附录 课程实验

《程序设计方法与艺术》是为培养 ICPC 竞赛选手而开的一门综合基础课, 主要开课对象是学有余力的学生通过本课程的学习, 学生应该在课堂学习和上机实践的基础上, 进一步掌握和巩固使用程序设计知识, 增强数学建模能力。最终学会融合多项技术进行 ICPC 竞赛程序开发, 提高学生的创新能力和团队协作能力。

ICPC 竞赛涉及的内容多、知识面宽, 主要有离散数学、组合数学、计算几何、 数论初步等数学知识, 还有数据结构、 算法设计、 动态规划、 网络流等计算机专业知识。 本课程通过有选择地搜索算法、 计算几何初步、 动态规划以及相关专题已达到以点盖面, 引导学生去学习更多的知识, 达到自主学习的目的。从而提高学生算法设计与分析的素质和能力。

鉴于本课程的特点以及大多数学生在学习过程中所出现的问题, 本实验指导中不是简单地安排学生编写一些算法上机通过, 而是设置了兼顾不同能力层次、 具有多种形式及规模的实验任务。 本实践环节设置了7次实验, 每次实验按排四至六小时, 前6次实验分别侧重于教科书中的某一方面, 最后一次实验侧重于课程知识的综合应用, 要求学生对给定的问题能利用所学知识进行分析、 设计、 调试, 并进行总结。 具体实验安排可根据实验机时作适当的调整及增、 删, 实验内容也可根据学员的基础作适当选择。

# 实验一 sTL 的熟悉与使用

## 实验目的

* 1. ) 掌握 C++中 STL 的容器类的使用。
  2. ) 掌握 C++中 STL 的算法类的使用。

## 试验设备

硬件环境: PC 计算机软件环境:

操作系统: Windows 2000 / Windows XP / Linux

语言环境: Dev cpp / gnu c++

## 试验内容

* 1. 练习 vector 和 list 的使用。

定义一个空的 vector, 元素类型为 int, 生成 10 个随机数插入到 vector 中, 用迭代器遍历vector 并输出其中的元素值。在 vector 头部插入一个随机数,用迭代器遍历 vector 并输出其中的元素值。用泛型算法 find 查找某个随机数, 如果找到便输出, 否则将此数插入 vector 尾部。用泛型算法 sort 将 vector 排序, 用迭代器遍历 vector 并输出其中的元素值。删除 vector 尾部的元素, 用迭代器遍历 vector 并输出其中的元素值。将 vector 清空。

定义一个 list, 并重复上述实验, 并注意观察结果。

* 1. 练习泛型算法的使用。

定义一个 vector, 元素类型为 int, 插入 10 个随机数, 使用 sort 按升序排序, 输出每个元素的值, 再按降叙排序, 输出每个元素的值。练习用 find 查找元素。用 min 和max 找出容器中的最小元素个最大元素, 并输出。

# 实验二 搜索算法的实现

## 实验目的

* 1. 掌握宽度优先搜索算法。
  2. 掌握深度优先搜索算法。

## 试验设备

硬件环境: PC 计算机软件环境:

操作系统: Windows 2000 / Windows XP / Linux

语言环境: Dev cpp / gnu c++

## 试验内容

* 1. 将书上的走迷宫代码上机运行并检验结果, 并注意体会搜索的思想。
  2. 八皇后问题:

在一个国际象棋棋盘上放八个皇后, 使得任何两个皇后之间不相互攻击, 求出所有的布棋方法。上机运行并检验结果。

思考: 将此题推广到 N 皇后的情况, 检验在 N 比较大的情况下, 比方说 N=16 的时候, 你的程序能否快速的求出结果, 如果不能, 思考有什么方法能够优化算法。

* 1. 骑士游历问题:

在国际棋盘上使一个骑士遍历所有的格子一遍且仅一遍,对于任意给定的顶点, 输出一条符合上述要求的路径。

* 1. 倒水问题:

给定 2 个没有刻度容器, 对于任意给定的容积, 求出如何只用两个瓶装出 L 升的水, 如果可以, 输出步骤, 如果不可以, 请输出 No Solution。

# 实验三 计算几何算法的实现

## 实验目的

* 1. 理解线段的性质、叉积和有向面积。
  2. 掌握寻找凸包的算法。
  3. 综合运用计算几何和搜索中的知识求解有关问题。

## 试验设备

硬件环境: PC 计算机软件环境:

操作系统: Windows 2000 / Windows XP / Linux

语言环境: Dev cpp / gnu c++

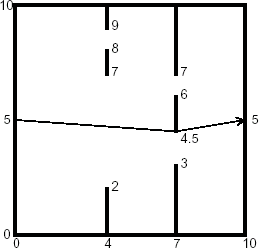
## 试验内容

* 1. 将讲义第三章第三节中的凸包代码上机运行并检验结果。
  2. 完成讲义第三章的课后习题, 上机运行并检验结果。
  3. 思考:

判线段相交时, 如果有个线段的端点在另一条线段上, 注意可能与另一条线段上的端点重合, 思考这样的情况怎么办。

* 1. 房间最短路问题:

给顶一个内含阻碍墙的房间, 求解出一条从起点到终点的最最短路径。房间的边界固定在 x=0,x=10,y=0 和 y=10。起点和重点固定在(0,5)和(10,5)。房间里还有 0 到 18 个墙, 每个墙有两个门。输入给定的墙的个数, 每个墙的 x 位置和两个门的 y 坐标区间, 输出最短路的长度。下图是个例子:



# 实验四 动态规划算法的实现

## 实验目的

* 1. 理解动态规划的基本思想、动态规划算法的基本步骤。
  2. 掌握动态规划算法实际步骤。

## 试验设备

硬件环境: PC 计算机软件环境:

操作系统: Windows 2000 / Windows XP / Linux

语言环境: Dev cpp / gnu c++

## 试验内容

1. 求两个字符串的最长公共子序列。

X 的一个子序列是相应于 X 下标序列{1, 2, … , m}的一个子序列, 求解两个序列的所有子序列中长度最大的, 例如输入: pear, peach 输出: pea。

1. 给定两个字符串 a 和 b, 现将串 a 通过变换变为串 b, 可用的操作为, 删除串 a 中的一个字符; 在串 a 的某个位置插入一个元素; 将串 a 中的某个字母换为另一个字母。对于任意的串 a 和串 b, 输出最少多少次能够将串变为串 b。

思考: 输出变换的步骤。

1. 输入一个矩阵, 计算所有的子矩阵中和的最大值。例如, 输入

0 -2 -7 0

9 2 -6 2

-4 1 -4 1

- 1 8 0 -2

输出为: 15

思考: 当矩阵很大时, 比如 100\*100 的矩阵, 你的程序还能够很快的得出结果吗, 如果不能, 请思考如何用动态规划的思想解决。

**实验五 模拟/密码类问题的建模与实现**

## 实验目的

* 1. 理解模拟类问题的建模与算法实现。
  2. 掌握密码类问题的建模与算法实现。

## 试验设备

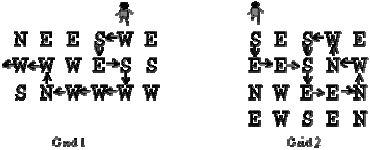
硬件环境: PC 计算机软件环境:

操作系统: Windows 2000 / Windows XP / Linux

语言环境: Dev cpp / gnu c++

## 试验内容

* 1. 模拟类问题
     1. 完成书上的 Kitty Fishing。
     2. 完成书上的 The Same Game 。
     3. 给定一个格子如下图所示, 机器人每次的移动方向如格子中的字母所示, 请根据所输入的起始位置, 求出机器人的运动情况。



* 1. 密码类问题
     1. 将书上的代码上机运行, 认真观察结果。
     2. 将书上的题目自己重做一遍。
     3. 给出一种加密算法如下: 输入原文, 将原文看做一个循环队列生成 n 个字符串, 对这 n 个串按首字母稳定排序, 取每个串的首字母做为密文。

例如: 输入 example, 得到 7 个字符串如下:

example, xamplee, ampleex, mpleexa, pleexam, leexamp,

eexampl,按尾字母稳定排序后得到:

ampleex, example, eexampl, leexamp, mpleexa, pleexam,

xamplee,取每个串的尾字母, 得到密文 xelpame, 现在要求输入一个原文对其加密, 或输入一个密文对其解密。

# 实验六 字符串/组合数学类问题的建模与实现

## 实验目的

* 1. 理解字符串处理问题的建模与算法实现。
  2. 掌握组合数学类问题的建模与算法实现。

## 试验设备

硬件环境: PC 计算机软件环境:

操作系统: Windows 2000 / Windows XP / Linux

语言环境: Dev cpp / gnu c++

## 试验内容

1. 字符串处理
   1. 将书上的代码上机运行。
   2. 求 100 个字符串的最长公共子序列。
   3. 自学 KMP 算法, 并上机练习。
2. 组合数学部分
3. 买票问题:

电影院卖电影票, 但是没有零钱找, 票价一张 5 元, 买票的人为 n 个持有 5 元, m 个持有 10 元, 求解出可能的买票序列的个数, 使得电影院能够将票卖完。例如, n=3,m=3 时, 输出 180 。 当 n=5,m=3 时, 输出 20160 。 当 n=100,m=100 时, 输出780849373628505449061745756368500061678352453155617092371032211133029158379

607270218123053477212498915026942711801622604215487911131323866397947153418

643496151943423040359720037026721726655886753912538851736666625608050720226

034508195568556839182082416159660797603533326956467231851806002328416691877

4048734879105185187102720000000000000000000000000000000000000000000000000。

1. 给定一个数字的集合{0, 1, ... , k}, 0 <= k <= 9, 定义一个 tight word iff 相临的两个数字之间的差最多是 1, 求解长度为 n 的tight word 在所有的 word 中的比率。比如 k=2,n=5 时, 输出 40.74074, k=8,n=7 时, 输出 0.10130。
2. 巧克力问题:

有袋子里均匀地装着 c 种颜色的巧克力, 每种巧克力均有无限多。每次从袋子里拿一块放在桌子上, 如果桌子上已经有一块颜色相同的巧克力, 就把两块巧克力都吃掉。一共取出了 n 块巧克力, 问: 最后桌子上有 m 块的概率为多大。例如 c=5,n=100,m=2 时, 概率为 0.625。

# 实验七 ICPC 设计实验

## 实验目的

综合运用所学知识, 对给定问题建模并求解。

## 试验设备

硬件环境: PC 计算机软件环境:

操作系统: Windows 2000 / Windows XP / Linux

语言环境: Dev cpp / gnu c++

## 试验内容

1. 字符串调整

考虑两个字符串 X = x1x2 ... xm 和 Y = y1y2 ... yn, 且字符属于集合{A, G,C, T`。现在要在 X 和Y 中插入空格, 得到字符串 X\*和 Y\*, 要求:

* 1. X\*和Y\*有同样的长度;
  2. 如果忽略空格, X\*=X, Y\*=Y。

举个例子: X='GATCCGA',Y='GAAAGCAGA',插入空格后(空格用-表示), 可以得到的一种结果是

X\*=G-A--TCCGA Y\*=GAAAG-CAGA

当然也可以是下面的结果:

X\*=GA---TCCGA Y\*=GAAAG-CAGA.

怎么评定它们的优劣呢? 我们又下面的法则: 如果 xi 和 yj 对齐, 得分是



如果一个X\*(或Y\*)的连续的子串和Y\*(或X\*)的一个长度为k 的空格子串对齐,得分是 -(4

+ k).

所以, 第一种结果的得分是 2 - (4 + 1) + 2 - (4 + 2) - (4 + 1) + 2 - 1 + 2 + 2 = -7;

第二种结果的得分是 2 + 2 - (4 + 3) - (4 + 1) + 2 - 1 + 2 + 2 = -3。

现在你的问题是对于给定的 X 和Y 的, 找出得分最高的 X\*和 Y\*。对于上面的例子, 最好的结果是:

X\*=GA--TCCGA Y\*=GAAAGCAGA.

得分为 2+2-(4+2)-1+2-1+2+2=2。在这题中, m 和 n 最大为 500, 并且X\*和 Y\*中没有任何两个空格是对齐的。

## 输入格式:

第一行为一个整数, 给出了测试的数目, 接下来每两行给出一对 X 和Y。

## 输出格式:

对于每一组输入, 输出得分的最大值。

## 输入样例:

3

ACGGCTTAGATCCGAGAGTTAGTAGTCCTAAGCTTGCA AGCTTAGAAAGCAGACACTTGATCCTGACGGCTTGAA TTGAGTAGTGTTTTAGTCCTACACGACACATCAAATTCGGACAAGGCCTAGCT TTCAAGTCCTACAATGTGTGTCAAATTCGCTTGGCCGAAAGCC TTTGGGAACGTGTGTAGACTTCCCCATGCGATGG AACACACACGGACTTCATGCTGG

## 输出样例:

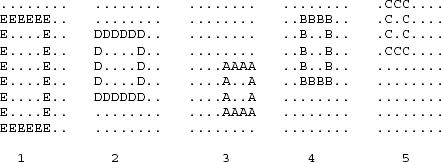
18

20

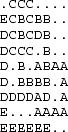
2

1. 框架覆盖

考虑下面的五个框架:



现在把它们安顺序一个个地叠起来, 上面的框架就会覆盖掉下面框架的一部分。假设我们就得到了下面的图:



它们是按照什么顺序从下往上搭建起来的呢? 答案是 EDABC。你的任务就是按照搭建后的图, 判断它们的顺序。为了便于处理, 有以下限制:

1. 框架的任何一边不小于 3;
2. 每个框架都没有任何一条边被完全覆盖, 框架的角同时属于两条边;
3. 框架用大写字母表示且互不相同。

## 输入格式:

每个测试的开头是两个整数 h 和w,表示了宽和高。接下来的 h\*w 个字符描述了这个图。**输出格式:**

按照从底向上输出框架搭建的顺序, 如果有多种可能, 按照字典序输出它们。

**输入样例:** 9

8

.CCC....

ECBCBB.. DCBCDB.. DCCC.B.. D.B.ABAA D.BBBB.A DDDDAD.A E AAAA

EEEEEE..

**输出样例:** EDABC

1. 猴子和香蕉

在一个动物园里, 科学家们设计了一个实验来测试猴子的 IQ: 把香蕉挂在很高的地方, 猴子可以利用一些箱子搭成一个塔, 每个箱子有三个参数(*xi*, *yi* 和 *zi*.), 表示箱子的三维。由于箱子可以翻转, 箱子 i 可以放在箱子 j 上面的充要条件是对于它们的接触面, 箱子 i 的长和宽都必须严格小于箱子 j 的长和宽。

现在你的任务是计算对于给定的箱子种类集合, 所能搭到的最大高度是多少, 每种箱子都有无限多个。

## 输入格式:

输入数据包含多个测试, 每个测试的第一行给出了箱子的种数 n(n<=30), n=0 时结束, 接下来的 n 行给出了每种箱子的三维。

## 输出格式:

对于每一个测试, 输出可以搭成塔的最大高度。**输入样例**:

1

10 20 30

2

6 8 10

5 5 5

7

1 1 1

2 2 2

3 3 3

4 4 4

5 5 5

6 6 6

7 7 7

5

31 41 59

26 53 58

97 93 23

84 62 64

33 83 27

0

## 输出样例:

Case 1: maximum height = 40

Case 2: maximum height = 21

Case 3: maximum height = 28

Case 4: maximum height = 342