目录

目录

[1处理器问题 3](#_Toc28589)

[2单向链表中间节点 6](#_Toc26419)

[3字符串重新排列、字符串重新排序 8](#_Toc26512)

[4完美走位 10](#_Toc1739)

[5.最多颜色的车辆 13](#_Toc6804)

[6.不含101的数 14](#_Toc21372)

[7租车骑绿岛 15](#_Toc12785)

[8等和子数组最小和 16](#_Toc2358)

[9过滤组合字符串 18](#_Toc16612)

[10真正的密码 20](#_Toc24855)

[11最小调整顺序次数、特异性双端队列 22](#_Toc27578)

[12羊、狼、农夫过河 25](#_Toc1866)

[13打印机队列 27](#_Toc19107)

[14模拟商场优惠打折 29](#_Toc9514)

[15二元组个数 31](#_Toc13263)

[16积木最远距离 32](#_Toc18788)

[17计算数组中心位置 3](#_Toc12168)5

[18最长的密码 3](#_Toc12168)7

[18对称美学 39](#_Toc12168)

[20端口合并 34](#_Toc12168)

[21开放日活动 41](#_Toc1569)

[22简单的自动曝光 43](#_Toc11832)

[23获取最大软件版本号 45](#_Toc7551)

[24预订酒店 47](#_Toc6004)

[25查找重复元素 4](#_Toc6004)9

[26找数字、等值元素 5](#_Toc6004)1

[27整理扑克牌 5](#_Toc6004)2

[28任务总执行时长 54](#_Toc5864)

[29査找单入口空闲区域 56](#_Toc17026)

[30货币单位换算 59](#_Toc11269)

[31猜字谜 62](#_Toc12893)

[32异常的打卡记录 64](#_Toc6498)

[33优秀学员统计 66](#_Toc8983)

[34静态扫描 71](#_Toc27397)

[35机房布局 73](#_Toc7580)

[36 人数最多的站点 74](#_Toc24293)

[37 快递投放问题 7](#_Toc24293)5

[38. 区块链文件转储系统 77](#_Toc26445)

[39 识图谱新词挖掘 79](#_Toc8957)

[40新员工座位 81](#_Toc1059)

[41.最长广播响应 82](#_Toc11066)

[42.考古学家 83](#_Toc208)

[43.贪吃蛇 84](#_Toc24628)

[44.最大社交距离 86](#_Toc8066)

[45.污染水域 87](#_Toc3925)

[46.连续出牌数量 88](#_Toc9237)

[47.简易内存池 89](#_Toc12504)

[48.单词搜索 91](#_Toc634)

[49最长的顺子 92](#_Toc23297)

[50.任务最优调度 93](#_Toc2454)

[51.仿LISP运算 94](#_Toc13326)

[52.学生方阵 95](#_Toc13805)

[53.发广播 96](#_Toc171)

[54.字符串比较 97](#_Toc29395)

[55.斗地主之顺子 98](#_Toc1531)

[56.数据分类 100](#_Toc150)

[57.高效的任务规划 102](#_Toc13710)

[58.完全二叉树非叶子部分后序遍历 104](#_Toc10858)

[59.数组二叉树 105](#_Toc15996)

[60分积木 106](#_Toc24774)

# 1处理器问题

**题目描述**

某公司研发了一款高性能AI处理器。每台物理设备具备8颗**AI**处理器，编号分别为0、1、

2、 3、 4、 5、 6、 7。

编号0-3的处理器处于同一个链路中，编号4-7的处理器处于另外一个链路中，不通链路中 的处理器不能通信。

如下图所示。现给定服务器可用的处理器编号数组array，以及任务申请的处理器数量num， 找出符合下列亲和性调度原则的芯片组合。

如果不存在符合要求的组合，则返回**空列表**

亲和性调度原则：

-如果申请处理器个数为1,则选择同一链路，剩余可用的处理器数量为1个的最佳，其次 是剩余3个的为次佳，然后是剩余2个，最后是剩余4个。

-如果申请处理器个数为2,则选择同一链路剩余可用的处理器数量2个的为最佳，其次是 剩余4个，最后是剩余3个。

-如果申请处理器个数为4,则必须选择同一链路剩余可用的处理器数量为4个。

-如果申请处理器个数为8,则申请节点所有8个处理器。

提示：

1. 任务申请的处理器数量只能是1、2、4、8。
2. 编号0-3的处理器处于一个链路，编号4-7的处理器处于另外一个链路。
3. 处理器编号唯一，且不存在相同编号处理器。

**输入描述**

输入包含可用的处理器编号数组array,以及任务申请的处理器数量num两个部分。

第一行为array，第二行为num。例如：

[0, 1,4, 5,6, 7]

1

表示当前编号为0、1、4、5、6、7的处理器可用。任务申请1个处理器。

* O <= array.length <= 8
* O <= array[i] <= 7
* num in [1, 2> 4, 8]

**输出描述**

输出为组合列表，当 array=[0，1, 4，5, 6，7]，num=1 时，输出为[[0], [1]]。

**用例**

|  |  |
| --- | --- |
|  | [0, 1, 4, 5, 6, 7] |
| 输入 | 1 |
| 输岀 | [[0], [1]] |

根据第一条亲和性调度原则，在剩余两个处理器的链路(0,1, 2, 3)中选择处理器。 由于只有0和1可用，则返回任意一颗处理器即可。

说明

**2单向链表中间节点**

题目描述

求[单向链表](https://so.csdn.net/so/search?q=%E5%8D%95%E5%90%91%E9%93%BE%E8%A1%A8&spm=1001.2101.3001.7020)中间的节点值，如果奇数个节点取中间，偶数个取偏右边的那个值。

输入描述

第一行链表头节点地址后续输入的节点数n

后续输入每行表示一个节点，格式节点地址节点值下一个节点地址（-1表示[空指针](https://so.csdn.net/so/search?q=%E7%A9%BA%E6%8C%87%E9%92%88&spm=1001.2101.3001.7020)） 输入保证链表不会岀现环，并且可能存在一些节点不属于链表。

输出描述

单向链表中间的节点值

用例

00010 4

00000 3 -1

输入 00010 5 12309

11451 6 00000

12309 7 11451

输出 6

说明 无

10000 3

输入

76892 7 12309

12309 5 -1

10000 1 76892

输岀 7

说明 无

# 3字符串重新排列、字符串重新排序

题目描述

给定一个字符串s, s包括以空格分隔的若干个单词，请对S进行如下处理后输岀：

1、 单词内部调整：对每个单词字母重新按[字典序排序](https://so.csdn.net/so/search?q=%E5%AD%97%E5%85%B8%E5%BA%8F%E6%8E%92%E5%BA%8F&spm=1001.2101.3001.7020)

2、 单词间顺序调整：

1） 统计每个单词出现的次数，并按次数[降序排列](https://so.csdn.net/so/search?q=%E9%99%8D%E5%BA%8F%E6%8E%92%E5%88%97&spm=1001.2101.3001.7020)

2） 次数相同，按[单词长度](https://so.csdn.net/so/search?q=%E5%8D%95%E8%AF%8D%E9%95%BF%E5%BA%A6&spm=1001.2101.3001.7020)升序排列

3） 次数和单词长度均相同，按字典升序排列

请输岀处理后的字符串，每个单词以一个空格分隔。

输入描述

一行字符串，每个字符取值范围：［a-zA-z0-9］以及空格，字符串长度范围：［1，1000］

输出描述

输岀处理后的字符串，每个单词以一个空格分隔。

用例

输入 This is an apple

输出 an is This aelpp

说明 无

|  |  |
| --- | --- |
| 输入 | My sister is in the house not in the yard |
| 输岀 | in in eht eht My is not adry ehosu eirsst |
| 说明 | 无 |

**4完美走位**

题目描述

在第一人称射击游戏中，玩家通过键盘的A、S、D、W四个按键控制游戏人物分别向左、 向后、向右、向前进行移动，从而完成走位。

假设玩家每按动一次键盘，游戏任务会向某个方向移动一步，如果玩家在操作一定次数的键 盘并且各个方向的步数相同时，此时游戏任务必定会回到原点，则称此次走位为完美走位。

现给定玩家的走位（例如：ASDA），请通过更换其中**一段连续走位的方式**使得原走位能够 变成一个完美走位。其中待更换的连续走位可以是相同长度的任何走位。

请返回待更换的连续走位的最小可能长度。

如果原走位本身是一个完美走位，则返回0。

输入描述

输入为由键盘字母表示的走位s，例如：ASDA

输出描述

输出为待更换的连续走位的最小可能长度。

用例

输入 WASDAASD

输岀 1

说明

将第二个A替换为W,即可得到完美走位

输入 AAAA

输岀 3

说明 将其中三个连续的A替换为WSD,即可得到完美走位

**5**.**最多颜色的车辆**

题目描述

在一个狭小的路口，每秒只能通过一辆车，假设车辆的颜色只有3种，找出N秒内经过 的最多颜色的车辆数量。

三种颜色编号为0,1,2

输入描述

第一行输入的是通过的车辆颜色信息

[0,1,1,2]代表4秒钟通过的车辆颜色分别是0 , 1 , 1 , 2

第二行输入的是统计时间窗，整型，单位为秒

输出描述

输出指定时间窗内经过的最多颜色的车辆数量。

用例

0 1 2 1

输入

3

输岀 2

说明 在3秒时间窗内，每个颜色最多出现2次。例如：[1,2,1]

**6.不含101的数**

题目描述

小明在学习二进制时，发现了一类不含101的数，也就是：

将数字用二进制表示，不能出现101。

现在给定一个整数区间［l,r］，请问这个区间包含了多少个不含101的数？

输入描述

输入的唯一一行包含两个正整数I， r( 1 <l <r<10A9)o

输出描述

输出的唯一一行包含一个整数，表示在［l,r］区间内一共有几个不含101的数。

|  |  |
| --- | --- |
| 输入 | 10 20 |
| 输出 | 7 |
| 说明 | 区间［10, 20］内，满足条件的数字有［12, 14, 15, 16, 17, 18, 19］因此答案为7。 |
| 输入 | 1 10 |
| 输出 | 8 |
| 说明 | 区间［1,10］内，5的二进制表示为101 , 10的二进制表示为1010，因此区间［1 ,10 ］内有10-2=8个不含101的数。 |

**7租车骑绿岛**

题目描述

部门组织绿岛骑行团建活动。租用公共双人自行车，每辆自行车最多坐两人，最大载重M。

给出部门每个人的体重，请问最多需要租用多少双人自行车。

输入描述

第一行两个数字m、n，分别代表自行车限重，部门总人数。

第二行，n个数字，代表每个人的体重，体重都小于等于自行车限重m。

* 0<m<=200
* 0<n<=1000000

输出描述

最小需要的双人自行车数量。

用例

|  |  |
| --- | --- |
| 输入 | 3 4  3 2 2 1 |
| 输出 | 3 |
| 说明 | 无 |

**8等和子数组最小和**

题目描述

给定一个数组nums,将元素分为若干个组，使得每组和相等，求出满足条件的所有分组中,

组内元素和的最小值。

输入描述

第一行输入m

接着输入m个数，表示此数组nums

数据范围：1<=m<=50, 1<=nums[i]<=50

输出描述

最小拆分数组和

用例

|  |  |
| --- | --- |
| 输入 | 7  4 3 2 3 5 2 1 |
| 输岀 | 5 |
| 说明 | 可以等分的情况有：  4 个子集(5), (1,4), (2,3), (2,3)  2 个子集(5,1,4), (2,3, 2,3)  但最小的为5。 |

**9过滤组合字符串**

题目描述

数字0、1、2、3、4、5、6、7、8、9分别关联a~z 26个英文字母。

* 0 关联 “a”,”b”,”c”
* 1 关联 “d”,”e”,”f”
* 2 关联 “g”,”h”,”i”
* 3 关联 “j”,”k”,”l”
* 4 关联 “m”,”n”,”o”
* 5 关联 “p”,”q”,”r”
* 6 关联 “s”,”t”
* 7 关联 “u”,”v”
* 8 关联 “w”,”x”
* 9 关联 “y”,”z”

输入一串数字后，通过数字和字母的对应关系可以得到多个字母字符串（要求按照数字的顺序组合字母字符串）；屏蔽字符串：屏蔽字符串中的所有字母不能同时在输出的字符串出现，如屏蔽字符串是abc，则要求字符串中不能同时出现a,b,c，但是允许同时出现a,b或a,c或b,c等；

给定一个数字字符串和一个屏蔽字符串，输出所有可能的字符组合；

例如输入数字字符串78和屏蔽字符串ux，输出结果为uw，vw，vx；数字字符串78，可以得到如下字符串uw，ux，vw，vx；由于ux是屏蔽字符串，因此排除ux，最终的输出是uw，vw，vx;

输入描述

第一行输入为一串数字字符串，数字字符串中的数字不允许重复，数字字符串的长度大于0，小于等于5；

第二行输入是屏蔽字符串，屏蔽字符串的长度一定小于数字字符串的长度，屏蔽字符串中字符不会重复；

输出描述

输出可能的字符串组合

注：字符串之间使用逗号隔开，最后一个字符串后携带逗号

用例

输入 78 ux

输出 uw vx vw

说明 ux完全包含屏蔽字符串ux,因此剔除

**10真正的密码**

题目描述

在一行中输入一个[字符串数组](https://so.csdn.net/so/search?q=%E5%AD%97%E7%AC%A6%E4%B8%B2%E6%95%B0%E7%BB%84&spm=1001.2101.3001.7020)，如果其中一个字符串的所有以索引0开头的子串在数组中 都有，那么这个字符串就是潜在密码，

在所有潜在密码中最长的是真正的密码，如果有多个长度相同的真正的密码，那么取[字典序](https://so.csdn.net/so/search?q=%E5%AD%97%E5%85%B8%E5%BA%8F&spm=1001.2101.3001.7020) 最大的为唯一的真正的密码，求唯一的真正的密码。

输入描述

无

输出描述

无

**用例**

|  |  |
| --- | --- |
| 输入 | h he hel hell hello o ok n ni nin ninj ninja |
| 输岀 | ninja  按要求，hello、ok、[ninja](https://so.csdn.net/so/search?q=ninja&spm=1001.2101.3001.7020)都是潜在密码。 |
| 说明 | 检查长度，hello、ninja是真正的密码。  检查字典序，ninja是唯一真正密码。 |
| 输入 | a b c d f |
| 输岀 | f |
| 说明 | 按要求，abcdf都是潜在密码。 检查长度，abcdf是真正的密码。 检查字典序，f是唯一真正密码。  说明 |

**11最小调整顺序次数、特异性双端队列**

题目描述

有一个特异性的[双端队列](https://so.csdn.net/so/search?q=%E5%8F%8C%E7%AB%AF%E9%98%9F%E5%88%97&spm=1001.2101.3001.7020)，该队列可以从头部或尾部添加数据，但是只能从头部移出数据。

小A依次执行2n个指令往队列中添加数据和移出数据。其中n个指令是添加数据（可能从

头部添加、也可能从尾部添加），依次添加1到n； n个指令是移出数据。

现在要求移除数据的顺序为1到n。

为了满足最后输出的要求，小A可以在任何时候调整队列中数据的顺序。

请问小A最少需要调整几次才能够满足移除数据的顺序正好是1到n；

输入描述

第一行一个数据n，表示数据的范围。

接下来的2n行，其中有n行为添加数据，指令为：

* **"**head add x"表示从头部添加数据x，
* **"**tail add x"表示从尾部添加数据x，

另外n行为移出数据指令，指令为："remove"的形式，表示移出1个数据；

1 <n <3\* 10A5o

所有的数据均合法。

输出描述

一个整数，表示小A要调整的最小次数。

用例

|  |  |
| --- | --- |
| 输入 | 5  head add 1 tail add 2 remove head add 3 tail add 4 head add 5 remove remove remove remove |
| 输岀 | 1 |
| 说明 | 无 |

**12羊、狼、农夫过河**

题目描述

羊、狼、农夫都在岸边，当羊的数量小于狼的数量时，狼会攻击羊，农夫则会损失羊。农夫

有一艘容量固定的船，能够承载固定数量的动物。

要求求出不损失羊情况下将全部羊和狼运到对岸需要的最小次数。

只计算农夫去对岸的次数，回程时农夫不会运送羊和狼。

备注：农夫在或农夫离开后羊的数量大于狼的数量时狼不会攻击羊。

输入描述

第一行输入为M，N，X，分别代表羊的数量，狼的数量，小船的容量。

输出描述

输出不损失羊情况下将全部羊和狼运到对岸需要的最小次数（若无法满足条件则输出0）。

用例

输入 5 3 3

输岀 3

第一次运2只狼

说明 第二次运3只羊

第三次运2只羊和1只狼

输入 5 4 1

输出 0

说明 如果找不到不损失羊的运送方案，输出0

**13打印机队列**

题目描述

有5台打印机打印文件，每台打印机有自己的待打印队列。

因为打印的文件内容有轻重缓急之分，所以队列中的文件有1~10不同的代先级，其中**数字**

**越大优先级越高**

打印机会从自己的待打印队列中选择**优先级最高**的文件来打印。

如果存在两个优先级一样的文件，则选择**最早进入队列***的*那个文件。

现在请你来模拟这5台打印机的打印过程。

输入描述

每个输入包含1个测试用例，

每个测试用例第一行给出发生事件的数量N(0< N< 1000)。

接下来有N行，分别表示发生的事件。共有如下两种事件：

1. “IN P NUM"，表示有一个拥有优先级NUM的文件放到了打印机P的待打印队列

中。(0< P<=5, 0< NUM <= 10)；

1. “OUT P"，表示打印机P进行了一次文件打印，同时该文件从待打印队列中取出。

(0<P<=5)。

输出描述

* 对于每个测试用例，每次”OUTP”事件，请在一行中**输出文件的编号**。
* 如果此时没有文件可以打印，请输出” **NULL**"。
* 文件的编号定义为”IN PNUM”事件发生第x次，此处待打印文件的编号为X。编号 从**1**开始。

用例

|  |  |
| --- | --- |
| 输入 | 7  IN 1 1  IN 1 2  IN 1 3  IN 2 1  OUT 1  OUT 2  OUT 2 |
| 输岀 | 3  4  NULL |
| 说明 | 无 |
| 输入 | 5  IN 1 1  IN 1 3  IN 1 1  IN 1 3  OUT 1 |
| 输岀 | 2 |
| 说明 | 无 |

**14模拟商场优惠打折**

题目描述

模拟商场优惠打折，有三种[优惠券](https://so.csdn.net/so/search?q=%E4%BC%98%E6%83%A0%E5%88%B8&spm=1001.2101.3001.7020)可以用，满减券、打折券和无门槛券。

满减券：满100减10,满200减20,满300减30,满400减40，以此类推不限制使用；

打折券：固定折扣92折，且打折之后[向下取整](https://so.csdn.net/so/search?q=%E5%90%91%E4%B8%8B%E5%8F%96%E6%95%B4&spm=1001.2101.3001.7020)，每次购物只能用1次；

无门槛券：一张券减5元，没有使用限制。

每个人结账使用优惠券时有以下限制：

每人每次只能用两种优惠券，并且同一种优惠券必须一次用完，不能跟别的穿插使用（比如 用一张满减，再用一张打折，再用一张满减，这种顺序不行）。

求不同使用顺序下每个人用完券之后得到的最低价格和对应使用优惠券的总数;如果两种顺 序得到的价格一样低，就取使用优惠券数量较少的那个。

输入描述

第一行三个数字m,n,k，分别表示每个人可以使用的满减券、打折券和无门槛券的数量；

第二行一个数字x,表示有几个人购物；

后面x行数字，依次表示是这几个人打折之前的商品总价。

输出描述

输出每个人使用券之后的最低价格和对应使用优惠券的数量

用例

输入 3 2 5

3

100

200

400

输出 65 6

135 8

275 8

输入：

第一行三个数字m,n,k,分别表示每个人可以使用的满减券、打折券和无门槛券的数量。

说明

输岀:

第一个人使用

张满减券和5张无门槛券价格最低。

(100-10=90, 90-5\*5=65)

第二个人使用

张满减券和5张无门槛券价格最低。

(200-20-10-10=160, 160-5\*5 = 135)

第二个人使用

张满减券和5张无门槛券价格最低。

(400-40-30-30=300, 300 - 5\*5=275)

**15二元组个数**

题目描述

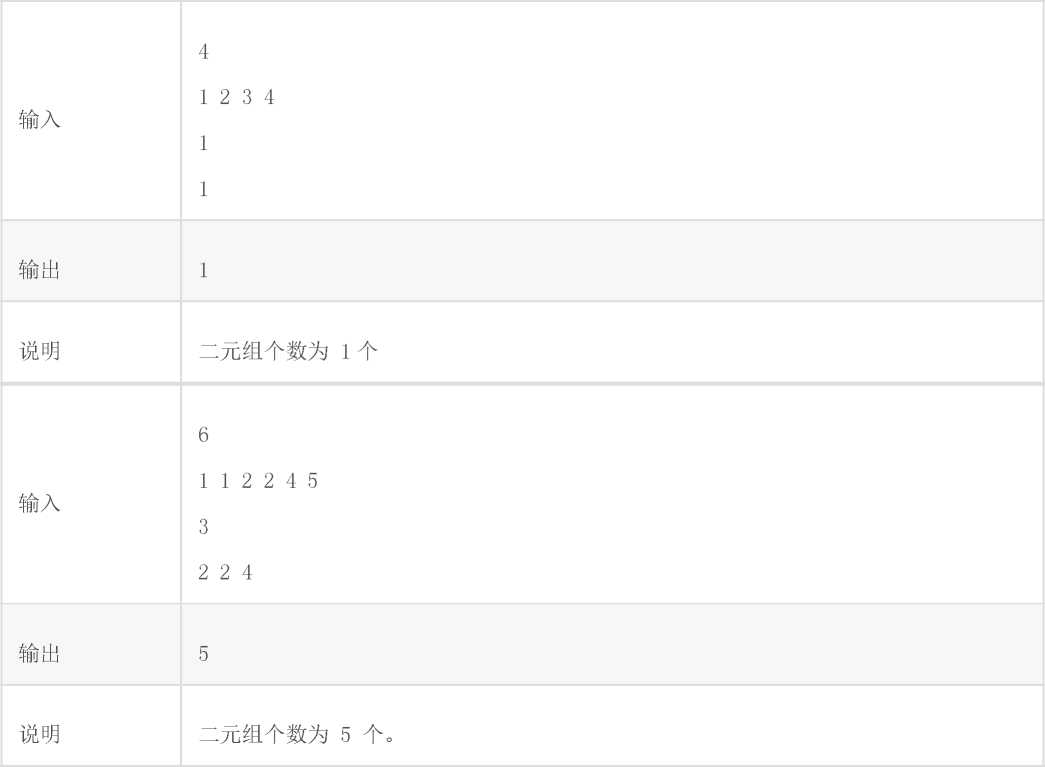
给定两个数组a，b，若a[i] == b[j]则称[i, j]为一个二元组，求在给定的两个数组中,

元组的个数。

输入描述

第一行输入m

第二行输入m个数，表示第一个数组

第三行输入n

第四行输入n个数，表示第二个数组

输出描述：二元组个

**16积木最远距离**

题目描述

小华和小薇一起通过玩积木游戏学习数学。

他们有很多积木，每个积木块上都有一个数字，积木块上的数字可能相同。

小华随机拿一些积木挨着排成一排，请小薇找到这排积木中数字相同且所处位置最远的2

块积木块，计算他们的距离，小薇请你帮忙替她解决这个问题。

输入描述

第一行输入为N，表示小华排成一排的积木总数。

接下来N行每行一个数字，表示小华排成一排的积木上数字。

输出描述

相同数字的积木的位置最远距离；如果所有积木数字都不相同，请返回-1。

备注

* 0<=积木上的数字<10人9
* 1<=积木长度<=10人5

用例



**17计算数组中心位置**

题目描述

给你一个整数数组nums,请计算数组的中心位置，数组的中心位置是数组的一个下标，其

左侧所有元素相乘的积等于右侧所有元素相乘的积。数组第一个元素的左侧积为1,最后一 个元素的右侧积为1。

如果数组有多个中心位置，应该返回最靠近左边的那一个，如果数组不存在中心位置，返回

**-1**

输入描述

输入只有一行，给出N个正整数用空格分隔：nums = 2 53656

1 <= nums.length <= 1024

1 <= nums[i] <= 10

输出描述

输出：3

解释：中心位置是3

用例

输入 253656

输岀 3

说明

题目描述

小王在进行游戏大闯关，有一个关卡需要输入一个密码才能通过，密码获得的条件如下：

在一个密码本中，每一页都有一个由26个小写字母组成的若干位密码，每一页的密码不同,

需要从这个密码本中寻找这样一个最长的密码，

从它的末尾开始依次去掉一位得到的新密码也在密码本中存在。

请输出符合要求的密码，如果有多个符合要求的密码，则返回**字典序最大**的密码。

若没有符合要求的密码，则返回**空字符串**

输入描述

密码本由一个[字符串数组](https://so.csdn.net/so/search?q=%E5%AD%97%E7%AC%A6%E4%B8%B2%E6%95%B0%E7%BB%84&spm=1001.2101.3001.7020)组成，不同元素之间使用空格隔开，每一个元素代表密码本每一页

的密码。

输出描述

一个字符串

用例

输入

h he hel hell hello

|  |  |
| --- | --- |
| 输出 | hello |
| 说明 | 无 |

b ereddred bw bww bwwl bwwlm bwwln

输入

输出 bwwln

说明 无

题目描述

对称就是最大的美学，现有一道关于对称字符串的美学。已知：

* 第1个字符串：R
* 第2个字符串：BR
* 第3个字符串：RBBR
* 第4个字符串：BRRBRBBR
* 第5个字符串：RBBRBRRBBRRBRBBR

相信你已经发现规律了，没错！就是第i个字符串=第i-1号字符串[取反](https://so.csdn.net/so/search?q=%E5%8F%96%E5%8F%8D&spm=1001.2101.3001.7020)+第i-1号

字符串；

取反(R->B, B->R);

现在告诉你n和k，让你求得第n个字符串的第k个字符是多少。(k的编号从0开始)

输入描述

第一行输入一个T，表示有T组用例；

解析来输入T行，每行输入两个数字，表示n，k

. 1 <T< 100；

* 1 < n <64；

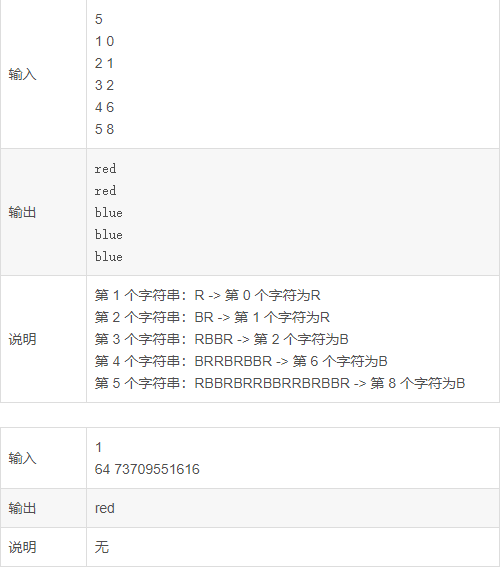
. 0<k < 2A(n-1)；

输出描述

输岀T行表示答案；输出 "blue" 表示字符是B；输出 "red" 表示字符是R。

备注：输出字符串区分大小写，请注意输出小写字符串，不带双引号。

用例



题目描述

有 M(1<=M<=10)个端 口组，

每个端口组是长度为N(1<=N<=100)的整数数组，

如果端口组间存在2个及以上不同端口相同，则认为这2个端口组互相关联，可以合并。

第一行输入端口组个数M，再输入M行，每行逗号分隔，代表端口组。

输出合并后的端口组，用二维数组表示。

输入描述

第一行输入一个数字M

第二行开始输入M行，每行是长度为N的整数数组，用逗号分割

4

4

输入 2, 3, 2

1,2

5

|  |  |
| --- | --- |
| 输岀 | [[4], [2, 3, 2], [1,2], [5]] |
| 说明 | 仅有一个端口 2相同，不可以合并。 |
| 输入 | 3  2.3.1  4.3.2  5 |
| 输岀 | [[1, 2, 3, 4], [5]] |
| 说明 | 无 |
| 输入 | 6  10  4,2,1  9  3, 6, 9, 2  6,3,4  8 |
| 输岀 | [[10], [1,2, 3, 4, 6, 9], [9], [8]] |
| 说明 | 无 |
| 输入 | 11 |
| 输岀 | [[]] |
| 说明 | 无 |

**21开放日活动**

题目描述

某部门开展**Family Day**开放日活动，其中有个从桶里取球的游戏，游戏规则如下：

有N个容量一样的小桶等距排开，

且每个小桶都默认装了数量不等的小球，

每个小桶装的小球数量记录在数组bucketBallNums中，

游戏开始时，要求所有桶的小球总数不能超过SUM,

如果小球总数超过SUM,则需对所有的小桶统一设置一个容量最大值maxCapacity，

并需将超过容量最大值的小球拿出来，直至小桶里的小球数量小于maxCapacity;

请您根据输入的数据，计算从每个小桶里拿岀的小球数量。

限制规则一：

所有小桶的小球总和小于SUM,则无需设置容量值maxCapacity，并且无需从小桶中拿球

出来，返回结果[]

限制规则二：

如果所有小桶的小球总和大于SUM,则需设置容量最大值maxCapacity，并且需从小桶中

拿球出来，返回从每个小桶拿出的小球数量组成的数组；

输入描述

第一行输入2个正整数，数字之间使用空格隔开，其中第一个数字表示SUM,第二个数字

表示bucketBallNums数组长度；

第二行输入N个正整数，数字之间使用空格隔开，表示bucketBallNums的每一项；

输出描述

找到一个maxCapacity，来保证取出尽量少的球，并返回从每个小桶拿出的小球数量组成的 数组。

用例

|  |  |
| --- | --- |
| 输入 | 14 7  2 3 2 5 5 1 4 |
| 输岀 | [0, 1, 0, 3, 3, 0, 2] |
| 说明 | 小球总数为22，SUM=14，超出范围了，需从小桶取球，  maxCapacity=1，取出球后，桶里剩余小球总和为7，远小于14  maxCapacity=2，取出球后，桶里剩余小球总和为13， maxCapacity=3，取出球后，桶里剩余小球总和为16，大于14 因此maxCapacity为2 ，每个小桶小球数量大于2的都需要拿出来； |

**22简单的自动曝光**

题目描述

一个图像有n个像素点，存储在一个长度为n的数组img里，每个像素点的取值范围［0,255］ 的正整数。

请你给图像每个像素点值加上一个整数k （可以是负数），得到新图newImg，使得新图 newImg的所有像素平均值最接近中位值128。

请输岀这个整数k。

输入描述

n个整数，中间用空格分开

输出描述

一个整数k

备注

• 1 <= n <= 100

•如有多个整数k都满足，输出小的那个k；

•新图的像素值会自动截取到［0,255］范围。当新像素值<0,其值会更改为0；当新像素值>255, 其值会更改为255；

例如newlmg=”-1 -2 256"，会自动更改为”0 0 255"

用例



**23获取最大软件版本号**

题目描述

*Maven*版本号定义，〈主版本＞.〈次版本＞.〈增量版本＞-〈里程碑版本〉，举例3.1.4-beta

其中，主版本和次版本都是必须的，主版本，次版本，增量版本由多位数字组成，可能包含 前导零，里程碑版本由字符串组成。

〈主版本＞.〈次版本＞.＜增量版本〉：基于**数字**比较；例如

1.5 > 1.4 > 1.3.11 > 1.3.9

里程碑版本：基于**字符串**比较，采用**字典序；**例如

1.2-beta-3 ＞ 1.2-beta-11

比较版本号时，按从左到右的顺序依次比较。

基于数字比较，只需比较**忽略任何前导零后的整数值**

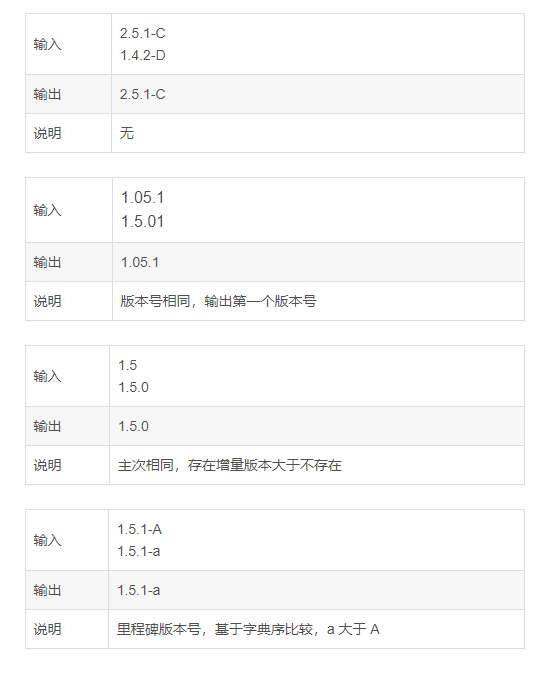
输入2个版本号，输出**最大版本号**

输入描述

输入2个版本号，换行分割，每个版本的最大长度小于50

输出描述

版本号相同时输出第一个输入版本号



**24预订酒店**

题目描述

放暑假了，小明决定到某旅游景点游玩，他在网上搜索到了各种价位的酒店（长度为n的

数组A），他的心理价位是x元，请帮他筛选出k个最接近x元的酒店（n>=k>0）,并**由低 到高**打印酒店的价格。

输入描述

第一行：n, k, x

第二行：A[0] A[1] A[2]^A[n-1]

输出描述

从低到高打印筛选出的酒店价格

用例

10 5 6

输入

123456789 10

输出 45678

说明 无

10 4 6

输入

10 987654321

说明 无

6 3 1000 输入

30 30 200 500 70 300

输出 200 300 500

说明 无

题目描述

小明负责维护项目下的代码，需要查找出重复代码，用以支撑后续的代码优化，请你帮助小 明找岀重复的代码。

重复代码查找方法：以字符串形式给定两行代码（字符串长度1〈length <= 100,由英文 字母、数字和空格组成），找出两行代码中的[最长公共子串](https://so.csdn.net/so/search?q=%E6%9C%80%E9%95%BF%E5%85%AC%E5%85%B1%E5%AD%90%E4%B8%B2&spm=1001.2101.3001.7020)

注：如果不存在公共子串，返回空字符串

输入描述

输入的参数textl, text2分别表示两行代码

输出描述

输岀任一最长公共子串

用例

|  |  |
| --- | --- |
| 输入 | hello123world  hello123abc4 |
| 输岀 | hello123 |
| 说明 | 无 |

private\_void\_method

输入

public\_void\_method

|  |  |
| --- | --- |
| 输岀 | \_void\_method |
| 说明 | 无 |
| 输入 | hiworld  hiweb |
| 输岀 | hiw |
| 说明 | 无 |

题目描述

给一个二维数组nums,对于每一个元素nums[i],找出距离最近的且值相等的元素，输出

横纵坐标差值的绝对值之和，如果没有等值元素，则输出-1。

输入描述

输入第一行为二维数组的行

输入第二行为二维数组的列

输入的数字以空格隔开。

输出描述

数组形式返回所有坐标值。

用例

|  |  |
| --- | --- |
|  | 3  5 |
| 输入 | 0 3 5 4 2  2 5 7 8 3  2 5 4 2 4 |
| 输出 | [[-1, 4, 2, 3, 3], [1, 1, -1, -1, 4], [1, 1, 2, 3, 2]] |

说明

题目描述

给定一组数字，表示扑克牌的牌面数字，忽略扑克牌的花色，请按如下规则对这一组扑克牌

进行整理：

**步骤1.**对扑克牌进行分组，形成组合牌，规则如下：

・ 当牌面数字相同张数大于等于4时，组合牌为“炸弹”；

・3张相同牌面数字+ 2张相同牌面数字，且3张牌与2张牌不相同时，组合牌为“葫 芦”；

・3张相同牌面数字，组合牌为“三张”；

・2张相同牌面数字，组合牌为“对子”；

・剩余没有相同的牌，则为“单张”；

**步骤2.**对上述组合牌进行由大到小排列，规则如下：

* 不同类型组合牌之间由大到小排列规则：“炸弹”〉“葫芦”〉“三张”〉“对子”〉“单张”;
* 相同类型组合牌之间，除“葫芦”外，按组合牌全部牌面数字加总由大到小排列；

・“葫芦”则先按3张相同牌面数字加总由大到小排列，3张相同牌面数字加总相同时, 再按另外2张牌面数字加总由大到小排列；

* 由于“葫芦”〉“三张”，因此如果能形成更大的组合牌，也可以将“三张”拆分为2张和1 张，其中的2张可以和其它“三张”重新组合成“葫芦”，剩下的1张为“单张”

**步骤3.**当存在多个可能组合方案时，按如下规则排序取最大的一个组合方案：

* 依次对组合方案中的组合牌进行大小比较，规则同上；
* 当组合方案A中的第n个组合牌大于组合方案B中的第n个组合牌时，组合方案A 大于组合方案B；

输入描述

第一行为空格分隔的N个正整数，每个整数取值范围[1,13], N的取值范围[1,1000]

输出描述

经重新排列后的扑克牌数字列表，每个数字以空格分隔

用例

|  |  |
| --- | --- |
| 输入 | 1 3 3 3 2 1 5 |
| 输岀 | 3 3 3 1 1 5 2 |
| 说明 | 无 |
| 输入 | 4421213334 |
| 输岀 | 4443322113 |
| 说明 | 无 |

28任务总执行时长

题目描述

任务编排服务负责对任务进行组合调度。

参与编排的任务有两种类型，其中一种执行时长为taskA，另一种执行时长为taskB。

任务一旦开始执行不能被打断，且任务可连续执行。

服务每次可以编排num个任务。

请编写一个方法，生成每次编排后的任务所有可能的总执行时长。

输入描述

第1行输入分别为第1种任务执行时长taskA，

第2种任务执行时长taskB，

这次要编排的任务个数num，以逗号分隔。

注：每种任务的数量都大于本次可以编排的任务数量

* 0 < taskA
* 0 < taskB
* 0 <= num <= 100000

输出描述

数组形式返回所有总执行时时长，需要按从小到大排列。

用例

输入 1, 2, 3

输出 [3 4, 5, 6]

说明 无

**29査找单入口空闲区域**

题目描述

给定一个mxn的矩阵，由若干字符’X'和‘O'构成，’X'表示该处已被占据，’O'表示该处

空闲，请找到最大的单入口空闲区域。

解释：

空闲区域是由连通的'O'组成的区域，位于边界的'O'可以构成入口，

单入口空闲区域即有且只有一个位于边界的'O '作为入口的由连通的'O '组成的区域。

如果两个元素在水平或垂直方向相邻，则称它们是“连通”的。

输入描述

第一行输入为两个数字，第一个数字为行数m，第二个数字为列数n，两个数字以空格分隔,

1<=m,n<=200o

剩余各行为矩阵各行元素，元素为X或’O'，各元素间以空格分隔。

输出描述

若有唯一符合要求的最大单入口空闲区域，输出三个数字

* 第一个数字为入口行坐标（0~m-1）
* 第二个数字为入口列坐标（0~n-1）

・第三个数字为区域大小

三个数字以空格分隔；

若有多个符合要求，则输出区域大小最大的,若多个符合要求的单入口区域的区域大小相同, 则此时只需要输出区域大小，不需要输出入口坐标。若没有，输出NULL。

用例

|  |  |
| --- | --- |
| 输入 | 4 4  X X X X  X O O X  X O O X  X O X X |
| 输岀 | 3 1 5 |
| 说明 | 存在最大单入口区域，入口坐标(3,1),区域大小5 |
| 输入 | 45  XXXXX  OOOOX  XOOOX  XOXXO |
| 输岀 | 341 |
| 说明 | 存在最大单入口区域，入口坐标(3,4)，区域大小1 |
| 输入 | 54  XXXX  XOOO  XOOO  XOOX  XXXX |
| 输岀 | NULL |

|  |  |
| --- | --- |
| 说明 | 不存在最大单入口区域 |
| 输入 | 5 4  X X X X  X O O O  X X X X  X O O O  X X X X |
| 输岀 | 3 |
| 说明 | 存在两个大小为3的最大单入口区域，两个入口坐标分别为（1,3）、（3,3） |

30货币单位换算

题目描述

记账本上记录了若干条多国货币金额，需要转换成人民币分（fen），汇总后输出。

每行记录一条金额，金额带有货币单位，格式为数字+单位，可能是单独元，或者单独分，

或者元与分的组合。

要求将这些货币全部换算成人民币分（fen）后进行汇总，汇总结果仅保留整数，小数部分

舍弃。

元和分的换算关系都是1:100,如下：

1CNY=100fen（1 元=100 分）

1HKD=100cents（1 港元=100 港分）

1JPY=100sen（1 日元=100 仙）

1EUR=100eurocents（1 欧元=100 欧分）

1GBP=100pence（1 英镑=100 便士）

汇率表如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CNY | JPY | HKD | EUR | GBP |
| 100 | 1825 | 123 | 14 | 12 |

即：100CNY = 1825JPY = 123HKD = 14EUR = 12GBP

输入描述

第一行输入为N, N表示记录数。0<N<100

之后N行，每行表示一条货币记录，且该行只会是一种货币。

输出描述

将每行货币转换成人民币分（fen）后汇总求和，只保留整数部分。

输岀格式只有整数数字，不带小数，不带单位。

用例

|  |  |
| --- | --- |
| 输入 | 1  100CNY |
| 输岀 | 10000 |
| 说明 | 100CNY转换后是10000fen，所以输出结果为10000 |
| 输入 | 1  3000fen |
| 输岀 | 3000 |
| 说明 | 3000fen，结果就是3000 |
| 输入 | 1  123HKD |
| 输岀 | 10000 |
| 说明 | HKD与CNY的汇率关系是123:100,所以换算后，输出结果为10000 |

2

输入 20CNY53fen

53HKD87cents

输出 6432

说明 20元53分+ 53港元87港分，换算成人民币分后汇总，为6432

31猜字谜

题目描述

小王设计了一个简单的猜字谜游戏，游戏的谜面是一个错误的单词，比如nesw,玩家需要 猜出谜底库中正确的单词。猜中的要求如下：

对于某个谜面和谜底单词，满足下面任一条件都表示猜中：

1. 变换顺序以后一样的，比如通过变换w和e的顺序，“nwes"跟"news"是可以完全对

应的；

1. 字母去重以后是一样的，比如“woood"和“wood"是一样的，它们去重后都是“wod"

请你写一个程序帮忙在谜底库中找到正确的谜底。谜面是多个单词，都需要找到对应的谜底, 如果找不到的话，返回"not found"

输入描述

1. 谜面单词列表，以",”分隔
2. 谜底库单词列表，以"，”分隔

输出描述

* 匹配到的正确单词列表，以分隔
* 如果找不到，返回"notfound"

备注

1. 单词的数量N的范围：0 < N < 1000
2. 词汇表的数量M的范围：0 < M < 1000
3. 单词的长度P的范围：0<P<20
4. 输入的字符只有小写英文字母，没有其他字符

用例

|  |  |
| --- | --- |
| 输入 | conection  connection, today |
| 输岀 | connection |
| 说明 | 无 |
| 输入 | bdni, wooood  bind, wrong, wood |
| 输岀 | bind, wood |
|  |  |

32异常的打卡记录

题目描述

考勤记录是分析和考核职工工作时间利用情况的原始依据，也是计算职工工资的原始依据， 为了正确地计算职工工资和监督工资基金使用情况，公司决定对员工的手机打卡记录进行异 常排查。

如果出现以下两种情况，则认为打卡异常：

1. 实际设备号与注册设备号不一样
2. 或者，同一个员工的两个打卡记录的时间小于60分钟并且打卡距离超过5km。

给定打卡记录的[字符串数组](https://so.csdn.net/so/search?q=%E5%AD%97%E7%AC%A6%E4%B8%B2%E6%95%B0%E7%BB%84&spm=1001.2101.3001.7020)clockRecords (每个打卡记录组成为：工号;时间(分钟)；打卡 距离(km);实际设备号;注册设备号)，返回其中异常的打卡记录(按输入顺序输出)。

输入描述

第一行输入为N，表示打卡记录数；

之后的N行为打卡记录，每一行为一条打卡记录。

输出描述

输岀异常的打卡记录。

备注

* clockRecords 长度 < 1000
* clockRecords[i] 格 式 ：

(id),(time),(distance),(actualDeviceNumber),(registeredDeviceNumber)

id由6位数字组成

|  |  |
| --- | --- |
| • | time由整数组成，范围为0~1000 |
| • | distance由整数组成，范围为0~100 |
| • | actualDeviceNumber 与 registeredDeviceNumber 由思维大写字母组成 |

用例

|  |  |
| --- | --- |
| 输入 | 2  100000, 10, 1, ABCD, ABCD  100000, 50, 10, ABCD, ABCD |
| 输岀 | 100000, 10, 1, ABCD, ABCD; 100000, 50, 10, ABCD, ABCD |
| 说明 | 第一条记录是异常得，因为第二题记录与它得间隔不超过60分钟，但是打卡距离超过 了 5km,同理第二条记录也是异常得。 |
| 输入 | 2  100000,10,1,ABCD,ABCD  100001, 80, 10, ABCE, ABCE |
| 输岀 | null |
| 说明 | 无异常打卡记录，所以返回null |
| 输入 | 2  100000,10,1,ABCD,ABCD  100000,80,10,ABCE,ABCD |
| 输岀 | 100000,80,10,ABCE,ABCD |
| 说明 | 第二条记录得注册设备号与打卡设备号不一致，所以是异常记录 |

33优秀学员统计

题目描述

公司某部门软件教导团正在组织新员工每日打卡学习活动，他们开展这项学习活动已经一个 月了，所以想统计下这个月优秀的打卡员工。每个员工会对应一个id，每天的打卡记录记录 当天打卡员工的id集合，一共30天。

请你实现代码帮助统计出打卡次数top5的员工。加入打卡次数相同，将较早参与打卡的员 工排在前面，如果开始参与打卡的时间还是一样，将id较小的员工排在前面。

注：不考虑并列的情况，按规则返回前5名员工的id即可，如果当月打卡的员工少于5个, 按规则排序返回所有有打卡记录的员工id。

输入描述

第一行输入为新员工数量N，表示新员工编号id为0到N-1, N的范围为[1,100]

第二行输入为30个整数，表示每天打卡的员工数量，每天至少有1名员工打卡。

之后30行为每天打卡的员工id集合，id不会重复。

输出描述

按顺序输出打卡top5员工的id，用空格隔开。

用例

11

441111111111111111111111111122 输入

0 1 7 10

|  |  |
| --- | --- |
|  | 10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10   1. 10 2. 10 |
| 输岀 | 10 0 1 7 6 |

|  |  |
| --- | --- |
| 说明 | 员工编号范围为0~10, id为10的员工连续打卡30天，排第一，id为0,1,6, 7的员工 打卡都是两天，id为0,1, 7的员工在第一天就打卡，比id为6的员工早，排在前面，  0, 1, 7按id升序排列，所以输出[10, 0, 1, 7, 6] |
| 输入 | 7  66666666666666666666666666666666  0 1 2 3 4 5  0 1 2 3 4 5  0 1 2 3 4 5  0 1 2 3 4 5  0 1 2 3 4 5  012345  012345  012345  012345  012345  012345  012345  012345  012345  012345  012345  012345  012345  012345  012345  012345  012345  012345  012345  012345 |

|  |  |
| --- | --- |
|  | 0 1 2 3 4 5  0 1 2 3 4 5  0 1 2 3 4 5  0 1 2 3 4 5  0 1 2 3 4 5 |
| 输岀 | 01234 |
| 说明 | 员工编号范围为0-6, id为0,1, 2, 3, 4, 5的员工打卡次数相同，最早开始打卡的时 间也一样，所以按id升序返回前5个id |
| 输入 | 2  111111111111111111111111111122  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 |

|  |  |
| --- | --- |
|  | 1  1  1  1  1  1  1  1  1  1  0 1  0 1 |
| 输岀 | 10 |
| 说明 | 只有两名员工参与打卡，按规则排序输岀两名员工的id |

34静态扫描

题目描述

静态扫描可以快速识别源代码的缺陷，静态扫描的结果以扫描报告作为输岀：

1、 文件扫描的成本和文件大小相关，如果文件大小为N,则扫描成本为N个[金币](https://so.csdn.net/so/search?q=%E9%87%91%E5%B8%81&spm=1001.2101.3001.7020)

2、 扫描报告的缓存成本和文件大小无关，每缓存一个报告需要M个金币

3、 扫描报告缓存后，后继再碰到该文件则不需要扫描成本，直接获取缓存结果

给出源代码文件标识序列和文件大小序列，求解采用合理的缓存策略，最少需要的金币数。 **输入描述**

第一行为缓存一个报告金币数M，L<= M <= 100

第二行为文件标识序列：F1,F2,F3,....,Fn。

第三行为文件大小序列：S1,S2,S3,....,Sn。

备注：

. 1 <= N <= 10000

. 1 <= Fi <= 1000

. 1 <= Si <= 10

输出描述

采用合理的缓存策略，需要的最少金币数

|  |  |
| --- | --- |
| 输入 | 5  1 2 2 1 2 3 4  1111111 |
| 输岀 | 7 |
| 说明 | 文件大小相同，扫描成本均为1个金币。缓存任意文件均不合算，因而最少成本为7金 币。 |
| 输入 | 5  222225222  333331333 |
| 输岀 | 9 |
| 说明 | 无 |

35机房布局

题目描述

小明正在规划一个大型[数据中心](https://so.csdn.net/so/search?q=%E6%95%B0%E6%8D%AE%E4%B8%AD%E5%BF%83&spm=1001.2101.3001.7020)机房，为了使得机柜上的机器都能正常满负荷工作，需要确 保在每个机柜边上至少要有一个电箱。

为了简化题目，假设这个机房是一整排，M表示机柜，I表示间隔，请你返回这整排机柜， 至少需要多少个电箱。如果无解请返回-1。

输入描述

无

输出描述



**36 人数最多的站点**

公园园区提供小火车单向通行，从园区站点编号最小到最大通行如1~2~3~4~1,然后供员 工在各个办公园区穿梭，通过对公司N个员工调研统计到每个员工的坐车区间，包含前后 站点，请设计一个程序计算出小火车在哪个园区站点时人数最多。

输入描述

第1个行，为调研员工人数

第2行开始，为每个员工的上车站点和下车站点。

使用数字代替每个园区用空格分割，如35表示从第3个园区上车，在第5个园区下车

输出描述

人数最多时的园区站点编号，最多人数相同时返回编号最小的园区站点

用例

3

1. 3

输入

1. 4

1 4

输岀 2

说明

37 快递投放问题

题目描述

有N个快递站点用字符串标识，某些站点之间有道路连接。

每个站点有一些包裹要运输，每个站点间的包裹不重复，路上有检查站会导致部分货物无法 通行，计算哪些货物无法正常投递？

输入描述

* 第一行输入M N，M个包裹N个道路信息.

. 0<=M,N<=100，

・检查站禁止通行的包裹如果有多个以空格分开

输出描述

* 输出不能送达的包裹，如：package2 package4，
* 如果所有包裹都可以送达则输出：none，

・输岀结果按照升序排列。

用例

4 2

packagel A C

package2 A C

输入 package3 B C

package4 A C

A B packagel

A C package2

输出 package2

说明 无

1. 区块链文件转储系统

题目描述

[区块链](https://so.csdn.net/so/search?q=%E5%8C%BA%E5%9D%97%E9%93%BE&spm=1001.2101.3001.7020)底层存储是一个链式文件系统，由顺序的N个文件组成，每个文件的大小不一，依 次为F1,F2,...,Fn。随着时间的推移，所占存储会越来越大。

[云平台](https://so.csdn.net/so/search?q=%E4%BA%91%E5%B9%B3%E5%8F%B0&spm=1001.2101.3001.7020)考虑将区块链按文件转储到廉价的SATA盘，只有连续的区块链文件才能转储到

SATA盘上，且转储的文件之和不能超过SATA盘的容量。

假设每块[SATA](https://so.csdn.net/so/search?q=SATA&spm=1001.2101.3001.7020)盘容量为M，求能转储的最大连续文件之和。

输入描述

第一行为 SATA 盘容量 M，1000 < M < 1000000

第二行为区块链文件大小序列F1,F2,...,Fn。其中1<n< 100000, 1 < Fi < 500

输出描述

求能转储的最大连续文件大小之和

用例

1000

|  |  |
| --- | --- |
|  | 1000 |
| 输入 | 100 300 500 400 400 150 100 |
| 输岀 | 950 |
| 说明 | 最大序列和为950，序列为[400, 400, 150] |

输入

100 500 400 150 500 100

输出 1000

说明 最大序列和为1000,序列为[100, 500, 400]

39 识图谱新词挖掘

题目描述

小华负责公司[知识图谱](https://so.csdn.net/so/search?q=%E7%9F%A5%E8%AF%86%E5%9B%BE%E8%B0%B1&spm=1001.2101.3001.7020)产品，现在要通过新词挖掘完善知识图谱。

新词挖掘：给出一个待挖掘问题内容字符串Content和一个词的字符串word，找到content

中所有word的新词。

新词：使用词word的字符排列形成的字符串。

请帮小华实现新词挖掘，返回发现的新词的数量。

输入描述

第一行输入为待挖掘的文本内容content；

第二行输入为词word；

输出描述

在content中找到的所有word的新词的数量。

备注

* 0 < content 的长度 < 10000000
* 1< word的长度< 2000

用例



40新员工座位

题目描述

工位由序列F1,F2...Fn组成，Fi值为0、1或2。其中0代表空置，1代表有人，2代表障

碍物。

1、 某一空位的友好度为左右连续老员工数之和，

2、 为方便新员工学习求助，优先安排友好度高的空位，

给出工位序列，求所有空位中**友好度的最大值**

输入描述

第一行为工位序列：F1, F2...Fn组成，

|  |  |
| --- | --- |
| 1<=n<=10000, | Fi值为0、1或2。其中0代表空置，1代表有人，2代表障碍物。 |

输出描述

所有空位中友好度的最大值。如果没有空位，返回0。

用例

|  |  |
| --- | --- |
| 输入 | 0 1 0 |
| 输岀 | 1 |
| 说明 | 第1个位置和第3个位置，友好度均为1。 |

**41.最长广播响应**

题目描述

某通信网络中有N个网络结点，用1到N进行标识。

网络中的结点互联互通，且结点之间的消息传递有时延，相连结点的时延均为一个时间单位。

现给定网络结点的连接关系link[i]={u，v}，其中u和v表示网络结点。

当指定一个结点向其他结点进行广播，所有被广播结点收到消息后都会在原路径上回复一条响应消息，请计算发送结点至少需要等待几个时间单位才能收到所有被广播结点的响应消息。

注：

N的取值范围为[1，100];

连接关系link的长度不超过3000，且1 <= u,v <= N;

网络中任意结点间均是可达的;

输入描述

输入的第一行为两个正整数，分别表示网络结点的个数N，以及时延列表的长度T；

接下来的T行输入，表示结点间的连接关系列表；

最后一行的输入为一个正整数，表示指定的广播结点序号；

输出描述

输出一个整数，表示发送结点接收到所有响应消息至少需要等待的时长。

用例

输入 5 7

1 4

2 1

2 3

2 4

3 4

3 5

4 5

2

输出 4

说明 结点2到5的最小时延为2，到剩余结点的最小时延均为1，所以至少要等待2\*2=4s。

# 42.考古学家

题目描述

有一个考古学家发现一个石碑，但是很可惜，发现时其已经断成多段，原地发现n个断口整齐的石碑碎片。为了破解石碑内容，考古学家希望有程序能帮忙计算复原后的石碑文字组合数，你能帮忙吗？

输入描述

第一行输入n，n表示石碑碎片的个数。

第二行依次输入石碑碎片上的文字内容s，共有n组。

输出描述

输出石碑文字的组合（按照升序排列），行末无多余空格。

用例

输入 3

a b c

输出 abc

acb

bac

bca

cab

cba

说明 无

输入 3

a b a

输出 aab

aba

baa

说明 无

# 43.贪吃蛇

题目描述

贪吃蛇是一个经典游戏，蛇的身体由若干方格连接而成，身体随蛇头移动。蛇头触碰到食物时，蛇的长度会增加一格。蛇头和身体的任一方格或者游戏版图边界碰撞时，游戏结束。

下面让我们来完成贪吃蛇游戏的模拟。

给定一个N\*M的数组arr，代表N\*M个方格组成的版图，贪吃蛇每次移动一个方格。

若arr[i][j] == ‘H’，表示该方格为贪吃蛇的起始位置；

若arr[i][j] == ‘F’，表示该方格为食物，

若arr[i][j] == ‘E’，表示该方格为空格。

贪吃蛇初始长度为1，初始移动方向为向左。

为给定一系列贪吃蛇的移动操作，返回操作后蛇的长度，如果在操作执行完之前已经游戏结束，返回游戏结束时蛇的长度。

贪吃蛇移动、吃食物和碰撞处理的细节见下面图示：

图1：截取了贪吃蛇移动的一个中间状态，H表示蛇头，F表示食物，数字为蛇身体各节的编号，蛇为向左移动，此时蛇头和食物已经相邻

图2：蛇头向左移动一格，蛇头和食物重叠，注意此时食物的格子成为了新的蛇头，第1节身体移动到蛇头位置，第2节身体移动到第1节身体位置，以此类推，最后添加第4节身体到原来第3节身体的位置。

图3：蛇头继续向左移动一格，身体的各节按上述规则移动，此时蛇头已经和边界相邻，但还未碰撞。

图4：蛇头继续向左移动一格，此时蛇头已经超过边界，发生碰撞，游戏结束。

图5和图6给出一个蛇头和身体碰撞的例子，蛇为向上移动。

图5时蛇头和第7节身体相邻，但还未碰撞；

图6蛇头向上移动一格，此时蛇头和第8节身体都移动到了原来第7节身体的位置，发生碰撞，游戏结束。

输入描述

输入第一行为空格分隔的字母，代表贪吃蛇的移动操作。

字母取值为U、D、L、R和G，

U、D、L、R分别表示贪吃蛇往上、下、左、右和转向，转向时贪吃蛇不移动 ，G表示贪吃蛇按当前的方向移动一格。

用例保证输入的操作正确。

第二行为空格分隔的两个数，指定N和M，为数组的行和列数。

余下N行每行是空格分隔的M个字母。字母取值为H、F和E，H表示贪吃蛇的起始位置，F表示食物，E表示该方格为空。

用例保证有且只有一个H，而F和E会有多个。

输出描述

输出一个数字，为蛇的长度。

用例

输入 D G G

3 3

F F F

F F H

E F E

输出 1

说明

地图表示为：

蛇头 H(Head)

食物 F(Food)

E表示该方格为空

四个方向分别表示为：

向上 U(up)

向下 D(down)

向左 L(Left)

向右 R(Right)

# 44.最大社交距离

题目描述

疫情期间需要大家保证一定的社交距离，公司组织开交流会议。座位一排共 N 个座位，编号分别为[0,N-1],

要求员工一个接着一个进入会议室，并且可以在任何时候离开会议室。

满足：

每当一个员工进入时，需要坐到最大社交距离（最大化自己和其他人的距离的座位）；

如果有多个这样的座位，则坐到索引最小的那个座位。

输入描述

会议室座位总数seatNum。(1 <= seatNum <= 500)

员工的进出顺序 seatOrLeave 数组，元素值为 1，表示进场；元素值为负数，表示出场（特殊：位置 0 的员工不会离开）。

例如 -4 表示坐在位置 4 的员工离开（保证有员工坐在该座位上）

输出描述

最后进来员工，他会坐在第几个位置，如果位置已满，则输出-1。

用例

输入 10

[1,1,1,1,-4,1]

输出 5

说明

seat -> 0,空在任何位置都行，但是要给他安排索引最小的位置，也就是座位 0

seat -> 9,要和旁边的人距离最远，也就是座位 9

seat -> 4,要和旁边的人距离最远，应该坐到中间，也就是座位 4

seat -> 2,员工最后坐在 2 号座位上

leave[4], 4 号座位的员工离开

seat -> 5,员工最后坐在 5 号座位上

# 45.污染水域

题目描述

输入一行字符串，字符串可转换为N\*N的数组，数组可认为是一个水域，判断多少天后，水域被全部污染。

数组中只有0和1，0表示纯净，1表示污染，每天只可污染上下左右的水域，如果开始全部被污染，或永远无法污染，则返回-1。

输入描述

无

输出描述

无

用例

输入 1,0,1,0,0,0,1,0,1

输出 2

说明

输入转化为数组为：

1 0 1

0 0 0

1 0 1

第一天后水域变为

1 1 1

1 0 1

1 1 1

第二天全部被污染

输入 0,0,0,0

输出 -1

说明 无

# 46.连续出牌数量

题目描述

有这么一款单人卡牌游戏，牌面由颜色和数字组成，颜色为红、黄、蓝、绿中的一种，数字为0-9中的一个。游戏开始时玩家从手牌中选取一张卡牌打出，接下来如果玩家手中有和他上一次打出的手牌颜色或者数字相同的手牌，他可以继续将该手牌打出，直至手牌打光或者没有符合条件可以继续打出的手牌。

现给定一副手牌，请找到最优的出牌策略，使打出的手牌最多。

输入描述

输入为两行，第一行是每张手牌的数字，数字由空格分隔，第二张为对应的每张手牌的颜色，用r y b g这4个字母分别代表4种颜色，字母也由空格分隔。手牌数量不超过10。

输出描述

输出一个数字，即最多能打出的手牌的数量。

用例

输入 1 4 3 4 5

r y b b r

输出 3

说明

如果打（1, r）-> (5, r)，那么能打两张。

如果打（4，y) -> (4, b) -> (3, b)，那么能打三张。

# 47.简易内存池

题目描述

请实现一个简易内存池,根据请求命令完成内存分配和释放。

内存池支持两种操作命令，REQUEST和RELEASE，其格式为：

REQUEST=请求的内存大小 表示请求分配指定大小内存，如果分配成功，返回分配到的内存首地址；如果内存不足，或指定的大小为0，则输出error。

RELEASE=释放的内存首地址 表示释放掉之前分配的内存，释放成功无需输出，如果释放不存在的首地址则输出error。

注意：

内存池总大小为100字节。

内存池地址分配必须是连续内存，并优先从低地址分配。

内存释放后可被再次分配，已释放的内存在空闲时不能被二次释放。

不会释放已申请的内存块的中间地址。

释放操作只是针对首地址所对应的单个内存块进行操作，不会影响其它内存块。

输入描述

首行为整数 N , 表示操作命令的个数，取值范围：0 < N <= 100。

接下来的N行, 每行将给出一个操作命令，操作命令和参数之间用 “=”分割。

输出描述

请求分配指定大小内存时，如果分配成功，返回分配到的内存首地址；如果内存不足，或指定的大小为0，则输出error

释放掉之前分配的内存时，释放成功无需输出，如果释放不存在的首地址则输出error。

用例

输入 2

REQUEST=10

REQUEST=20

输出 0

10

说明 无

输入 5

REQUEST=10

REQUEST=20

RELEASE=0

REQUEST=20

REQUEST=10

输出 0

10

30

0

说明

第一条指令，申请地址0~9的10个字节内存，返回首地址0

第二条指令，申请地址10~29的20字节内存，返回首地址10

第三条指令，释放首地址为0的内存申请，0~9地址内存被释放，变为空闲，释放成功，无需输出

第四条指令，申请20字节内存，09地址内存连续空间不足20字节，往后查找到3049地址，返回首地址30

第五条指令，申请10字节，0~9地址内存空间足够，返回首地址0

# 48.单词搜索

题目描述

找到它是一个小游戏，你需要在一个矩阵中找到给定的单词。

假设给定单词 HELLOWORD，在矩阵中只要能找到 H->E->L->L->O->W->O->R->L->D连成的单词，就算通过。

注意区分英文字母大小写，并且您只能上下左右行走，不能走回头路。

输入描述

输入第 1 行包含两个整数 n、m (0 < n,m < 21) 分别表示 n 行 m 列的矩阵，

第 2 行是长度不超过100的单词 W (在整个矩阵中给定单词 W 只会出现一次)，

从第 3 行到第 n+2 行是指包含大小写英文字母的长度为 m 的字符串矩阵。

输出描述

如果能在矩阵中连成给定的单词，则输出给定单词首字母在矩阵中的位置(第几行 第几列)，

否则输出“NO”。

用例

输入 5 5

HELLOWORLD

CPUCY

EKLQH

CHELL

LROWO

DGRBC

输出 3 2

说明 无

输入 5 5

HELLOWORLD

CPUCY

EKLQH

CHELL

LROWO

AGRBC

输出 NO

说明 无

# 49最长的顺子

题目描述

斗地主起源于湖北十堰房县，据说是一位叫吴修全的年轻人根据当地流行的扑克玩法“跑得快”改编的，如今已风靡整个中国，并流行于互联网上。

牌型：单顺，又称顺子，最少5张牌，最多12张牌(3…A)不能有2，也不能有大小王，不计花色。

例如： 3-4-5-6-7-8，7-8-9-10-J-Q，3-4-5-6-7-8-9-10-J-Q-K-A

可用的牌 3<4<5<6<7<8<9<10<J<Q<K<A<2<B(小王)<C(大王)，每种牌除大小王外有四种花色

(共有13×4+2张牌)

输入：

手上有的牌

已经出过的牌(包括对手出的和自己出的牌)

输出：

对手可能构成的最长的顺子(如果有相同长度的顺子，输出牌面最大的那一个)，

如果无法构成顺子，则输出 NO-CHAIN。

输入描述

输入的第一行为当前手中的牌

输入的第二行为已经出过的牌

输出描述

最长的顺子

用例

输入 3-3-3-3-4-4-5-5-6-7-8-9-10-J-Q-K-A

4-5-6-7-8-8-8

输出 9-10-J-Q-K-A

说明 无

输入 3-3-3-3-8-8-8-8

K-K-K-K

输出 NO-CHAIN

说明 剩余的牌无法构成顺子

# 50.任务最优调度

题目描述

给定一个正整数数组表示待系统执行的任务列表，数组的每一个元素代表一个任务，元素的值表示该任务的类型。

请计算执行完所有任务所需的最短时间。

任务执行规则如下:

任务可以按任意顺序执行，且每个任务执行耗时间均为1个时间单位。

两个同类型的任务之间必须有长度为N个单位的冷却时间，比如N为2时，在时间K执行了类型3的任务，那么K+1和K+2两个时间不能执行类型3任务。

系统在任何一个单位时间内都可以执行一个任务，或者等待状态。

说明：数组最大长度为1000，速度最大值1000。

输入描述

第一行记录一个用半角逗号分隔的数组，数组长度不超过1000，数组元素的值不超过1000，

第二行记录任务冷却时间，N为正整数，N<=100。

输出描述

输出为执行完所有任务所需的最短时间。

用例

输入 2,2,2,3

2

输出 7

说明 时间1：执行类型2任务。

时间2：执行类型3的任务（因为冷却时间为2，所以时间2不能执行类型2的任务）。

时间3：系统等待（仍然在类型2的冷却时间）。

时间4：执行类型2任务。

时间5：系统等待。

时间6：系统等待。

时间7：执行类型2任务。

因此总共耗时7。

# 51.仿LISP运算

题目描述

LISP 语言唯一的语法就是括号要配对。

形如 (OP P1 P2 …)，括号内元素由单个空格分割。

其中第一个元素 OP 为操作符，后续元素均为其参数，参数个数取决于操作符类型。

注意：

参数 P1, P2 也有可能是另外一个嵌套的 (OP P1 P2 …) ，当前 OP 类型为 add / sub / mul / div（全小写），分别代表整数的加减乘除法，简单起见，所有 OP 参数个数均为 2 。

举例：

输入：(mul 3 -7)输出：-21

输入：(add 1 2) 输出：3

输入：(sub (mul 2 4) (div 9 3)) 输出 ：5

输入：(div 1 0) 输出：error

题目涉及数字均为整数，可能为负；

不考虑 32 位溢出翻转，计算过程中也不会发生 32 位溢出翻转，

除零错误时，输出 “error”，

除法遇除不尽，向下取整，即 3/2 = 1

输入描述

输入为长度不超过512的字符串，用例保证了无语法错误

输出描述

输出计算结果或者“error”

用例

输入

(div 12 (sub 45 45))

输出 error

说明 无

# 52.学生方阵

题目描述

学校组织活动，将学生排成一个矩形方阵。

请在矩形方阵中找到最大的位置相连的男生数量。

这个相连位置在一个直线上，方向可以是水平的，垂直的，成对角线的或者呈反对角线的。

注：学生个数不会超过10000

输入描述

输入的第一行为矩阵的行数和列数，接下来的n行为矩阵元素，元素间用”,”分隔。

输出描述

输出一个整数，表示矩阵中最长的位置相连的男生个数。

用例

输入 3,4

F,M,M,F

F,M,M,F

F,F,F,M

输出 3

说明

# 53.发广播

题目描述

某地有N个广播站，站点之间有些有连接，有些没有。有连接的站点在接受到广播后会互相发送。

给定一个N\*N的二维数组matrix,数组的元素都是字符’0’或者’1’。

matrix[i][j] = ‘1’, 代表i和j站点之间有连接，

matrix[i][j] = ‘0’, 代表没连接，

现在要发一条广播，问初始最少给几个广播站发送，才能保证所有的广播站都收到消息。

输入描述

从stdin输入，共一行数据，表示二维数组的各行，用逗号分隔行。保证每行字符串所含的字符数一样的。

比如：110,110,001。

输出描述

返回初始最少需要发送广播站个数

用例

输入 110,110,001

输出 2

说明 站点1和站点2直接有连接，站点3和其他的都没连接，所以开始至少需要给两个站点发送广播。

输入

100,010,001

输出 3

说明 3台服务器互不连接，所以需要分别广播这3台服务器。

输入

11,11

输出 1

说明 2台服务器相互连接，所以只需要广播其中一台服务器

# 54.字符串比较

题目描述

给定字符串A、B和正整数V，A的长度与B的长度相等， 请计算A中满足如下条件的最大连续子串的长度：

1、该连续子串在A和B中的位置和长度均相同。

2、该连续子串|A[i] – B[i]|之和小于等于V。其中|A[i] – B[i]|表示两个字母ASCII码之差的绝对值。

输入描述

输入为三行：

第一行为字符串A，仅包含小写字符，1 <= A.length <=1000。

第二行为字符串B，仅包含小写字符，1 <= B.length <=1000。

第三行为正整数V，0<= V <= 10000。

输出描述

字符串最大连续子串的长度，要求该子串|A[i] – B[i]|之和小于等于V。

用例

输入

xxcdefg

cdefghi

5

输出 2

说明 无

# 55.斗地主之顺子

题目描述

在斗地主扑克牌游戏中， 扑克牌由小到大的顺序为：3,4,5,6,7,8,9,10,J,Q,K,A,2，玩家可以出的扑克牌阵型有：单张、对子、顺子、飞机、炸弹等。

其中顺子的出牌规则为：由至少5张由小到大连续递增的扑克牌组成，且不能包含2。

例如：{3,4,5,6,7}、{3,4,5,6,7,8,9,10,J,Q,K,A}都是有效的顺子；而{J,Q,K,A,2}、 {2,3,4,5,6}、{3,4,5,6}、{3,4,5,6,8}等都不是顺子。

给定一个包含13张牌的数组，如果有满足出牌规则的顺子，请输出顺子。

如果存在多个顺子，请每行输出一个顺子，且需要按顺子的第一张牌的大小（必须从小到大）依次输出。

如果没有满足出牌规则的顺子，请输出No。

输入描述

13张任意顺序的扑克牌，每张扑克牌数字用空格隔开，每张扑克牌的数字都是合法的，并且不包括大小王：

2 9 J 2 3 4 K A 7 9 A 5 6

不需要考虑输入为异常字符的情况

输出描述

组成的顺子，每张扑克牌数字用空格隔开：

3 4 5 6 7

用例

输入 2 9 J 2 3 4 K A 7 9 A 5 6

输出 3 4 5 6 7

说明 13张牌中，可以组成的顺子只有1组：3 4 5 6 7。

输入 2 9 J 10 3 4 K A 7 Q A 5 6

输出

3 4 5 6 7

9 10 J Q K A

说明 13张牌中，可以组成2组顺子，从小到大分别为：3 4 5 6 7 和 9 10 J Q K A

输入 2 9 9 9 3 4 K A 10 Q A 5 6

输出 No

说明 13张牌中，无法组成顺子。

# 56.数据分类

题目描述

对一个数据a进行分类，分类方法为：

此数据a（四个字节大小）的四个字节相加对一个给定的值b取模，如果得到的结果小于一个给定的值c，则数据a为有效类型，其类型为取模的值；如果得到的结果大于或者等于c，则数据a为无效类型。

比如一个数据a=0x01010101，b=3，按照分类方法计算（0x01+0x01+0x01+0x01）%3=1，

所以如果c=2，则此a为有效类型，其类型为1，如果c=1，则此a为无效类型；

又比如一个数据a=0x01010103，b=3，按照分类方法计算（0x01+0x01+0x01+0x03）%3=0，

所以如果c=2，则此a为有效类型，其类型为0，如果c=0，则此a为无效类型。

输入12个数据，第一个数据为c，第二个数据为b，剩余10个数据为需要分类的数据，

请找到有效类型中包含数据最多的类型，并输出该类型含有多少个数据。

输入描述

输入12个数据，用空格分隔，第一个数据为c，第二个数据为b，剩余10个数据为需要分类的数据。

输出描述

输出最多数据的有效类型有多少个数据。

用例

输入 3 4 256 257 258 259 260 261 262 263 264 265

输出 3

说明

10个数据4个字节相加后的结果分别为1 2 3 4 5 6 7 8 9 10，

故对4取模的结果为1 2 3 0 1 2 3 0 1 2，c为3，所以0 1 2都是有效类型，类型为1和2的有3个数据，类型为0的只有2个数据，故输出3。

输入 1 4 256 257 258 259 260 261 262 263 264 265

输出 2

说明

10个数据4个字节相加后的结果分别为1 2 3 4 5 6 7 8 9 10，

故对4取模的结果为1 2 3 0 1 2 3 0 1 2，c为1，

所以只有0是有效类型，类型为0的有2个数据，故输出2。

# 57.高效的任务规划

题目描述

你有 n 台机器，编号为 1~n，每台都需要完成一项工作，机器经过配置后都能完成独立完成一项工作。

假设第 i 台机器你需要花 B 分钟进行设置，然后开始运行，J 分钟后完成任务。

现在，你需要选择布置工作的顺序，使得用最短的时间完成所有工作。

注意，不能同时对两台进行配置，但配置完成的机器们可以同时执行他们各自的工作。

输入描述

第一行输入代表总共有 M 组任务数据（1<M<=10）。

每组数第一行为一个整数指定机器的数量 N（0<N<=1000）。

随后的 N 行每行两个整数，第一个表示 B（0<=B<=10000），第二个表示 J（0<=J<=10000）。

每组数据连续输入，不会用空行分隔。各组任务单独计时。

输出描述

对于每组任务，输出最短完成时间，且每组的结果独占一行。例如，两组任务就应该有两行输出。

用例

输入 1

1

2 2

输出 4

说明

第一行1：为一组任务，

第二行1：代表只有一台机器，

第三行2 2：表示该机器配置需2分钟，执行需2分钟。

输入 2

2

1 1

2 2

3

1 1

2 2

3 3

输出 4

7

说明

第一行2：代表两组任务，

第二行2：代表第一组任务有2个机器，

第三行1 1：代表机器1配置需要1分，运行需要1分，

第四行2 2：代表机器2配置需要2分，运行需要2分，

第五行3：代表第二组任务需要3个机器，

第6-8行分别表示3个机器的配置与运行时间。

# 58.完全二叉树非叶子部分后序遍历

题目描述

给定一个以顺序储存结构存储整数值的完全二叉树序列（最多1000个整数），请找出此完全二叉树的所有非叶子节点部分，然后采用后序遍历方式将此部分树（不包含叶子）输出。

1、只有一个节点的树，此节点认定为根节点（非叶子）。

2、此完全二叉树并非满二叉树，可能存在倒数第二层出现叶子或者无右叶子的情况

其他说明：二叉树的后序遍历是基于根来说的，遍历顺序为：左-右-根

输入描述

一个通过空格分割的整数序列字符串

输出描述

非叶子部分树结构。备注：输出数字以空格分隔

用例

输入 1 2 3 4 5 6 7

输出 2 3 1

说明 找到非叶子部分树结构，然后采用后序遍历输出。

# 59.数组二叉树

题目描述

二叉树也可以用数组来存储，给定一个数组，树的根节点的值存储在下标1，对于存储在下标N的节点，它的左子节点和右子节点分别存储在下标2\*N和2\*N+1，并且我们用值-1代表一个节点为空。

给定一个数组存储的二叉树，试求从根节点到最小的叶子节点的路径，路径由节点的值组成。

输入描述

输入一行为数组的内容，数组的每个元素都是正整数，元素间用空格分隔。

注意第一个元素即为根节点的值，即数组的第N个元素对应下标N，下标0在树的表示中没有使用，所以我们省略了。

输入的树最多为7层。

输出描述

输出从根节点到最小叶子节点的路径上，各个节点的值，由空格分隔，用例保证最小叶子节点只有一个。

用例

输入 3 5 7 -1 -1 2 4

输出 3 7 2

说明 最小叶子节点的路径为3 7 2。

输入 5 9 8 -1 -1 7 -1 -1 -1 -1 -1 6

输出 5 8 7 6

说明 最小叶子节点的路径为5 8 7 6，注意数组仅存储至最后一个非空节点，故不包含节点“7”右子节点的-1。

# 60分积木

题目描述

Solo和koko是两兄弟，妈妈给了他们一大堆积木，每块积木上都有自己的重量。现在他们想要将这些积木分成两堆。哥哥Solo负责分配，弟弟koko要求两个人获得的积木总重量“相等”（根据Koko的逻辑），个数可以不同，不然就会哭，但koko只会先将两个数转成二进制再进行加法，而且总会忘记进位（每个进位都忘记）。如当25（11101）加11（01011）时，koko得到的计算结果是18（10010）：

11001

+01011

--------

10010

Solo想要尽可能使自己得到的积木总重量最大，且不让koko哭。

输入描述

第一行是一个整数N(2≤N≤100)，表示有多少块积木；

第二行为空格分开的N个整数Ci(1≤Ci≤106)，表示第i块积木的重量。

输出描述

如果能让koko不哭，输出Solo所能获得积木的最大总重量；否则输出“NO”。

用例

输入

3

3 5 6

输出 11

说明