和算法的设计描述，给出关键算法的流程图或伪代码，并给出各算法之间的结构关系描述 （3-1） |  |
| 4、具备一定的人机交互设计意识，人机交互设计合理、友好，操作简便（3-1） |  |
| 5、具备一定的算法与数据结构设计分析能力，能够完成课题要求的各项任务和指标（4-1） |  |
| 6、能够结合计算机软硬件资源，合理选用算法、数据结构、数据存储方式等技术手段，对求解的工程问题进行有效建模和求解（4-1） |  |
| 7、具备一定自学能力与探索创新意识，能够充分利用教科书及其资源（如网络等）自学新知识与新技能（4-1） |  |
| 8、掌握调试方法与工具，对程序开发过程中出现的问题进行分析、跟踪与调试，并能够进行充分测试（4-1） |  |
| 9、能够正确、完整地回答指导教师关于课题的问询，反映其对课题内容，以及相关的工程基础知识具有较好的理解和掌握（4-2） |  |
| 10、具备一定的语言表达能力与文字处理能力，能够结合复杂工程问题撰写报告，报告内容详实，格式规范（4-2） |  |
| 算法与数据结构设计能力测评总分 |  |
| **指导教师： 年 月 日** | |
| **备注：** | |

**具体课题题目**

**一、课题内容和要求**

为《算法与数据结构设计》课程开发实验室安排系统。该课程开课时间为两周，在两周内需要上4-8次课，每次连续4个学时。每天上午下午可各安排一次课，周末不安排上课。

在实验室安排系统中包含四大功能：

1. 实验室信息录入功能：可录入现有的实验室信息，包括实验室地址、实验室所能容纳学生数量；
2. 实验课申请录入功能：可录入申请班级ID、班级人数、教师期望的实验安排（例如选择第一周的周一至周五上午，第二周周一上午）和不期望的实验安排；
3. 课表生成功能：实验室安排算法读入已有申请和实验室信息，生成总课程安排，并写入数据库；
4. 课表查询功能：给定实验教室，查询实验安排；给定班级，查询实验安排。
5. 数据库可随意选择。

基本要求：

(1) 算法生成的课程安排，要求尽可能为每个老师安排课程，所安排实验室能容纳所上课班级；按照教师申请的时间次序，尽可能优先满足先申请教师的期望实验安排；当申请较多而管理员录入的实验室较少，允许出现有的教师申请无法满足的情况；

(2) 实物演示时要求能够说明所采用算法思想；

(3) 程序操作友好、健壮；操作界面简洁美观。

提高要求：

(1) 实验课申请批量录入功能，例如导入excel表格；

(2) 先进而高效的算法：使用最少的教室满足最多的需求；

(3) 界面美观，课表整齐美观。

**二、课题需求分析**

本课题目标系统“实验室安排系统”的功能框架图如图1所示。

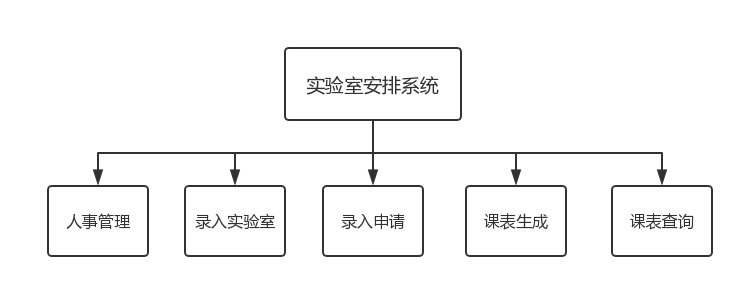


图1 功能框架图

（1）人事管理：用于登录认证，修改个人信息和密码；

（2）录入实验室：录入实验室地址和座位；

（3）录入申请：提交老师，班级以及老师期望和不期望的上课时间，支持excel批量导入；

（4）课表生成：根据上述实验室和申请使用回溯法找出可行方案；

（5）课表查询：显示一张课程表，可以根据关键字查询课程安排。

我将两周的时间分为20个时间片，以老师的期望时间为基础来给他们安排合适的时间片来开展教学活动。时间片的排布如下（时间片编号为019）：

表1 时间片排布表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 课节数 | 周一 | 周二 | 周三 | 周四 | 周五 |
| 第九周上午 | 0 | 1 | 2 | 3 | 4 |
| 第九周下午 | 5 | 6 | 7 | 8 | 9 |
| 第十周上午 | 10 | 11 | 12 | 13 | 14 |
| 第十周下午 | 15 | 16 | 17 | 18 | 19 |

**三、课题相关数据结构及算法设计**

1 主要数据结构

1.1定义JTable对应的存取Model

//由于本实验处理的主要对象是数据库和JTable表格，使用该类从数据库向表格同步信息

public class EmpModel extends AbstractTableModel {

public Vector<String> colums; //对象的列

public Vector<Vector> rows; //对象的行

public boolean UpdateModel(String sql, String[] params){

…… //传入相应的SQL语句和数值更新数据库

}

……

}

1.2获取每个时间片的申请数据

public class App\_Super {

private String classid; //班级ID

private String hope; //老师在该时间片的期望

private EmpModel em\_stu; //学生信息

private EmpModel em\_lab; //实验室信息，查看实验室能否容纳该班学生

…… //getter和setter

}

1.3申请数据和申请状态

public class BaseInfo { //获取基本信息

public Vector<App\_Super> s; //将App\_Super封装成集合

public int [][] state; //判断老师的选择，是否有漏选的周

…… //getter和setter

}

2 算法思路

本题主要使用回溯法进行排课，当确定该时间片能否为老师排课时将用到以下显式约束和隐式约束：

显式约束：

（1）同一班级在同一时间只能安排一门课程；

（2）同一教师在同一时间只能安排一门课程；

（3）同一教室在同一时间只能安排一门课程；

（4）教室总数要大于同一时间安排的课程总数；

（5）教室容量必须大于上课学生人数。

隐式约束：

（1）优先满足先申请老师的要求；

（2）在两周内需要上48次课；

（3）同一门课程尽量安排在固定的教室。

3 主要算法流程

排课算法主要分为两部分：

1. 主调用函数：将申请表和实验室表加载到内存后以深度优先的方式搜索解空间，用Place()函数判断是否可以在该时间片上课。如果可以则将该层该时间片置为1并且将课程数加1，若该层课程数大于8就进入进入下一层。递归返回后再次判断该层课程数，若小于4则在在该层寻找下一个可行的时间片，重复执行上述操作找到可行解后返回。该算法的递归结构流程图如图2所示。

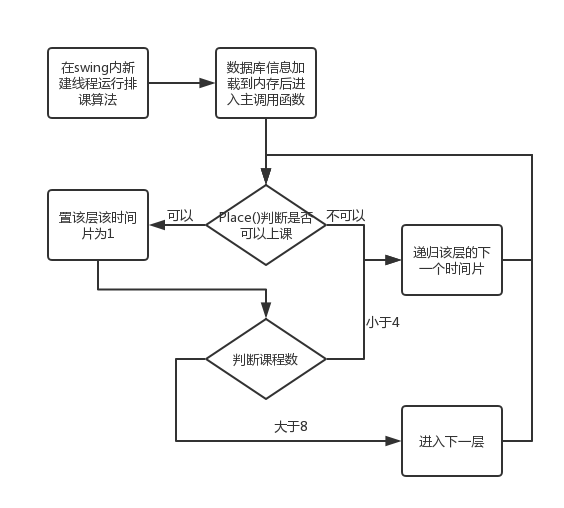


图2 主调用函数算法递归结构流程图

1. Place()判断函数：在搜索过程中用剪枝函数避免无效搜索。1.判断该时间片所选择的实验室座位数是否大于该班级人数；2.判断老师在该时间片的期望；3.判断该时间片有没有别的班级已经占用实验室；4.判断在该时间片是否已经选择了一个实验室。任何一个条件不满足都返回false，都满足就返回true。假设提供4个实验室，对比过程如图3所示（每一级向上判断实验室是否已被占用，向左判断该时间片是否已经选择一个实验室）。

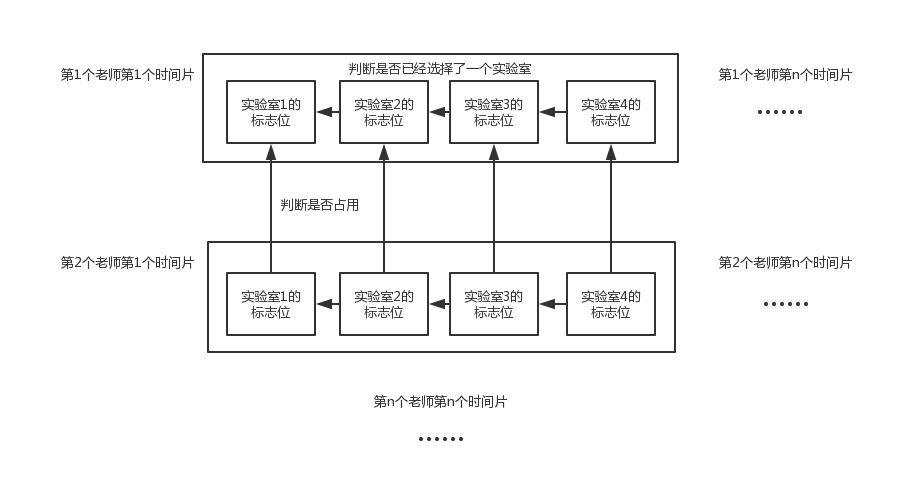


图3 判断函数对比过程图

4 主要模块设计

4.1用户模块：

在用户登录时判断用户类型，对用户进行分权操作，从而确保系统的安全性。例如只有管理员可以修改和删除用户。

4.2数据库连接模块：

为提高代码可读性，将数据库的连接和查询等操作进行封装。在实际运用时只需要传入指定的SQL语句和对应的数值就可以完成数据库的查询和添加等操作。

4.3表格增删查改模块：

表格展示主要由数据库的增删查改进行展示，对每一个模块进行包装以提高代码的重用性。

1. 增加模块：增加模块判断是否已有关键字。

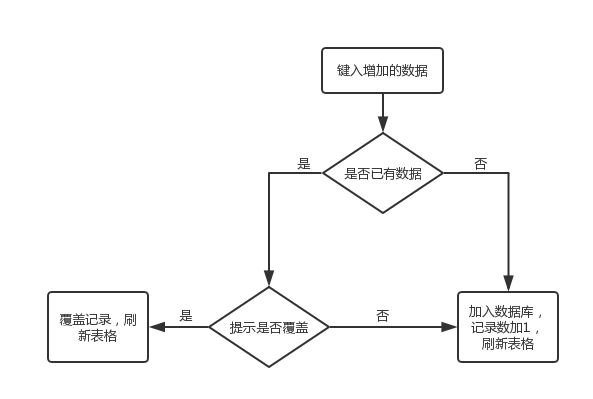


图4 增加模块过程图

1. 删除模块：删除模块会二次提醒是否删除避免删除失误。
2. 查询模块：可以按多个关键字组合查询。
3. 修改模块：和删除模块相同，会事先判断用户权限。
4. 批量导入模块：多次判断数据类型，向用户反馈异常情况。

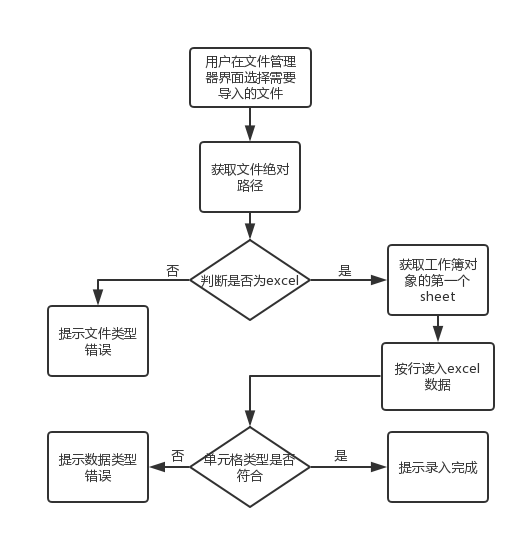


图5 批量导入模块过程图

4.4课表生成模块：

在swing界面操作程序时容易使界面失去响应，所以我将排课算法用线程的形式运行。

4.5课表重组模块：

最后为了展示一张完整的课表，直接从数据库读取课表数据，然后将数据重组成一张课表。

**四、源程序代码**

1 NQueens.java 仿照所学回溯法中NQueens的做法并加以改进，结合实际要求，完成了比较高效的排课算法

package com.view;

import ……

public class NQueens implements Runnable{

JLabel jl2=null;

//该标志位用于提前返回一组可行解

boolean flag=true;

//参数依次为行号，列号,对象信息，时间片号，结果，选择状态

public boolean Place(int r, int l, Vector<App\_Super> Vx, int timepice, int res[][], int state[][]) {

//判断同一列是否有已选，同一时间片是否有已选，学生人数有没有超,其实只要和前面的信息比较就行

//测试每个实验室是位置够并得到实验室编号

//两个标志位都为1说明这两周都申请了

if (state[r][0]==1&&state[r][2]==1) {

int lab = l - timepice \* 4;

//把行号转化为Vector编号

//int a = r \* 20 + timepice;

int a=0;

for(int m=0;m<r;m++){

a=a+state[m][0]+state[m][2];

}

a=a\*10+timepice;

if (Vx.get(a).getStuNum() <= Vx.get(a).getDeskNum(lab)

&& Vx.get(a).getHope().equals("√")) {

//实验室作为数量够

//判断是否和前面冲突

for (int i = 0; i < r; i++) {

if (res[i][l] == 1) {

return false;

}

}

//搜索时间片

for (int j = timepice \* 4; j < l; j++) {

if (res[r][j] == 1) {

return false;

}

}

return true;

}

//只在第10周申请

} else if (state[r][0]==0&&state[r][2]==1) {

//第5组第9周不开课

if (l <= 39) {

return false;

} else {

int lab = l - timepice \* 4;

//把行号转化为Vector编号

int a = 0;

for (int m = 0; m < r; m++) {

a = a + state[m][0] + state[m][2];

}

a = a \* 10 + timepice;

a=a-10;

if (Vx.get(a).getStuNum() <= Vx.get(a).getDeskNum(lab)

&& (Vx.get(a).getHope().equals("√")||Vx.get(a).getHope().equals(""))) {

//实验室够

//判断是否和前面冲突

for (int i = 0; i < r; i++) {

if (res[i][l] == 1) {

return false;

}

}

//搜索时间片

for (int j = timepice \* 4; j < l; j++) {

if (res[r][j] == 1) {

return false;

}

}

return true;

}

}

}else if (state[r][0]==1&&state[r][2]==0) {

//只在第9周选

if (l >= 40) {

return false;

} else {

int lab = l - timepice \* 4;

//把行号转化为Vector编号

int a = 0;

for (int m = 0; m < r; m++) {

a = a + state[m][0] + state[m][2];

}

a = a \* 10 + timepice;

if (Vx.get(a).getStuNum() <= Vx.get(a).getDeskNum(lab)

&& (Vx.get(a).getHope().equals("√")||Vx.get(a).getHope().equals(""))) {

//实验室够

//判断是否和前面冲突

for (int i = 0; i < r; i++) {

if (res[i][l] == 1) {

return false;

}

}

//搜索时间片

for (int j = timepice \* 4; j < l; j++) {

if (res[r][j] == 1) {

return false;

}

}

return true;

}

}

}

//任何一个条件不满足，该时间片的标志位都为false

return false;

}

//n为定值，在后续程序中不会改变

public int[][] NQueens1(int r, int n, Vector<App\_Super> Vx, int[][] res, int first,int [][]state,int tea\_num) {

//遍历列

int course = 0;

int timepice = 0;

for (int l = first; l < n; l++) {

//计算所属的时间片,损失精度

timepice = l / 4;

//判断是否可行

if (Place(r, l, Vx, timepice, res,state)) {

//该时间片该实验室可行

res[r][l] = 1;

course++;

//没有排完遍历下一个列

if (course < 4) {

if (l != n-1) {

NQueens1(r, n, Vx, res, l + 1,state,tea\_num);

}

} else {

//最后一组排课完成,输出一组可行解

if (r == tea\_num-1) {

flag=false;

return res;

} else {

if(flag)

NQueens1(r + 1, n, Vx, res, 0,state,tea\_num);

}

}

}

}

return res;

}

public NQueens(JLabel jl) {

jl2=jl;

}

//将数据库内容加载到集合，提高调用的效率

public BaseInfo Init(EmpModel em2,EmpModel em\_lab) {

//连接数据库取出数据

……

}

//将排课调用程序放在线程函数中，避免占用swing的线程

@Override

public void run() {

……

BaseInfo baseInfo=Init(em2,em3);

……

//从第0行开始，总共80列，结果存在res中

res = NQueens1(0, 20\*lab\_num, Vx, res, 0,state,tea\_num);

//将结果存到数据库中

……

}

}

2 SqlHelper.java为了方便使用数据库的连接和增删查改，封装了连接数据库的调用方法

package com.db;

import java.sql.\*;

public class SqlHelper {

//定义需要的对象

Connection ct=null;

PreparedStatement ps=null;

ResultSet rs=null;

String driver="com.mysql.jdbc.Driver";

String url="jdbc:mysql://localhost:3306/time\_table?characterEncoding=utf8&useSSL=false";

String user="root";

String passwd="123123";

int sum=0;

//构造函数，初始化ct

public SqlHelper()

{

……

}

//[]paras，通过?赋值方式可以防止漏洞注入方式，保证安全性

public ResultSet query(String sql,String []paras)

{

……

}

//刷新数据库

public boolean updateExecete(String sql,String []params)

{

……

}

……

}

3 ExcelOperationUtil.java 批量导入excel的函数

package com.excel;

import ……

public class ExcelOperationUtil {

public List<GetApp> readExcelData(String filePath) {

List<GetApp> list = new ArrayList<GetApp>();

try {

String path = filePath;

File xlsFile = new File(path);

FileInputStream fs = new FileInputStream(xlsFile);

Workbook book = Workbook.getWorkbook(fs);//获取工作簿对象

Sheet sheet = book.getSheet(0);//获取工作表对象,第一个sheet

int rows = sheet.getRows();//获取工作表中的数据行数

for (int i = 1; i <= rows - 1; i++) {//循环Excel工作表的行，并读取单元格数据

……

}

return list;

} catch (Exception e) {

……

}

}

}

**五、测试数据及其结果分析**

登录窗口会将输入的账户和密码与数据库内容比对，确认身份后才可以进入系统，当登入的用户身份不同时，功能权限也会不同，这样做可以确保安全性，避免修改其他用户的密码。下面展示了登录窗口和申请表（图7左）和课程安排（图7右）数据库格式。



图6 登录验证窗口

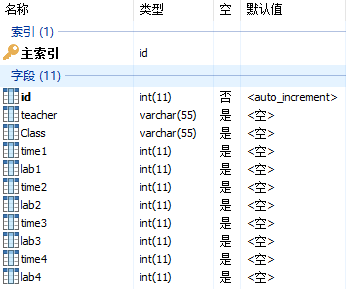
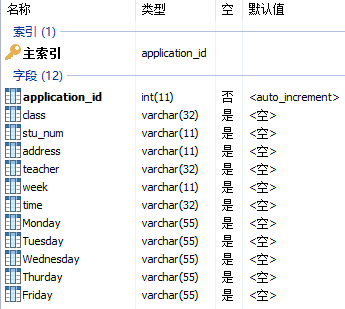


图7 申请表（左）和课程安排（右）数据库格式

人事管理界面可以增加用户，只有管理员才可以删除和修改修改用户，若想修改个人密码可以从菜单栏系统管理中进行修改。分权操作可以保证系统的安全性。

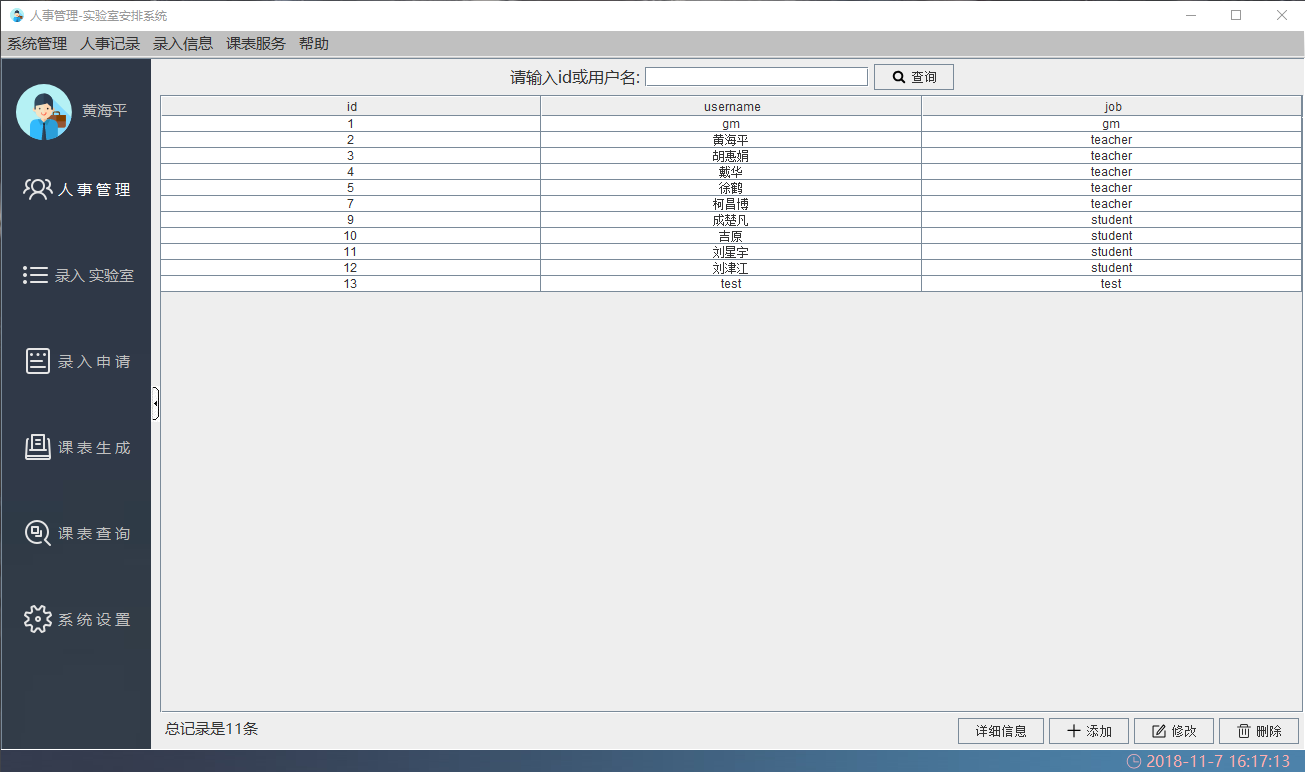


图8 人事管理窗口

录入实验室界面可以录入实验室的地址和座位数，生成课表时会将所有记录在数据库中的实验室进行安排，右下角支持增删改，顶部搜索框支持按提示关键字查询。

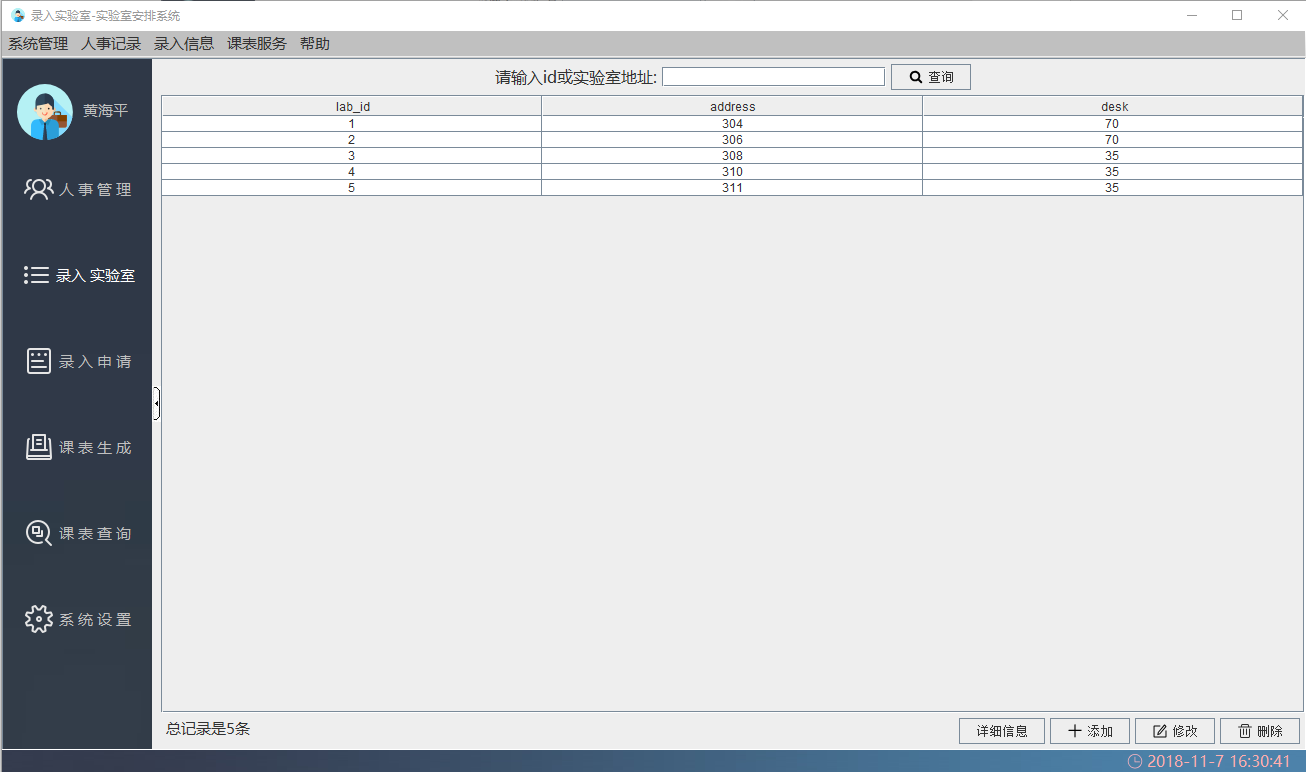


图9 实验室录入界面

录入申请界面可以录入老师的课程申请，可以根据老师或者班级进行查询。支持增删改查，与上述面板不同的是，该面板中支持批量导入。不仅可以从窗口中选取想要的excel文件，还可以将文件拖拽到窗口中实现批量导入。

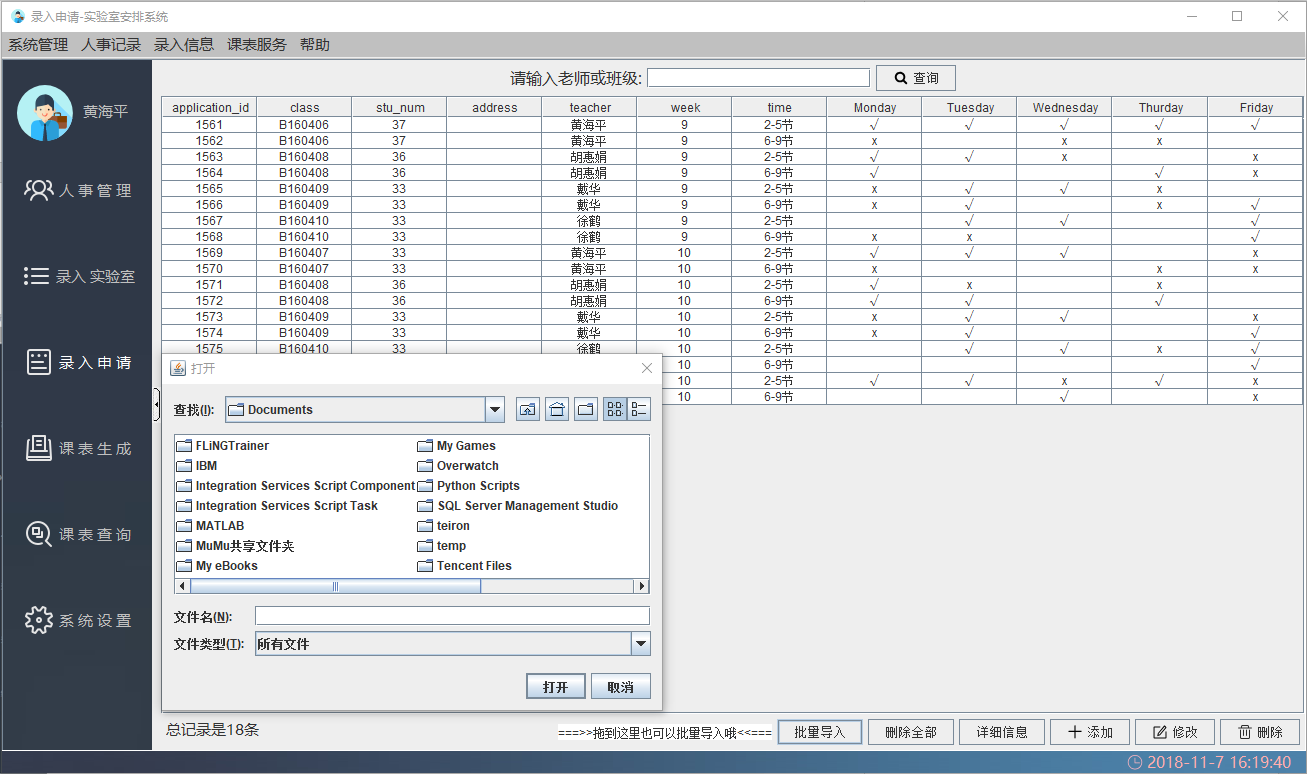


图10 申请录入界面以及批量导入窗口

课表生成界面是等待课表生成的界面，需要在此处新建一个线程执行排课算法程序避免占用swing组件监听线程。排课完成后提示用户查看课表。

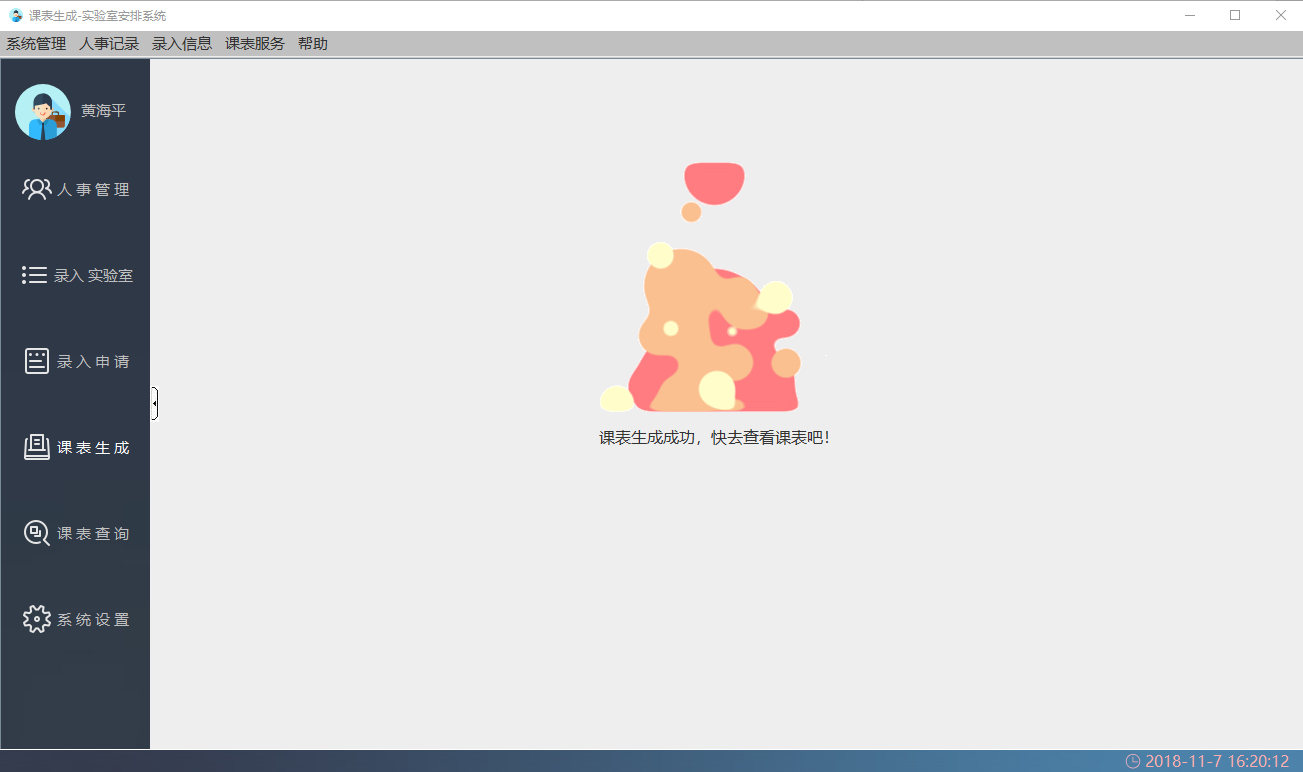


图11 课表生成及等待界面

课表生成完后可进行课表查询，顶部查询框支持按实验室地址或班级或老师进行查询，查看全表可以生成一张完整的课程表。



图12 课表查询界面

上述课表是按照实验要求中的默认数据所生成的课表，生成课表所用时间约为2s。当我在默认实验室表中添加实验室311并且在录入申请表内增加测试数据AAA后得到一张新的课表，结果和预期相符合，详细结果如图13所示。



图13 修改测试数据后的课表查询界面

通过分析可以得出结论：随着排课申请数量的增多和实验室数量的增加，系统的排课时间也将相应加长。系统在排课前先进行初始化，录入实验室信息、教师信息、班级信息和教室期望表，这些信息均可进行修改，信息无误后可进行排课。但是由于实验室排课案例规模较小，系统的排课时间能控制在2s以内。运用改进后的回溯法提高了排课的效率，本排课系统基本能够满足使用者的实际要求。

**六、课题完成过程中遇到的问题及解决方法**

问题1：excel批量导入时当路径上存在空格会找不到文件。

解决方法：使用repaceAll("%20",' ')替换后,能解决空格问题。原因是URL对空格，特殊字符(%,#,[]等)和中文进行了编码处理，空格变为%20。

问题2：频繁连接数据库输入SQL语言十分繁琐。

解决方法：将对数据库的所有操作都封装到SqlHelper.java，每次调用query(String sql,String []paras)传入相应的SQL语句和数值即可完成想要的操作。

问题3：JTable默认使用JLabel作为单元格，显示课表时无法展现多行信息。

解决方法：用TableCellTextAreaRenderer实现TableCellRenderer接口可以让单元格展现出JTextArea风格。

问题4：添加记录时，有单元格为空时会无法录入。

解决方法：将数据库中可能为空的单元格都改为varchar类型，原因是进行空值判断时，int类型的记录永远不会为空。

问题5：当修改数据库后JTable没有立即更新。

解决方法：每当处理一次数据库应该立即调用jtable.setModel(em)保持表格和数据库的同步。

问题6：当数据量增大后运行排课算法时，swing组件会失去响应，界面卡死。

解决方法：经过查阅资料,发现原来swing是线程不安全的,是单线程设计。swing有初始线程、事件派发线程、工作线程。UI展示完毕之后初始线程就结束了，控制就交给了事件派发线程,这之后所有组件的事件行为都由事件派发线程处理,事件派发线程是调用paint和update等回调方法的线程，它还是事件监听器接口中定义的事件处理方法。程序由事件派发线程执行一个任务，这时就要等待这个线程处理完这个任务，所以界面出现卡死现象,等这个任务结束了，事件派发线程就可以更新组件了。所以，运行长时间的排课算法应当新建一个线程单独运行。

**七、总结**

实验室安排系统主要分为两个方面：

（1）工程技术方面：需要熟练掌握数据库的增删查改操作，并且要在程序中将数据库与swing界面表格同步展示。同时也学会了批量导入excel至数据库的方法。通过这次训练，各方面的能力都得到了很大的提高。

（2）算法设计方面：尝试将算法运用到实际排课问题当中，建立了合适的模型进行存储和比对从而提高算法的效率。深刻领会了回溯法以深度优先的方式搜索解空间，并在搜索过程中用调用剪枝函数避免无效搜索的战略思想。

回溯算法的主要优点是对空间的消耗较少，并且对所求最优解在解答树中层次较深的问题时会获得比较理想的效果；在时间分配冲突时，向上回溯搜索到发生冲突的前一个记录，对其进行重新选择以解决问题。系统应用回溯法快速地实现了排课，系统根据申请的先后顺序赋予选择优先权。为得到满意的排课结果，在排课前进行预处理，当生成基本的排课结果后，可以根据用户需要进行关键字查询和展示完整课表。

总之，通过本次的算法与数据结构课程设计，理论与实际得到一次很好的结合，深入理解了回溯法解决一般问题的步骤。感觉自己在计算机自动排课方面的理论和实践的研究上都有很大的提高，使自己具备了独立解决计算机排课问题的能力，也得到了比较令人满意的排课结果。所设计的排课系统实现了排课要求的的各种基本功能和提高功能，包括各种排课初始数据的录入和编辑功能、课表的自动生成功能、课表的查询功能、课表展示功能以及excel批量导入的功能，使我平时学到的理论知识得到了很好的升华，为我今后的算法设计能力打下了坚实的基础，使我受益匪浅。