|  |
| --- |
| 《算法分析与设计》  **课 程 设 计 报 告**  学院（系）： 软件工程系  班 级： 116030804  学生姓名： 李鑫瑜 学号 11603080406  指导教师： 董世都    **时间： 从2018年12月17日 到 2018年 12月27日** |

目录

[题目一 二项式公式计算 3](#_Toc533879332)

[1. 问题描述 3](#_Toc533879333)

[2. 解决问题所用的算法设计方法及基本思路 3](#_Toc533879334)

[3. 采用的数据结构描述 3](#_Toc533879335)

[4. 算法描述 3](#_Toc533879336)

[5. 算法的时间空间复杂度分析 4](#_Toc533879337)

[6. 算法实例 4](#_Toc533879338)

[题目一 绘制分形树 5](#_Toc533879339)

[1. 问题描述 5](#_Toc533879340)

[2. 解决问题所用的算法设计方法及基本思路 5](#_Toc533879341)

[3. 采用的数据结构描述 5](#_Toc533879342)

[4. 算法描述 5](#_Toc533879343)

[5. 算法的时间空间复杂度分析 6](#_Toc533879344)

[6. 算法实例 6](#_Toc533879345)

[题目一 强联通分量 6](#_Toc533879346)

[1. 问题描述 6](#_Toc533879347)

[2. 解决问题所用的算法设计方法及基本思路 6](#_Toc533879348)

[3. 采用的数据结构描述 7](#_Toc533879349)

[4. 算法描述 7](#_Toc533879350)

[5. 算法的时间空间复杂度分析 7](#_Toc533879351)

[6. 算法实例 7](#_Toc533879352)

[题目一 二十四点 8](#_Toc533879353)

[1. 问题描述 8](#_Toc533879354)

[2. 解决问题所用的算法设计方法及基本思路 8](#_Toc533879355)

[3. 采用的数据结构描述 8](#_Toc533879356)

[4. 算法描述 8](#_Toc533879357)

[5. 算法的时间空间复杂度分析 9](#_Toc533879358)

[6. 算法实例 9](#_Toc533879359)

[题目一 最佳购买方案 9](#_Toc533879360)

[1. 问题描述 9](#_Toc533879361)

[2. 解决问题所用的算法设计方法及基本思路 9](#_Toc533879362)

[3. 采用的数据结构描述 10](#_Toc533879363)

[4. 算法描述 10](#_Toc533879364)

[5. 算法的时间空间复杂度分析 10](#_Toc533879365)

[6. 算法实例 10](#_Toc533879366)

[题目一 最少游艇租金问题 11](#_Toc533879367)

[1. 问题描述 11](#_Toc533879368)

[2. 解决问题所用的算法设计方法及基本思路 11](#_Toc533879369)

[3. 采用的数据结构描述 11](#_Toc533879370)

[4. 算法描述 11](#_Toc533879371)

[5. 算法的时间空间复杂度分析 11](#_Toc533879372)

[6. 算法实例 11](#_Toc533879373)

[题目一 数独问题 12](#_Toc533879374)

[1. 问题描述 12](#_Toc533879375)

[2. 解决问题所用的算法设计方法及基本思路 12](#_Toc533879376)

[3. 采用的数据结构描述 13](#_Toc533879377)

[4. 算法描述 13](#_Toc533879378)

[5. 算法的时间空间复杂度分析 13](#_Toc533879379)

[6. 算法实例 13](#_Toc533879380)

[题目一 地图着色问题 14](#_Toc533879381)

[1. 问题描述 14](#_Toc533879382)

[2. 解决问题所用的算法设计方法及基本思路 14](#_Toc533879383)

[3. 采用的数据结构描述 14](#_Toc533879384)

[4. 算法描述 14](#_Toc533879385)

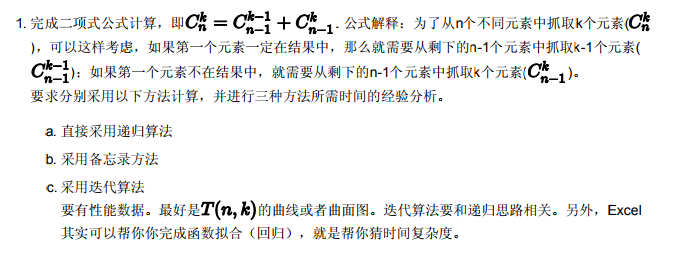
[5. 算法的时间空间复杂度分析 15](#_Toc533879386)

[6. 算法实例 15](#_Toc533879387)

[课程设计总结 15](#_Toc533879388)

# 题目一 二项式公式计算

## 问题描述



## 2. 解决问题所用的算法设计方法及基本思路

如动态规划,写出递归解结构,

回溯法/分支限界:解空间,状态空间树,如何剪枝

分治法:如何划分,如何合并,

贪心:贪心准则?如何能保证最有解

## 3. 采用的数据结构描述

## 4. 算法描述

算法名称

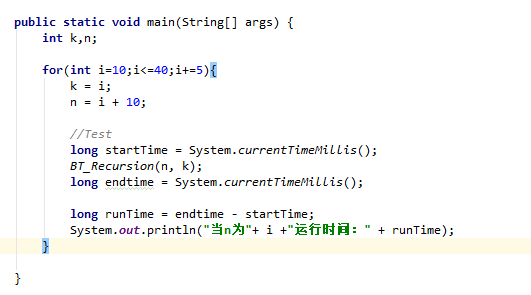
输入：

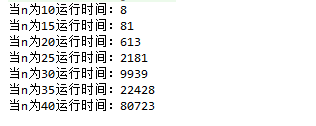
输出：

算法实现细节（可以用流程图，伪代码）

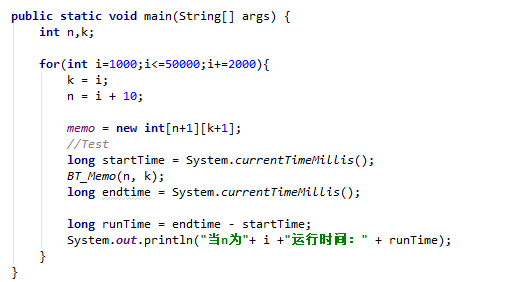
## 5. 算法的时间空间复杂度分析

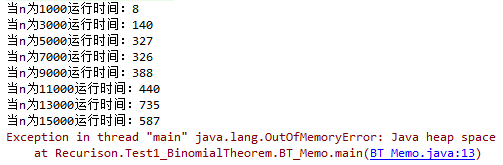
## 6. 算法实例

递归法

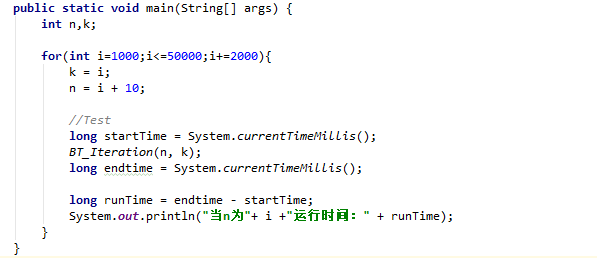


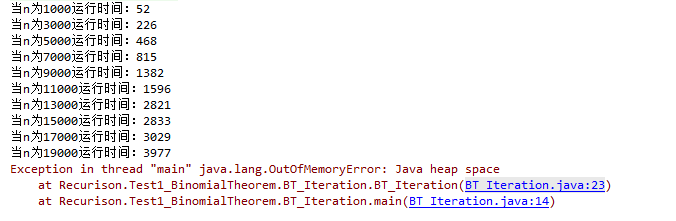
备忘录法





迭代





# 题目二 绘制分形树

## 问题描述



## 2. 解决问题所用的算法设计方法及基本思路

如动态规划,写出递归解结构,

回溯法/分支限界:解空间,状态空间树,如何剪枝

分治法:如何划分,如何合并,

贪心:贪心准则?如何能保证最有解

## 3. 采用的数据结构描述

## 4. 算法描述

算法名称

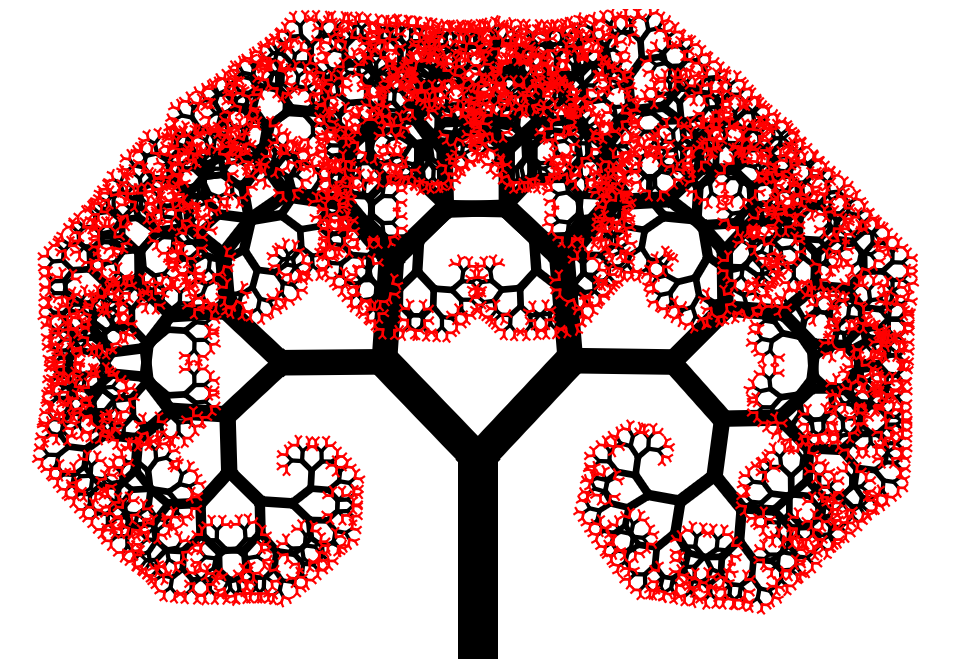
输入：

输出：

算法实现细节（可以用流程图，伪代码）

## 5. 算法的时间空间复杂度分析

## 6. 算法实例



# 题目三 强联通分量

## 问题描述



## 2. 解决问题所用的算法设计方法及基本思路

如动态规划,写出递归解结构,

回溯法/分支限界:解空间,状态空间树,如何剪枝

分治法:如何划分,如何合并,

贪心:贪心准则?如何能保证最有解

## 3. 采用的数据结构描述

## 4. 算法描述

算法名称

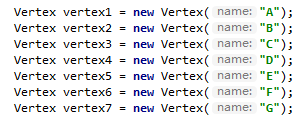
输入：

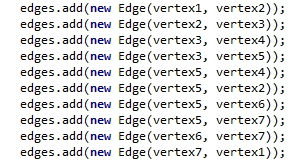
输出：

算法实现细节（可以用流程图，伪代码）

## 5. 算法的时间空间复杂度分析

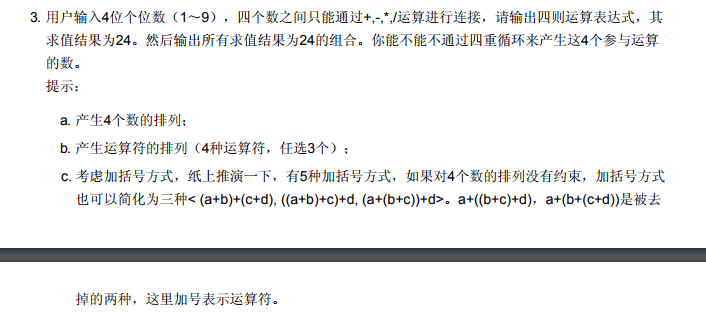
## 6. 算法实例





# 题目四 二十四点

## 问题描述



## 2. 解决问题所用的算法设计方法及基本思路

如动态规划,写出递归解结构,

回溯法/分支限界:解空间,状态空间树,如何剪枝

分治法:如何划分,如何合并,

贪心:贪心准则?如何能保证最有解

## 3. 采用的数据结构描述

## 4. 算法描述

算法名称

输入：

输出：

算法实现细节（可以用流程图，伪代码）

## 5. 算法的时间空间复杂度分析

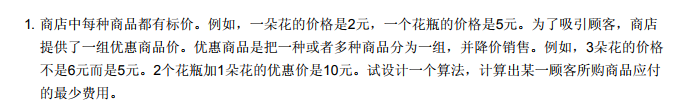
## 6. 算法实例





# 题目五 最佳购买方案

## 问题描述



## 2. 解决问题所用的算法设计方法及基本思路

如动态规划,写出递归解结构,

回溯法/分支限界:解空间,状态空间树,如何剪枝

分治法:如何划分,如何合并,

贪心:贪心准则?如何能保证最有解

## 3. 采用的数据结构描述

## 4. 算法描述

算法名称

输入：

输出：

算法实现细节（可以用流程图，伪代码）

## 5. 算法的时间空间复杂度分析

## 6. 算法实例







花瓶：5元 花：2元 馒头：3元

买6个花瓶，6朵花，6个馒头

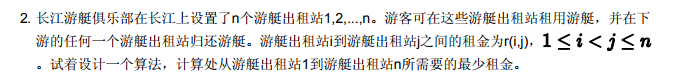
A：3个花瓶，1个馒头 8元

B：2个花瓶，1朵花 10元

|  |  |  |
| --- | --- | --- |
| 优惠方案 | 优惠方案列表 | 金额 |
| 0 | A B C D E  0.444 0.833 1 1 1 | 0 |
| A | A B C D E  0.444 0.833 1 1 1 | 0 + 8\*2 = 16 |
| B | A B C D E  0.444 0.833 1 1 1 | 16 |
| C | A B C D E  0.444 0.833 1 1 1 | 16 |
| D | A B C D E  0.444 0.833 1 1 1 | 16+2\*6=28 |
| E | A B C D E  0.444 0.833 1 1 1 | 28+3\*4=40 |

# 题目六 最少游艇租金问题

## 问题描述



## 2. 解决问题所用的算法设计方法及基本思路

如动态规划,写出递归解结构,

回溯法/分支限界:解空间,状态空间树,如何剪枝

分治法:如何划分,如何合并,

贪心:贪心准则?如何能保证最有解

## 3. 采用的数据结构描述

## 4. 算法描述

算法名称

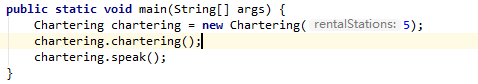
输入：

输出：

算法实现细节（可以用流程图，伪代码）

## 5. 算法的时间空间复杂度分析

## 6. 算法实例





|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | a | b | c | d | E |
| a | 0 | 2 | 3 | 6 | 10 |
| b | 2 | 0 | 4 | 1 | 6 |
| c | 3 | 4 | 0 | 8 | 2 |
| d | 6 | 1 | 8 | 0 | 4 |
| e | 10 | 6 | 2 | 4 | 0 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | a | b | c | d | E |
| a | 0 | 2 | 3 | 6 | 10 |
| b | 2 | 0 | 4 | 1 | 6 |
| c | 3 | 4 | 0 | 8 | 2 |
| d | 6 | 1 | 8 | 0 | 4 |
| e | 10 | 6 | 2 | 4 | 0 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | a | b | c | d | E |
| a | 0 | 2 | 3 | 3 | 8 |
| b | 2 | 0 | 4 | 1 | 6 |
| c | 3 | 4 | 0 | 8 | 2 |
| d | 3 | 1 | 8 | 0 | 4 |
| e | 8 | 6 | 2 | 4 | 0 |

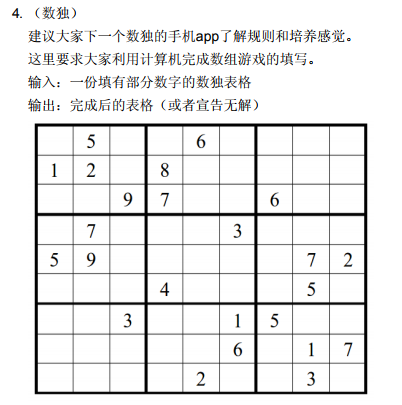
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | a | b | c | d | E |
| a | 0 | 2 | 3 | 3 | 5 |
| b | 2 | 0 | 4 | 1 | 6 |
| c | 3 | 4 | 0 | 8 | 2 |
| d | 3 | 1 | 8 | 0 | 4 |
| e | 5 | 6 | 2 | 4 | 0 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | a | b | c | d | E |
| a | 0 | 2 | 3 | 3 | 5 |
| b | 2 | 0 | 4 | 1 | 5 |
| c | 3 | 4 | 0 | 8 | 2 |
| d | 3 | 1 | 8 | 0 | 4 |
| e | 5 | 5 | 2 | 4 | 0 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | a | b | c | d | E |
| a | 0 | 2 | 3 | 3 | 5 |
| b | 2 | 0 | 4 | 1 | 5 |
| c | 3 | 4 | 0 | 6 | 2 |
| d | 3 | 1 | 6 | 0 | 4 |
| e | 5 | 5 | 2 | 4 | 0 |

# 题目七 数独问题

## 问题描述



## 2. 解决问题所用的算法设计方法及基本思路

如动态规划,写出递归解结构,

回溯法/分支限界:解空间,状态空间树,如何剪枝

分治法:如何划分,如何合并,

贪心:贪心准则?如何能保证最有解

## 3. 采用的数据结构描述

## 4. 算法描述

算法名称

输入：

输出：

算法实现细节（可以用流程图，伪代码）

## 5. 算法的时间空间复杂度分析

## 6. 算法实例

给定具体实例:

动态规划:填表

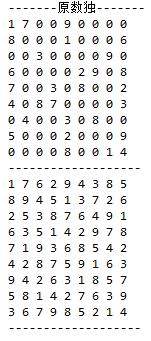
回溯法/分支限界:画状态空间树

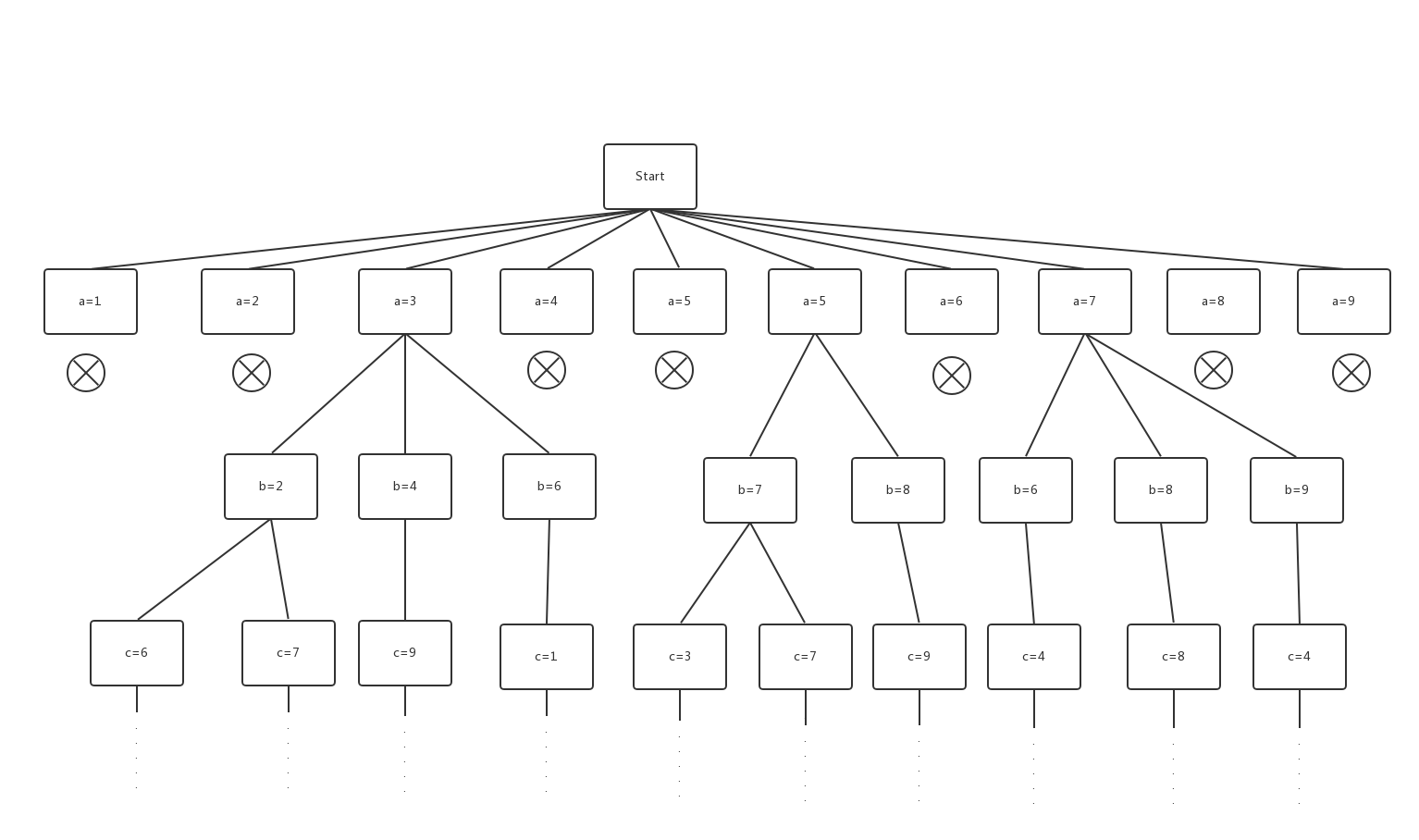
分治法: 划分及合并的步骤

贪心: 执行的步骤(参照书上Prim算法)

数据的输入（可以有屏幕截图）

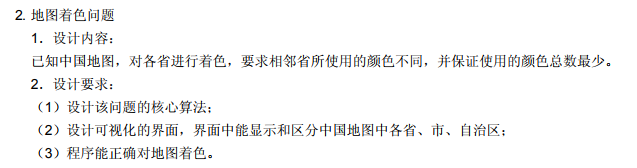
数据的输出（可以有屏幕截图）





# 题目八 地图着色问题

## 问题描述



## 2. 解决问题所用的算法设计方法及基本思路

如动态规划,写出递归解结构,

回溯法/分支限界:解空间,状态空间树,如何剪枝

分治法:如何划分,如何合并,

贪心:贪心准则?如何能保证最有解

## 3. 采用的数据结构描述

## 4. 算法描述

算法名称

输入：

输出：

算法实现细节（可以用流程图，伪代码）

## 5. 算法的时间空间复杂度分析

## 6. 算法实例

给定具体实例:

动态规划:填表

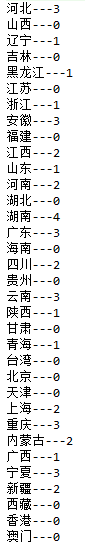
回溯法/分支限界:画状态空间树

分治法: 划分及合并的步骤

贪心: 执行的步骤(参照书上Prim算法)

数据的输入（可以有屏幕截图）

数据的输出（可以有屏幕截图）



# 课程设计总结