《算法设计与分析课程设计》

任 务 书

**1.加法接力赛（基础算法：模拟法）**

【问题描述】

卡卡同学最近开始学习数的加法运算，一天老师为了测试大家对加法的掌握情况，让同学们做了一个有趣的加法接力赛游戏。老师在黑板上从左到右写下了n个大于零的整数，老师接下来找出n个同学完成下面的计算任务。第一个同学在黑板上抄下最左边的数字，第二个同学将第一个同学写下的数字加上老师写的第二个数字所得到的和写在黑板上。以此类推，第m个同学将前面第m-1个同学所写下的数字加上老师在黑板上写下的第m个数字所得之和，并将结果写到黑板上。当计算完成这n个数字之后，老师又让同学从最右边开始按相同的方法再次计算出n个数值。最后老师让同学们找出这所有数字当中共有多少个是相同的。

例如：老师在黑板上写下了7个数，从左到右依次为：3，6，2，1，4，5，2，则同学们第一次从左边开始计算所得到的7个数值应该是：3，9，11，12，16，21，23；而第二次从右边开始计算所得到的7个数值应该是：2，7，11，12，14，20，23；于是第一次与第二次计算出的数值中相同的有3个，它们分别是：11、12与23。

【输入】

共两行，第一行为n（1<n<100），第二行为n个数，用空格隔开， 每个数的大小在0～200之间。

【输出】

第一次与第二次计算出的数值中相同的个数。

【输入样例】

7

3 6 2 1 4 5 2

【输出样例】

3

**2.求根节点到叶子节点的数字之和（基础题）**

【问题描述】

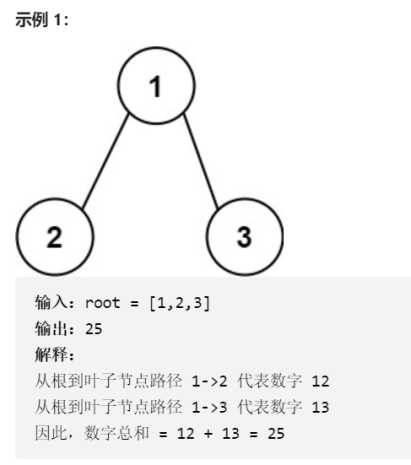
给你一个二叉树的根节点 root ，树中每个节点都存放有一个 0 到 9 之间的数字。

每条从根节点到叶节点的路径都代表一个数字：

例如，从根节点到叶节点的路径 1 -> 2 -> 3 表示数字 123 。

计算从根节点到叶节点生成的所有数字之和 。

叶节点是指没有子节点的节点。



**3.机器人的逻辑（基础算法）**

【问题描述】

2035年，智能机器人在各行各业中的应用已经十分普遍了，毕竟它做事时的精度与力量比一个普通人是强多了。

王涛的运输队里就有一个，是用来装卸货物的。

这天，他们的任务是要把n根废旧的条形钢材运送到钢铁厂重新冶炼。这些钢材长短不同（有些还特别的长），为了便于运输，只好把它们切割成小段。所以，他给机器人的任务是：把这些钢材切割并装上卡车。

等机器人做完这事的时候，王涛一看结果，自己都被逗笑了：机器人的逻辑就是和人不同啊——装在车上的所有小段的钢材，居然长度都是一样的（以米为单位），而且，还是所有可行方案中，切割次数最少的那种方案！

如果告诉你最开始那n根钢材的长度，你能算出机器人切割出的小段的长度吗？

【输入】

第一行为整数n，表示原始钢材的数量。

第二行中是n个用空格分开的整数，表示每根废旧钢材的长度（以米为单位），已知这些整数不小于1，不超过400000。

【输出】

只有一个整数，表示机器人切割出来的每个小段的长度。

【样例输入】

4

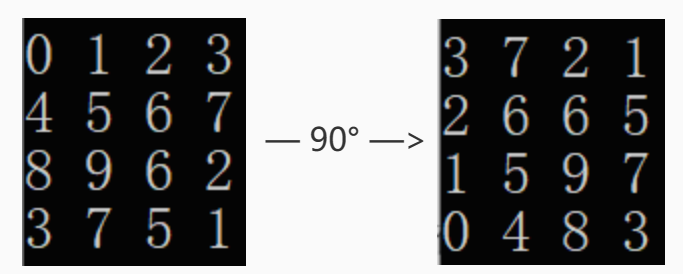
4 22 8 12

【样例输出】

2

**4.**给定一个 n∗n 数组矩阵，请编写一个程序将其按照逆时针方向进行旋转90°。注意题目要求：不可以借助任何其他矩阵变量空间，必须在矩阵本身完成翻转，可以使用常数级别的内存空间。时间复杂度为O（n）。（递归算法）

例子如下：



输入输出格式要求：

输入格式

第一行有一个整数 n，表示数组矩阵每一行数字的数量；

接下来有n行，每一行各有 n 个数字，共同组成一个 n∗n 的数字矩阵。

输出格式

输出也有n行，每一行各有 n 个数字，共同组成一个 n∗n 的数字矩阵，矩阵是翻转后的整数矩阵。

输入输出样例

输入

4

0 1 2 3

4 5 6 7

8 9 6 2

3 7 5 1

输出

3 7 2 1

2 6 6 5

1 5 9 7

0 4 8 3

**5.**乐羊羊饮料厂正在举办一次促销优惠活动。乐羊羊C型饮料，凭3个瓶盖可以再换一瓶C型饮料，并且可以一直循环下去，但不允许赊账。请你计算一下，如果小明不浪费瓶盖，尽量地参加活动，那么，对于他初始买入的n瓶饮料，最后他一共能得到多少瓶饮料。（递归算法）

输入格式

一个整数n，表示开始购买的饮料数量（0<n<10000）

输出格式

一个整数，表示实际得到的饮料数

输入输出样例

输入

100

输出

149

**6.大整数乘法(分治算法)**

【题目描述】

基于分治思想设计一个大整数乘法的实现程序，可以进行两个大整数的乘法运算，要求时间复杂度小于O(n2)。

【输入】

分行输入两个大整数，两个整数的长度可以不相同，如果输入的不是合法的整数，应提示重新输入。

【输出】

计算输出两个大整数的乘法结果。

【输入样例1】

Please input num1 and num2

Input num1:123456789

Input num2：987654321

【输出样例1】

result：121932631112635269

【输入样例2】

Please input num1 and num2

Input num1:99999999999999999999

Input num2：8888

【输出样例2】

result：888799999999999999991112

【输入样例3】

Please input num1 and num2

Input num1:123abc999

Error，try again：123999

Input num2：0000000

【输出样例2】

result：0

**7.第k小的元素（分治算法）**

【题目描述】

在包含n个元素的整数序列中，查找并输出第k小的元素，请基于分治思想设计一个输出第K小元素的程序，要求算法时间复杂度为O(n)。

【输入】

第1行输入整数n(0<n<=100)，表示有n个整数，第2行依次输入n个整数，第3行输入整数k(k<=n)。

【输出】

输出第k小的元素。

【输出样例1】

6

20 5 -6 9 0 7

3

【输出样例1】

5

【输出样例2】

14

20 18 16 14 12 10 8 6 4 2 0 -2 -4 -6

4

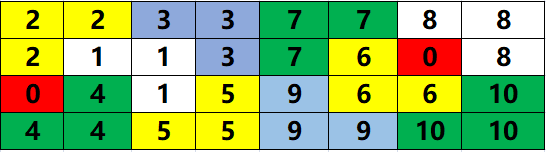
【输出样例2】

0

**8.二残缺棋盘覆盖问题（分治算法）**

【题目描述】

在一个2k \* 2k+1的棋盘中，可以在水平方向上分成两个2k \* 2k的子棋盘，下图所示为 22 \* 23的棋盘，在水平方向上可以分为两个22 \* 22的子棋盘，先假定水平方向上的两个子棋盘各有一块残缺，要求用3格板将其铺满。



【输入】

第1行输入整数k，第2行输入左边子棋盘残缺位置的坐标，第3行输入右边子棋盘残缺位置的坐标。

【输出】

以二维数组的形式输出铺满残缺棋盘的方案，0表示残缺，各3格板依次用数字表示。

【输入样例1】

2

3 1

2 7

【输出样例1】

2 2 3 3 7 7 8 8

2 1 1 3 7 6 0 8

0 4 1 5 9 6 6 10

4 4 5 5 9 9 10 10

【输入样例2】

1

1 1

2 4

【输出样例2】

0 1 2 2

1 1 2 0

**9.比赛日程安排（分治算法）**

【问题描述】

某班级选拔出8名同学参加围棋比赛，要求参加比赛的每个选手必须与其它7名选手各比赛一次，每个选手一天只能赛一次，比赛共进行 7天，要求每天没有选手轮空，若只有8名同学比赛，设计好的比赛安排方案如下：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 第1天 | 第2天 | 第3天 | 第4天 | 第5天 | 第6天 | 第7天 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 2 | 1 | 4 | 3 | 6 | 5 | 8 | 7 |
| 3 | 4 | 1 | 2 | 7 | 8 | 5 | 6 |
| 4 | 3 | 2 | 1 | 8 | 7 | 6 | 5 |
| 5 | 6 | 7 | 8 | 1 | 2 | 3 | 4 |
| 6 | 5 | 8 | 7 | 2 | 1 | 4 | 3 |
| 7 | 8 | 5 | 6 | 3 | 4 | 1 | 2 |
| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |

但随着比赛规模的扩大，如果有k(k=2x)位同学参加比赛,要求每名参赛选手与其他k-1名选手都比赛一次，每个选手一天只能赛一次，比赛共进行 k-1天，要求每天没有选手轮空，请设计算法完成该项工作。

【输入】第1行输入正整数x，表示有2x位同学参加比赛。

【输出】用二维数组的形式输出比赛安排方案。

【输入样例】

3

【输出样例】

1 2 3 4 5 6 7 8  
2 1 4 3 6 5 8 7  
3 4 1 2 7 8 5 6  
4 3 2 1 8 7 6 5  
5 6 7 8 1 2 3 4  
6 5 8 7 2 1 4 3  
7 8 5 6 3 4 1 2  
8 7 6 5 4 3 2 1

**10.选择问题（分治算法）**

对于给定的有n个元素的数组a[0:n-1],要求从中找出第k小的元素。

输入

输人有多组测试例。对每一个测试例都有2行，第一行是整数m和k(1≤k<π≤100)，第二行是n个整数。

输出

第k小的元素。

示例1：

输入：

|  |
| --- |
| 5 2  3 9 4 1 6 |

输出：

|  |
| --- |
| 3 |

示例2：

输入：

|  |
| --- |
| 7 3  4 59 7 23 61 55 46 |

输出：

|  |
| --- |
| 23 |

**11. (分治算法)**

给定一个大小为n的数组，找到其中的多数元素。多数元素是指在数组中出现次数大于⌊n/2⌋的元素。

你可以假设数组是非空的，并且给定的数组总是存在多数元素。

**示例1:**

**输入:**

|  |
| --- |
| 3 2 3 |

**输出:**

|  |
| --- |
| 3 |

**示例2:**

**输入:**

|  |
| --- |
| 2 2 1 1 1 2 2 |

**输出:**

|  |
| --- |
| 2 |

**12.银行卡测试（分治算法）**

一家银行没收了n张涉嫌欺诈的银行卡。每张卡对应一个唯一的账户，但一个账户可以绑定多张银行卡。因此，如果两张银行卡属于同一账户，则它们是等价的。银行有一台测试机器来检查两张卡是否等价。他们想知道在n张银行卡的集合中，是否有超过n/2张卡是等价的，即某个账户绑定了超过n/2张的银行卡。注意测试机器能做的唯一操作就是选择两张卡并测试它们的等价性。要求仅使用O(n log n) 次测试机器完成任务。

【输入】

第1行，n

第2行，银行卡号

第3~n+2行，银行卡等价矩阵

【输出】

有超过n/2张卡的账号，输出True；否则输出False。

【输入样例】

5

1 2 3 4 5

1 0 0 1 1

0 1 1 0 0

0 1 1 0 0

1 0 0 1 1

1 0 0 1 1

【输出样例】

True

【输入样例】

6

1 2 3 4 5 6

1 0 0 0 0 1

0 1 1 0 0 0

0 1 1 0 0 0

0 0 0 1 1 0

0 0 0 1 1 0

1 0 0 0 0 1

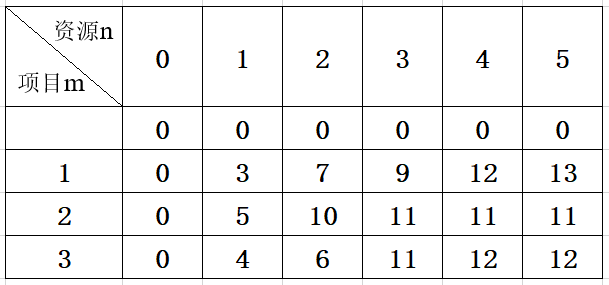
【输出样例】

False

**13.资源分配问题（动态规划算法）**

【问题描述】

某公司将资源n分配给m个项目，各项目得到相应的资源后，每年都有相应的收益。例如当n=5，m=3时，各项目的收益情况如下表所示。



请设计算法计算如何将资源n分配给m个项目，使总收益最大。

【输入】

第一行输入m和n，再依次输入m+1行，每行为n+1个数，位置(i,j)上的数表示给i项目分配资源j的收益，0≤i≤m,0≤j≤n。

【输出】

输出前m行每行两个数，分别表示项目编号及分配的资源数，最后一行表示获得的最大收益。

【输入样例】

3 5

0 0 0 0 0 0

0 3 7 9 12 13

0 5 10 11 11 11

0 4 6 11 12 12

【输出样例】

3 3

2 2

1 0

21

**14.加密字符串相似度计算（动态规划算法）**

【问题描述】

现有4个加密的字符串(中间不含空格)，需要分析研究它们的相似度，两个字符串的相似度用其最长公共子系列的长度表示，例如，字符串“ABDECFG”和字符串“ADCGEFA”的一个最长公共子序列为 “ADEF”，所以这两个字符串的相似度为4。现在需要对输入的4个字符串，分别计算它们的相似度，找出相似度最高的一对字符串，若存在多对相似度最高的字符串，应全部输出。

【输入】 分4行输入4个字符串。

【输出】 按行依次输出相似度最高的一对字符串、它们的相似度、对应的最长公共子序列，若有多组最高相似度相同，应全部输出。

【输入样例】

ABDECFEB

ADCEBGH

ADCFEB

BECEBFBD

【输出样例】

ABDECFEB ADCFEB 6 ADCFEB

输出样例说明：以上4个字符串，共可以构成6组，依次计算它们的相似度如果，因此，相似度最高的是字符串ABDECFEB和ADCFEB，相似度为6，对应的一个最长公共子序列为ADCFEB。

ABDECFEB ADCEBGH 5 ADCEB

ABDECFEB ADCFEB 6 ADCFEB

ABDECFEB BECEBFBD 5 BECFB

ADCEBGH ADCFEB 5 ADCEB

ADCEBGH BECEBFBD 3 CEB

ADCFEB BECEBFBD 3 CFB

**15. 数字串分解（动态规划）**

【题目描述】

最近班级组织了一场别开生面的趣味竞赛活动，老师出了这样一道题目：

设有一个长度为X的数字字符串，要求使用Y个乘号将它分成Y+1个子数字串，你需要找出一种方法，使得这Y+1个部分的乘积最大。例如一个数字串：615， Y=1时会有以下两种分法：

(1）6\*15=90

(2）61\*5=305

符合要求的答案是：61\*5=305。

请设计一个程序计算符合要求的答案。

【输入】

第一行共有2个自然数X，Y（6≤X≤10，1≤Y≤6）

第二行是一个长度为X的数字字符串。

【输出】

输出所求得的最大乘积（一个自然数）。

【输入样例】

6 3

310143

【输出样例】

3720

**16.滑雪（动态规划）**

Michael喜欢滑雪。为了获得速度，滑的区域必须向下倾斜，而且当滑倒坡底时，不得不再次走上坡或者等待搭载升降机。

Michael想知道在一个区域中最长的滑坡。区域由一个二维数组给出额的每个数字代表点的高度。下面是一个例子：

1 2 3 4 5

16 17 18 19 6

15 24 25 20 7

14 23 22 21 8

13 12 11 10 9

一个人可以从某个点滑向上、下、左、右相邻的四个点之一，当且仅当高度减小。在上面的例子中，一条可滑行的滑坡为24 - 17 - 16 - 1。当然，25 - 24 - 23 - … - 3 - 2 - 1更长。事实上，这是最长的一条滑坡。

**输入数据：**

输人的第一行表示区域的行数R和列数C(1≤R，C<100)。下面是R行，每行有C个整数，代表高度h，0≤h≤10 000。

**输出要求：**

输出最长滑行区域的长度

**输入样例**

5 5

1 2 3 4 5

16 17 18 19 6

15 24 25 20 7

14 23 22 21 8

13 12 11 10 9

**输出样例**

25

**17.灌溉草场（动态规划）**

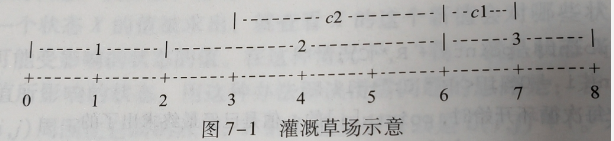
**问题描述：**

在一片草场上，有一条长度为L(1≤L≤1000000,L为偶数)的线段。John的N(I≤N≤1000)头奶牛都沿着草场上这条线段吃草，每头牛的活动范围是一个开区间(S，E)，S，E都是整数。不同奶牛的活动范围可以重叠。John要在这条线段上安装喷水头灌溉草场。每个喷水头的喷洒半径可以随意调节，调节范围是[A,8](1≤A≤B≤1000)，A,B都是整数。要求：

* 线段上的每个整点恰好位于一个喷水头的喷洒范围内；
* 每头奶牛的活动范围要位于一个喷水头的喷洒范围内；
* 任何喷水头的喷洒范围不可越过线段的两端(左端是0，右端是L)。

请问，John最少需要安装多少个喷水头?

灌溉草场示意如图7-1所示。



在位置2和6，喷水头的喷洒范围不算重叠。

**输入数据：**

第1行:整数N，L。

第2行:整数A，B。

第3~N+2行:每行两个整数S,E(0≤S<E≤L)，表示某头牛活动范围的起点和终点在线段上的坐标(即到线段起点的距离)。

**输出要求：**

最少需要安装的喷水头数量;若没有符合要求的喷水头安装方案，则输出-1。

**输入样例：**

1. 8
2. 2
3. 7
4. 6

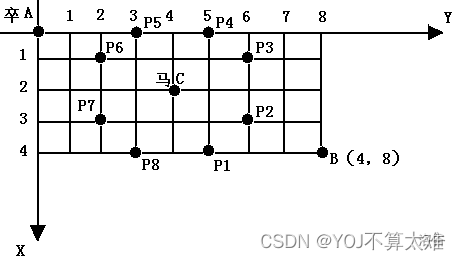
**输出样例：**

3

**18.马拦过河卒（动态规划）**

【问题描述】

棋盘上A点有一个过河卒，需要走到目标B点。卒行走的规则：可以向下、或者向右。同时在棋盘上的某一点有一个对方的马（如C点），该马所在的点和所有跳跃一步可达的点称为对方马的控制点，如图中的C点和P1，……，P8，卒不能通过对方马的控制点。棋盘用坐标表示，A点(0,0)、B点(n, m) (n,m为不超过20的整数),同样马的位置坐标是需要给出的，C≠A且C≠B。现在要求你计算出卒从A点能够到达B点的路径的条数。



【输入】

给出n、m和C点的坐标。

【输出】

从A点能够到达B点的路径的条数。

【输入样例】

8 6 0 4

【输出样例】

1617

**19. 子矩阵的大小（动态规划）**

【问题描述】

设矩阵的大小为矩阵中所有元素的和，现输入一个N\*N的矩阵，请设计算法计算最大的非空(大小至少是1\*1)子矩阵的大小。

例如4×4的矩阵为：

1 0 1 1

1 1 1 1

1 1 1 1

0 1 -6 -8

其最大子矩阵为：

1 0 1 1

1 1 1 1

1 1 1 1

子矩阵大小是11。

【输入】

输入是一个N×N的矩阵。输入的第一行给出N(0<N≤100)。再后面的若干行依次输入矩阵各行的元素，用空格作为分隔符，设矩阵中整数的范围都在[−127,127]。

【输出】

输出最大子矩阵的大小。

【输入样例】

4

1 0 1 1

1 1 1 1

1 1 1 1

0 1 -6 -8

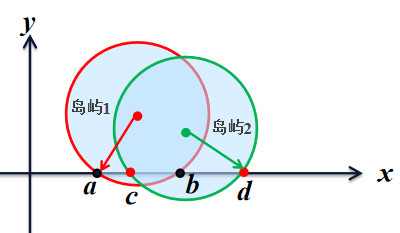
【输出样例】

11

**20.最优雷达设置问题（贪心算法）**

【问题描述】

为保卫国家的主权和领土安全，海岸线上要安装一些雷达监控岛屿，假设海岸线可视为一条直线，一边为大海，一边为大陆，岛屿可看作一个点，雷达的监控范围为一个半径为d的圆。为方便表示，使用笛卡尔坐标系，令海岸线为X轴，X轴上方为大海，X轴下方为大陆，所有岛屿的位置都由坐标给出。

可将岛屿坐标作为圆心，以半径d画圆,如果圆与X轴没有交点，就无法覆盖岛屿；如果圆与X轴的交点为a,b,雷达放到a、b之间那就是可以将该岛屿覆盖住。给定每个岛屿的位置，以及雷达的监控半径，请设计算法，编写程序找出覆盖所有岛屿的雷达安装的最小数量。

【输入】有多组测试数据，第1行给出岛屿的数目n(1<=n<=1000)和雷达监控范围d，接下来n行输入第i个岛屿的X坐标和Y坐标，当n和d都是零时结束。

【输出】每组测试数据输出一行，要包括样例的编号和最小雷达安装数，如果无法覆盖所有岛屿，输出“-1”。

【输入样例】

2 5

-3 4

-6 3

4 5

-5 3

-3 5

2 3

3 3

0 0

【输入样例】

Case 1: 1

Case 2: 2

**21.** **采购农产品（贪心算法）**

【问题描述】

某公司从一些农户手中采购农产品，每位农户能提供的产品数量是一定的。该公司每天可以从农户手中采购到小于或者等于农户最大产量的整数数量的农产品。

给出公司每天对农产品的需求量，还有每位农户提供的产品单价和产量。计算采购足够数量的农产品所需的最小花费。

注：每天所有农户的总产量大于公司的需求量。

【输入】

第一行二个整数n,m，表示需要产品的总量和提供产品的农户个数。

接下来 m 行，每行两个整数，表示第*i* 个农户提供产品单价和能卖出的产品数量。

【输出】

输出公司收购农产品的最小费用。

【输入样例】

100 5

5 20

9 40

3 10

8 80

6 30

【输出样例】

630

**22.** **删数字问题（贪心算法）**

【问题描述】

给定n个纯数字组成的数字串，删除其中k(k<n)个数字后，剩下的数字按原来的顺序组成一个新的正整数，确定删除方案，使得剩下的数字组成的新的正整数最小。

【输入】

第1行输入由n个纯数字组成的字符串；

第2行输入整数k；

【输出】

输出一个整数，表示最后剩下的最小数。

【输入样例】

178542

4

【输出样例】

12

**23. 安排工作以达到最大收益（贪心算法）**

【问题描述】

你有n个工作和 m 个工人。给定三个数组：difficulty,profit和worker，其中:

difficulty[i]表示第 i 个工作的难度，profit[i] 表示第 i 个工作的收益。

worker[i] 是第 i 个工人的能力，即该工人只能完成难度小于等于 worker[i] 的工作。每个工人最多只能安排一个工作，但是一个工作可以完成多次 。

举个例子，如果 3 个工人都尝试完成一份报酬为 $1 的同样工作，那么总收益为 $3。如果一个工人不能完成任何工作，他的收益为 $0 。

返回 在把工人分配到工作岗位后，我们所能获得的最大利润 。

示例：

输入: difficulty = [2,4,6,8,10], profit = [10,20,30,40,50], worker = [4,5,6,7]

输出: 100

解释: 工人被分配的工作难度是 [4,4,6,6] ，分别获得 [20,20,30,30] 的收益。

**24.轮班监督委员会（贪心算法）**

学生会主席带着以下问题来找你。她负责一群学生的轮班安排。不同的轮班有不同的工作，但我们可以把每个班次看作是一个连续几天的任务。一个轮班可能需要多名同学参加，并且一名同学可能会参加多次轮班，但一位同学的多次轮班之间不能有时间间隔，当然也不能有重叠，例如，一位同学可以参加周1到周2，与周3到周4的两次轮班；但参加周1到周3，与周2到周4的两次轮班不合法；参加周1到周2，与周4到周4的两次轮班也不合法。可以有多个轮班同时进行。学生会主席正试图从这些学生中找到一个子集，组成一个监督委员会，以每周与之会面一次。她认为这样一个委员会是合格的：对于每一个不在委员会中的学生来说，该学生的班次与委员会中的某个学生的班次重叠（至少有部分时间重叠）。这样一来，每个学生的表现可以被至少一个在委员会任职的人观察到。

给出一个有效的算法，该算法采用n个班次的时间表并产生一个完整的监督委员会，其中包含尽可能少的学生。

【输入格式】

N个班次的时间表，n行，每行3列及以上，其中前两列是该班次的起始时间，后面的若干列是该班次学生的学号。

1 1 1 5

1 3 2 6 8

2 3 3 5 7

3 3 4 9

4 4 4

4 6 10

4 7 9

5 6 11

6 6 12

【输出格式】

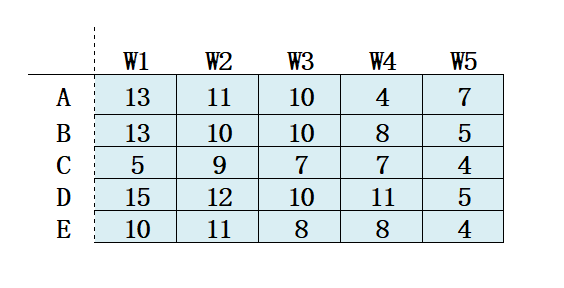
监督委员会的成员

5 9

**25.回溯法求工作安排问题 （回溯法）**

【问题描述】

现有五项工作W1、W2、W3、W4、W5需要分配给A、B、C、D、E 5人，每人只能选择一项，他们开展工作得到的效益如下图所示。



请设计回溯算法，在各种选择组合中，找到效益最高的一种组合。

【输入】

输入5行5列的整数，表示每人开展各项工作的效益。

【输出】

第1行输出各项工作的安排情况，第2行输出最大效益。

【输入样例】

13 11 10 4 7

13 10 10 8 5

5 9 7 7 4

15 12 10 11 5

10 11 8 8 4

【输出样例】

W5 W3 W4 W1 W2

50

**26.回溯法求解装载问题 （回溯法）**

【问题描述】

给定*n*个货箱,货箱*i* 重为 *wi*， 船可以装载的货箱总重量为 *W*。 货箱装载问题是在不超过载重量的前提下装载尽可能重的货箱。

装载问题的解可以用向量 (*x1,x2,…,xn*)表示，*xi*∈{0,1}, *xi* =1 表示货箱*i*被装上船， *xi* =0表示货箱 *i*不装上船。请设计一个基于回溯算法的装载问题的求解方案。

【输入】

第1行输入正整数n和船的装载重量W，第2行依次输入n个整数，表示n个货箱的重量。

【输出】

输出两行，第1行输出最大的装载重量，第2行输出解向量 (*x1,x2,…,xn*)。

【输入样例】

6 26

1 7 4 20 9 11

【输出样例】

25

1 0 1 1 0 0

**27. 完成工作的最短时间（回溯法）**

【问题描述】

给你一个整数数组 jobs ，其中 jobs[i] 是完成第 i 项工作要花费的时间。

请你将这些工作分配给 k 位工人。所有工作都应该分配给工人，且每项工作只能分配给一位工人。工人的工作时间是完成分配给他们的所有工作花费时间的总和。请你设计一套最佳的工作分配方案，使工人的最大工作时间得以最小化。

返回分配方案中尽可能最小的最大工作时间 。

【输入】

jobs = [3,2,3], k = 3

【输出】

3

**【**解释**】**给每位工人分配一项工作，最大工作时间是 3 。

**28. 输出给定集合的所有子集（回溯算法）**

给定数组S，打印该数组的所有子集。

**示例1**

|  |
| --- |
| Plain Text **Input:** 2,4 **Output:**  {} {2} {2, 4} {4} |

**示例2**

|  |
| --- |
| Plain Text **Input:** 1,2,3 **Output:**  {} {1} {1, 2} {1, 2, 3} {1, 3} {2} {2, 3} {3} |

说明:这些是给定数组可以形成的所有子集，可以证明除了给定输出之外不存在其他子集。

**29. 拔河比赛（回溯算法）**

给定一个整数集合，将其分成两个大小几乎相同（偶数长度相同，奇数相差为1）子集和，求的最小值，其中 Sum 是集合求和函数。

**示例1**

|  |
| --- |
| Plain Text **Input**: 3,4,5,-3,100,1,89,54,23,20 **Output**: S1: {4,100,1,23,20}, Sum(S1)=148 S2: {3,5,-3,89,54}, Sum(S2)=148 Diff: 0 |

**示例2**

|  |
| --- |
| Plain Text **Input**: 23,45,-34,12,0,98,-99,4,189,-1,4 **Output**: S1: {45,-34,12,98,-1}, Sum(S1)=120 S2: {23,0,-99,4,189,4}, Sum(S2)=121 Diff: 1 |

**30.最小长度电路板排列问题（分支限界法）**

【问题描述】

最小长度电路板排列问题是大规模电子系统设计中提出的实际问题。该问题的提法是，将n 块电路板以最佳排列方案插入带有n 个插槽的机箱中。n 块电路板的不同的

排列方式对应不同的电路板插入方案。

设B={1,2,··,n是n块电路板的集合。集合L=N1,N2,……,Nm是n块电路板的m个连接块。其中每个连接块Ni是 B 的一个子集，且Ni中的电路板用同一根导线连接在一起。在最小长度电路板排列问题中，连接块的长度是指该连接块中第1块电路板到最后1块电路板之间的距离。

试设计一个队列式分支限界法找出所给n个电路板的最佳排列，使得m个连接块中最大长度达到最小。

**算法设计**:对于给定的电路板连接块，设计一个队列式分支限界法，找出所给 n 个电路板的最佳排列，使得 m 个连接块中最大长度达到最小。

**【**输入**】**由文件input.txt给出输入数据。第1行有2个正整数n和m(1<=m,n<=20)。接下来的n行中，每行有m个数。第k行的第j个数为0表示电路板不在连接块j中，为1表示电路板k在连接块i中。

**【**输出**】**将计算的电路板排列最小长度及其最佳排列输出到文件output.txt。文件的第1行是最小长度;接下来的1行是最佳排列

输入文件示例

input.txt

8 5

1 1 1 1 1

0 1 0 1 0

0 1 1 1 0

1 0 1 1 0

1 0 1 0 0

1 1 0 1 0

0 0 0 0 1

0 1 0 0 1

输出文件示例

output.txt

4

5 4 3 1 6 2 8 7