

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：157603

本题知识点：[动态规划](#)

 [算法知识视频讲解](#)

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

王强今天很开心，公司发给N元的年终奖。王强决定把年终奖用于购物，他把想买的物品分为两类：主件与附件，附件是从属于某个主件的，下表就是一些主件与附件的例子：

主件	附件
电脑	打印机，扫描仪
书柜	图书
书桌	台灯，文具
工作椅	无

如果要买归类为附件的物品，必须先买该附件所属的主件。每个主件可以有 0 个、1 个或 2 个附件。附件不再有从属于自己的附件。王强想买的东西很多，为了不超出预算，他把每件物品规定了一个重要度，分为 5 等：用整数 1 ~ 5 表示，第 5 等最重要。他还从因特网上查到了每件物品的价格（都是 10 元的整数倍）。他希望在不超过 N 元（可以等于 N 元）的前提下，使每件物品的价格与重要度的乘积的总和最大。

设第 j 件物品的价格为 $v[j]$ ，重要度为 $w[j]$ ，共选中了 k 件物品，编号依次为 j_1, j_2, \dots, j_k ，则所求的总和为： $v[j_1]*w[j_1]+v[j_2]*w[j_2]+\dots+v[j_k]*w[j_k]$ 。（其中 * 为乘号）

请你帮助王强设计一个满足要求的购物单。

输入描述:

输入的第 1 行，为两个正整数，用一个空格隔开：N m
（其中 N（ < 32000 ）表示总钱数，m（ < 60 ）为希望购买物品的个数。）

从第 2 行到第 m+1 行，第 j 行给出了编号为 j-1 的物品的基本数据，每行有 3 个非负整数 v p q

（其中 v 表示该物品的价格（ $v < 10000$ ），p 表示该物品的重要度（ $1 \sim 5$ ），q 表示该物品是主件还是附件。如果 q=0，表示该物品为主件，如果 q>0，表示该物品为附件，q 是所属主件的编号）

输出描述:

输出文件只有一个正整数，为不超过总钱数的物品的价格与重要度乘积的总和的最大值（ < 200000 ）。

示例1

输入

[复制](#)

```
1000 5
800 2 0
400 5 1
300 5 1
400 3 0
500 2 0
```

输出

[复制](#)

```
2200
```

通过的代码

收起

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	<div>没有昵称的用户</div> <div>向日葵201...</div> <div>关注</div>	18天前	22	7512KB	Javascript_V8	收起代码
<pre>1 /*let n; 2 while(n = readline()){ 3 let N = n.split(' ')[0]/10, len=n.split(' ')[1]; 4 let v=[],p=[],a=[],u=[]; 5 while(len>0){ 6 let d = readline().split(' ').map(Number); 7 v.push(d[0]); 8 p.push(d[1]); 9 a.push(d[2]); 10 len--; 11 } 12 //console.log(len,N,v,p,a,u) 13 v = v.map((i,index,arr) => { 14 i = i/10; 15 if(a[index]){ 16 return arr[a[index]-1]+i; 17 } 18 return i; 19 }); 20 u = v.map((i,index,arr) => { 21 if(a[index]){ 22 return (i-arr[a[index]-1])*p[index] + arr[a[index]-1]*p[a[index]-1]; 23 } 24 return i*p[index]; 25 }); 26 console.log(getMaxUtilities(N,v,u)); 27 } 28 29 function getMaxUtilities(N,v,u){ 30 let resultTable = new Array((N+1).fill(0)); 31 for(let m=1;m<=u.length;m++){ 32 for(let j=N;j>=1;j--){ 33 if(j>=v[m-1]){ 34 resultTable[j] = Math.max(resultTable[j],resultTable[j-v[m-1]]+u[m-1]); 35 } 36 } 37 }; 38 return resultTable[N]*10; 39 }*/ 40 41 var arr=readline().split(' ').map(val=>parseInt(val)); 42 var c=arr[0]/10, 43 k=arr[1]; 44 var goods=[], 45 count=1; 46 while(count<=k){ 47 var item=readline().split(' ').map(val=>parseInt(val)); 48 if(item[2]===0){ 49 goods[count]=[{v:item[0]/10,w:item[0]/10*item[1]}]; 50 }else{ 51 var add=goods[item[2]].map(val=>{ 52 return { 53 v:item[0]/10+val.v, 54 w:item[0]/10*item[1]+val.w 55 }; 56 }); 57 goods[item[2]]=...goods[item[2]],...add; 58 } 59 count++; 60 } 61 console.log(choose(goods)); 62 //console.log(goods[1][2].v); 63 64 function choose(goods){ 65 var res=Array(c+1).fill(0); 66 for(var i=1;i<goods.length;i++){ 67 for(var j=c;j>=0;j--){ 68 if(goods[i]){ 69 goods[i].forEach(val=>{ 70 if(val.v<=j){ 71 res[j]=Math.max(res[j],res[j-val.v]+val.w); 72 } 73 }) 74 } 75 } 76 } 77 return res[c]*10; 78 }</pre>						

复制代码



2



关注

63天前

22

7516KB

Javascript_V8

查看代码

题解 讨论 通过的代码

笔记 纠错 收藏

华为机试 > 坐标移动

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：302918

本题知识点：字符串

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

开发一个坐标计算工具，A表示向左移动，D表示向右移动，W表示向上移动，S表示向下移动。从（0,0）点开始移动，从输入字符串里面读取一些坐标，并将最终输入结果输出到输出文件里面。
输入：
合法坐标为A(或者D或者W或者S) + 数字（两位以内）
坐标之间以;分隔。
非法坐标点需要进行丢弃。如AA10; A1A; \$\$\$; YAD; 等。
下面是一个简单的例子 如：
A10;S20;W10;D30;X;A1A;B10A11;;A10;
处理过程：
起点 (0,0)
+ A10 = (-10,0)
+ S20 = (-10,-20)
+ W10 = (-10,-10)
+ D30 = (20,-10)
+ X = 无效
+ A1A = 无效
+ B10A11 = 无效
+ 一个空 不影响
+ A10 = (10,-10)
结果 (10, -10)

注意请处理多组输入输出

输入描述:

一行字符串

输出描述:

最终坐标，以;分隔

示例1

输入

A10;S20;W10;D30;X;A1A;B10A11;;A10;

输出

10,-10

题解 讨论 通过的代码

笔记 纠错 收藏

通过的代码

收起 v

排行	用户	提交时间	时间(ms)	内存	使用语言 ▾
1	 牛客38806... 关注	5天前	8	4984KB	Javascript_V 8
收起代码					
<pre>1 while (zxw=readline()){ 2 var zxw1=zxw.split(";"); 3 var a=0; 4 var b=0; 5 for (var i=0;i<zxw1.length;i++){ 6 if (zxw1[i][0]=="A"){ 7 a=a+parseInt(zxw1[i].substr(1,2)); 8 }; 9 if (zxw1[i][0]=="D"){ 10 a=a+parseInt(zxw1[i].substr(1,2)); 11 }; 12 if (zxw1[i][0]=="S"){ 13 b=b+parseInt(zxw1[i].substr(1,2)); 14 }; 15 if (zxw1[i][0]=="W"){ 16 b=b+parseInt(zxw1[i].substr(1,2)); 17 }; 18 } 19 console.log(a+","+b) 20 } 21 22</pre>					

华为机试 > 识别有效的IP地址和掩码并进行...

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：123076

本题知识点：字符串 查找

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

请解析IP地址和对应的掩码，进行分类识别。要求按照A/B/C/D/E类地址归类，不合法的地址和掩码单独归类。
所有的IP地址划分为 A,B,C,D,E五类
A类地址1.0.0.0~126.255.255.255;
B类地址128.0.0.0~191.255.255.255;
C类地址192.0.0.0~223.255.255.255;
D类地址224.0.0.0~239.255.255.255;
E类地址240.0.0.0~255.255.255.255

私网IP范围是：
10.0.0.0 ~ 10.255.255.255
172.16.0.0 ~ 172.31.255.255
192.168.0.0 ~ 192.168.255.255

子网掩码为二进制下前面是连续的1，然后全是0。（例如：255.255.255.32就是一个非法的掩码）
注意二进制下全是1或者全是0均为非法

注意：
1. 类似于【0.*.*.*】和【127.*.*.*】的IP地址不属于上述输入的任何一类，也不属于不合法ip地址，计数时可以忽略
2. 私有IP地址和A,B,C,D,E类地址是不冲突的

输入描述:

多行字符串。每行一个IP地址和掩码，用~隔开。

输出描述:

统计A、B、C、D、E、错误IP地址或错误掩码、私有IP的个数，之间以空格隔开。

示例1

输入

10.70.44.68~255.254.255.0
1.0.0.1~255.0.0.0
192.168.0.2~255.255.255.0
19..0..~255.255.255.0

输出

1 0 1 0 0 2 1

题解 讨论 通过的代码

笔记 纠错 收藏

通过的代码

收起

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 晓风20190... 关注	37天前	10	5556KB	Javascript_V 8	收起代码
<pre>1 let line 2 let A = 0 3 let B = 0 4 let C = 0 5 let D = 0 6 let E = 0 7 let errIp = 0 8 let privateIp = 0 9 while(line = readline()){ 10 let tempArr = line.split('~') 11 let ip = tempArr[0] 12 let yanma = tempArr[1] 13 deal(ip, yanma) 14 } 15 console.log(A+' '+B+' '+C+' '+D+' '+E+' '+errIp+' '+privateIp) 16 function deal(ip, yanma){ 17 let ipArr = ip.split('.') 18 let yanmaArr = yanma.split('.') 19 let ma2 = '' 20 // for(let ele in yanmaArr){ 21 // yanmaArr[ele] = Number(yanmaArr[ele]).toString(2); 22 // yanmaArr[ele].padStart(8) 23 // ma2+=yanmaArr[ele] 24 // } 25 for(var ele in yanmaArr){ 26 yanmaArr[ele] = Number(yanmaArr[ele]).toString(2); 27 if(yanmaArr[ele].length<8){ 28 for(let j=8-yanmaArr[ele].length;j>0;j--){ 29 yanmaArr[ele] = '0'+yanmaArr[ele]; 30 } 31 } 32 ma2 += yanmaArr[ele]; 33 } 34 for(let i=0;i<ipArr.length;i++){ 35 ipArr[i] = parseInt(ipArr[i]) 36 } 37 if(ipArr[0]>=1 && ipArr[0]<=126 && ipArr[1]>=0 && ipArr[1]<=255 && ipArr[2] >=0 && ipArr[3]<=255 && ipArr[3]>=0 && ipArr[3] 38 if((ma2.lastIndexOf('1')+1==ma2.indexOf('0'))&&(ma2.indexOf('0')!==-1)&&(ma2.lastIndexOf('1')!==-1)){ 39 A++; 40 if(ipArr[0] === 10){ 41 privateIp++; 42 } 43 }else{ 44 errIp++ 45 } 46 }else if(ipArr[0]>=128 && ipArr[0]<=191 && ipArr[1]>=0 && ipArr[1]<=255 && ipArr[2] >=0 && ipArr[3]<=255 && ipArr[3]>=0 && 47 if((ma2.lastIndexOf('1')+1==ma2.indexOf('0'))&&(ma2.indexOf('0')!==-1)&&(ma2.lastIndexOf('1')!==-1)){ 48 B++; 49 if(ipArr[0] === 172 && ipArr[2] >=16 && ipArr[2]<=31){ 50 privateIp++; 51 } 52 }else{ 53 errIp++ 54 } 55 }else if(ipArr[0]>=192 && ipArr[0]<=223 && ipArr[1]>=0 && ipArr[1]<=255 && ipArr[2] >=0 && ipArr[3]<=255 && ipArr[3]>=0 && 56 if((ma2.lastIndexOf('1')+1==ma2.indexOf('0'))&&(ma2.indexOf('0')!==-1)&&(ma2.lastIndexOf('1')!==-1)){ 57 C++; 58 if(ipArr[0] === 192 && ipArr[2] === 168){ 59 privateIp++; 60 } 61 }else{ 62 errIp++ 63 } 64 }else if(ipArr[0]>=224 && ipArr[0]<=239 && ipArr[1]>=0 && ipArr[1]<=255 && ipArr[2] >=0 && ipArr[3]<=255 && ipArr[3]>=0 && 65 if((ma2.lastIndexOf('1')+1==ma2.indexOf('0'))&&(ma2.indexOf('0')!==-1)&&(ma2.lastIndexOf('1')!==-1)){ 66 D++; 67 }else{ 68 errIp++ 69 } 70 }else if(ipArr[0]>=240 && ipArr[0]<=255 && ipArr[1]>=0 && ipArr[1]<=255 && ipArr[2] >=0 && ipArr[3]<=255 && ipArr[3]>=0 && 71 if((ma2.lastIndexOf('1')+1==ma2.indexOf('0'))&&(ma2.indexOf('0')!==-1)&&(ma2.lastIndexOf('1')!==-1)){ 72 E++; 73 }else{ 74 errIp++ 75 } 76 }else if(ipArr[0] === 0 ipArr[0]=== 127){ 77 if(!(ma2.lastIndexOf('1')+1==ma2.indexOf('0'))&&(ma2.indexOf('0')!==-1)&&(ma2.lastIndexOf('1')!==-1)){ 78 errIp++; 79 } 80 }else{ 81 errIp++ 82 } 83 }</pre>						

复制代码



2



梅梅20190...

关注

35天前

10

5624KB

Javascript_V
8

查看代码

华为机试 > 简单错误记录

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：202740

本题知识点：字符串

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

开发一个简单错误记录功能小模块，能够记录出错的代码所在的文件名称和行号。

处理：

1、记录最多8条错误记录，循环记录（或者说最后只输出最后出现的八条错误记录），对相同的错误记录（净文件名（保留最后16位）称和行号完全匹配）只记录一条，**错误计数增加**；

2、超过16个字符的文件名称，只记录文件的最后有效16个字符；

3、输入的文件可能带路径，记录文件名称不能带路径。

输入描述:

一行或多行字符串。每行包括带路径文件名称，行号，以空格隔开。

输出描述:

将所有的记录统计并将结果输出，格式：文件名 代码行数 数目，一个空格隔开，如：

示例1

输入


E:\V1R2\product\fpgadriver.c 1325

输出

fpgadriver.c 1325 1

题解 讨论 通过的代码

笔记 纠错 收藏

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 迷7 关注	47天前	7	5080KB	Javascript_V8	收起代码

```

1  var nameArr = []
2  var numArr = []
3  var initArr = []
4  while(str = readline()) {
5      initArr.push(str)
6  }
7
8  for(var j = 0; j< initArr.length; j++) {
9      let str2 = initArr[j]
10     let msgArr = str2.split(/\s{1,}/g)
11     let name = msgArr[0].substr(msgArr[0].lastIndexOf('\\') + 1)
12     name = name.length > 16 ? name.substr(name.length - 16) : name
13     let keysw = name + ' ' + msgArr[1]
14     if (!~nameArr.indexOf(keysw)) {
15         nameArr.push(keysw)
16         numArr.push(1)
17     } else {
18         numArr[nameArr.indexOf(keysw)] += 1
19     }
20 }
21 nameArr = nameArr.slice(nameArr.length - 8, nameArr.length)
22 numArr = numArr.slice(numArr.length - 8, numArr.length)
23 for(var i = 0, len = nameArr.length; i< len; i++) {
24     console.log(nameArr[i] + ' ' + numArr[i])
25 }

```

[复制代码](#)

华为机试 > 密码验证合格程序

相似的企业真题

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

密码要求:

1.长度超过8位

2.包括大小写字母,数字,其它符号,以上四种至少三种

3.不能有相同长度大于2的子串重复

输入描述:

一组或多组长度过2的子字符串。每组占一行

输出描述:

如果符合要求输出：OK，否则输出NG

示例1

输入

021Abc9000
021Abc9Abc1
021ABC9000
021\$bc9000

输出



OK
NG
NG
OK

题解 讨论 通过的代码

笔记 纠错 收藏

通过的代码

收起

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 DreamMak... 关注	36天前	7	5112KB	Javascript_V 8	收起代码
<pre>1 while(line = readline()){ 2 3 handleData(line) 4 } 5 6 function handleData(str){ 7 if(str.length<=8){ 8 console.log('NG') 9 return; 10 } 11 var point = 0 12 if(/[A-Z]/.test(str)){ 13 point++ 14 } 15 if(/[a-z]/.test(str)){ 16 point++ 17 } 18 if(/\d/.test(str)){ 19 point++ 20 } 21 if(/^[A-Za-z\d]/.test(str)){ 22 point++ 23 } 24 25 if(point<3){ 26 console.log('NG') 27 return; 28 } 29 30 for(var i=0;i<str.length-3;i++){ 31 var tempStr = str.substr(i,3); 32 if(str.lastIndexOf(tempStr)>i){ 33 console.log("NG") 34 return; 35 } 36 } 37 38 console.log("OK") 39 40 } 41 }</pre>						
2	 牛客74421... 关注	17天前	8	5088KB	Javascript_V 8	查看代码

题解

讨论

通过的代码

笔记

纠错

收藏

华为机试 > 简单密码

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：111125

本题知识点：字符串

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

密码是我们生活中非常重要的东东，我们的那么一点不能说的秘密就全靠它了。哇哈哈。接下来渊子要在密码之上再加一套密码，虽然简单但也安全。

假设渊子原来一个BBS上的密码为zvbo9441987,为了方便记忆，他通过一种算法把这个密码变换成YUANzhi1987，这个密码是他的名字和出生年份，怎么忘都忘不了，而且可以明目张胆地放在显眼的地方而不被别人知道真正的密码。

他是这么变换的，大家都知道手机上的字母：1--1, abc--2, def--3, ghi--4, jkl--5, mno--6, pqrs--7, tuv--8 wxyz--9, 0--0,就这么简单，渊子把密码中出现的小写字母都变成对应的数字，数字和其他的符号都不做变换，

声明：密码中没有空格，而密码中出现的大写字母则变成小写之后往后移一位，如：X,先变成小写，再往后移一位，不就是y了嘛，简单吧。记住，z往后移是a哦。

输入描述:

输入包括多个测试数据。输入是一个明文，密码长度不超过100个字符，输入直到文件结尾

输出描述:

输出渊子真正的密文

示例1

输入

YUANzhi1987

输出

zvbo9441987

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	<div><div>晴风沐语</div><div>关注</div></div>	12天前	7	4988KB	Javascript_V8	<div>收起代码</div>
<div><div>1</div><div>while(n = readline()) {</div><div>2</div><div> getPassword(n);</div><div>3</div><div>}</div><div>4</div><div>function getPassword(str) {</div><div>5</div><div> let newStr = '';</div><div>6</div><div> let str1 = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz';</div><div>7</div><div> let str2 = 'bcdefghijklmnopqrstuvwxyz0123456789';</div><div>8</div><div> for(let i = 0; i < str.length; i++) {</div><div>9</div><div> let s = str[i];</div><div>10</div><div> if((s >= 'a' && s <= 'z') (s >= 'A' && s <= 'Z')) {</div><div>11</div><div> s = str2[str1.indexOf(s)];</div><div>12</div><div> }</div><div>13</div><div> newStr += s;</div><div>14</div><div> }</div><div>15</div><div> console.log(newStr);</div><div>16</div><div>}</div></div> <div>复制代码</div>						

华为机试 > 汽水瓶

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：218492

本题知识点：数学 模拟

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

有这样一道智力题：“某商店规定：三个空汽水瓶可以换一瓶汽水。小张手上有十个空汽水瓶，她最多可以换多少瓶汽水喝？”答案是5瓶，方法如下：先用9个空瓶子换3瓶汽水，喝掉3瓶满的，喝完以后4个空瓶子，用3个再换一瓶，喝掉这瓶满的，这时候剩2个空瓶子。然后你让老板先借给你一瓶汽水，喝掉这瓶满的，喝完以后用3个空瓶子换一瓶满的还给老板。如果小张手上有n个空汽水瓶，最多可以换多少瓶汽水喝？

输入描述:

输入文件最多包含10组测试数据，每个数据占一行，仅包含一个正整数n（1<=n<=100），表示小张手上的空汽水瓶数。n=0表示输入结束，你的程序应当处理这一行。

输出描述:

对于每组测试数据，输出一行，表示最多可以喝的汽水瓶数。如果一瓶也喝不到，输出0。

示例1

输入

3
10
81
0

复制

输出

1
5
40

复制

通过的代码收起

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 牛客75139... 关注	6天前	7	4984KB	Javascript_V8	收起代码
<pre>1 while(line=readline()){ 2 const num = parseInt(line) 3 for(let i =0; i < num; i++){ 4 if((num+i)/3 < i){ 5 console.log(i-1) 6 break 7 } 8 } 9 }</pre>						
复制代码						

23

华为机试 > 删除字符串中出现次数最少的字符相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：179456
本题知识点：字符串
算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

实现删除字符串中出现次数最少的字符，若多个字符出现次数一样，则都删除。输出删除这些单词后的字符串，字符串中其它字符保持原来的顺序。
注意每个输入文件有多组输入，即多个字符串用回车隔开

输入描述:

字符串只包含小写英文字母，不考虑非法输入，输入的字符串长度小于等于20个字节。

输出描述:

删除字符串中出现次数最少的字符后的字符串。

示例1

输入	复制
abcedd	
输出	复制
dd	

排行	用户	提交时间	时间(ms)	内存	使用语言
----	----	------	--------	----	------

1	 c牛气冲天 关注	56天前	7	4984KB	JavaScript_V8 收起代码
---	---	------	---	--------	---------------------------------------

```

1 while(line = readline()) {
2   let arr = line.split('');
3   let obj = {};
4   for(let i = 0; i < arr.length; i++) {
5     if(obj[arr[i]]) {
6       obj[arr[i]] += 1;
7     } else {
8       obj[arr[i]] = 1;
9     }
10  }
11
12  let min = arr.length;
13  for(let j in obj) {
14    if(obj[j] < min) {
15      min = obj[j];
16    }
17  }
18
19  let res = '';
20  for(let j = 0; j < arr.length; j++) {
21    if(obj[arr[j]] != min) {
22      res += arr[j]
23    }
24  }
25  console.log(res);
26 }

```

[复制代码](#)

华为机试合唱队相似的企业真题

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

计算最少出列多少位同学，使得剩下的同学排成合唱队形
说明：
N位同学站成一排，音乐老师要请其中的(N-K)位同学出列，使得剩下的K位同学排成合唱队形。
合唱队形是指这样的一种队形：设K位同学从左到右依次编号为1，2...，K，他们的身高分别为T1，T2，...，TK，则他们的身高满足存在i（1<=i<=K）使得T1<T2<.....<Ti-1<Ti>Ti+1>.....>TK。
你的任务是，已知所有N位同学的身高，计算最少需要几位同学出列，可以使得剩下的同学排成合唱队形。

注意不允许改变队列元素的先后顺序
请注意处理多组输入输出！

输入描述:

整数N

输出描述:

最少需要几位同学出列

示例1

输入



8
186 186 150 200 160 130 197 200

输出

4

题解讨论通过的代码

笔记纠错收藏

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 牛客29352... 关注	24天前	14	6008KB	Javascript_V8	收起代码
<pre>1 function bs(arr, len, value) { 2 let lo = 0, hi = len - 1 3 while (lo <= hi) { 4 let mid = lo + ((hi - lo) >> 1) 5 if (arr[mid] === value) return mid 6 else if (arr[mid] > value) hi = mid - 1 7 else lo = mid + 1 8 } 9 return lo 10 } 11 12 function LIS(arr) { 13 let tmp = [arr[0]] 14 let res = [1] 15 let max = 1 16 let pos = 0 17 for (let i = 1; i < arr.length; i++) { 18 let arrLen = tmp.length 19 let val = arr[i] 20 if (val > tmp[max - 1]) { 21 tmp[max] = val 22 max++ 23 } else { 24 pos = bs(tmp, max, val) 25 tmp[pos] = val 26 } 27 res[i] = max 28 } 29 return res 30 } 31 while(num=readline()){ 32 var arrList = readline().split(' ').map(n=>parseInt(n)) 33 var dp1 = LIS(arrList) 34 var dp2 = LIS(arrList.reverse()) 35 dp2 = dp2.reverse() 36 var max = 0 37 for(var i=0;i<dp1.length;i++){ 38 max = Math.max(dp1[i]+dp2[i],max) 39 } 40 console.log(arrList.length-max+1) 41 }</pre>						
2	 夏日浅笑 关注	35天前	14	6088KB	Javascript_V8	查看代码

华为机试 > 数据分类处理

相似的企业真题

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

信息社会，有海量的数据需要分析处理，比如公安局分析身份证号码、QQ 用户、手机号码、银行帐号等信息及活动记录。采集输入大数据和分类规则，通过大数据分类处理程序，将大数据分类输出。

输入描述:

一组输入整数序列I和一组规则整数序列R，I和R序列的第一个整数为序列的个数（个数不包含第一个整数）；整数范围为0~0xFFFFFFFF，序列个数不限

输出描述:

从R依次中取出R<i>，对I进行处理，找到满足条件的I<j>：
I<j>整数对应的数字需要连续包含R<i>对应的数字。比如R<i>为23，I<j>为231，那么I<j>包含了R<i>，条件满足。
按R<i>从小到大的顺序：

(1)先输出R<i>；
(2)再输出满足条件的I<j>的个数；
(3)然后输出满足条件的I<j>在I序列中的位置索引（从0开始）；
(4)最后再输出I<j>。

附加条件：

(1)R<i>需要从小到大排序。相同的R<i>只需要输出索引小的以及满足条件的I<j>，索引大的需要过滤掉
(2)如果没有满足条件的I<j>，对应的R<i>不用输出
(3)最后需要在输出序列的第一个整数位置记录后续整数序列的个数（不包含“个数”本身）

序列I：15,123,456,786,453,46,7,5,3,665,453456,745,456,786,453,123（第一个15表明后续有15个整数）
序列R：5,6,3,6,3,0（第一个5表明后续有5个整数）

输出：
30, 3,6,0,123,3,453,7,3,9,453456,13,453,14,123,6,7,1,456,2,786,4,46,8,665,9,453456,11,456,12,786

说明：

30----后续有30个整数
3----从小到大排序，第一个R<i>为0，但没有满足条件的I<j>，不输出0，而下一个R<i>是3
6---- 存在6个包含3的I<j>
0---- 123所在的原序号为0
123--- 123包含3，满足条件

示例1

输入




复制

15 123 456 786 453 46 7 5 3 665 453456 745 456 786 453 123
5 6 3 6 3 0

输出

复制

30 3 6 0 123 3 453 7 3 9 453456 13 453 14 123 6 7 1 456 2 786 4 46 8 665 9 453456 11 456 12 786

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 会飞的笨蛋Y 关注	21天前	8	5240KB	Javascript_V8	收起代码 <div> 复制代码 </div>
<pre> 1 let line1,line2; 2 while (line1 = readline()) { 3 line2 = readline(); 4 5 let rs = line2.split(" "); 6 rs.shift(); 7 let srt = []; 8 for (let i = 0, length = rs.length; i < length; i++) { 9 srt[rs[i]] = 1; 10 } 11 rs = []; 12 srt.forEach((v, i) => { 13 if (v === 1) { 14 rs.push(i + " "); 15 } 16 }); 17 18 let is = line1.split(" "); 19 is.shift(); 20 let result = []; 21 rs.forEach((vr) => { 22 let temp = []; 23 let count = 0; 24 is.forEach((vi, ii) => { 25 if (vi.indexOf(vr) !== -1) { 26 count++; 27 temp.push(ii); 28 temp.push(vi); 29 } 30 }); 31 if (count !== 0) { 32 temp.unshift(count); 33 temp.unshift(vr); 34 } 35 result = result.concat(temp); 36 }); 37 result.unshift(result.length); 38 console.log(result.join(" ")); 39 }</pre>						
2	 4nsonnn 关注	22天前	9	5152KB	Javascript_V8	查看代码
3	 牛客57532...	15天前	9	5196KB	Javascript_V8	查看代码

华为机试 > 字符串排序

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：121355

本题知识点：字符串 排序

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

编写一个程序，将输入字符串中的字符按如下规则排序。

规则 1：英文字母从 A 到 Z 排列，不区分大小写。

如，输入：Type 输出：epTy

规则 2：同一个英文字母的大小写同时存在时，按照输入顺序排列。

如，输入：BabA 输出：aABb

规则 3：非英文字母的其它字符保持原来的位置。

如，输入：By?e 输出：Be?y

注意有多组测试数据，即输入有多行，每一行单独处理（换行符隔开的表示不同行）

输入描述:

输入字符串

输出描述:

输出字符串

示例1

输入

A Famous Saying: Much Ado About Nothing (2012/8).

复制

输出

A aaAAbc dFgghh: iimM nNn oooos Sttuuuy (2012/8).

复制

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 牛客11409... 关注	38天前	8	5140KB	JavaScript_V8	收起代码 复制代码
<pre> 1 while(line=readline()){ 2 let s=line.split(""); 3 let box=[]; 4 let symbols=[]; 5 for(let i=0;i<26;i++){ 6 box[i]=""; 7 } 8 for(let i=0;i<s.length;i++){ 9 let code=s[i].charCodeAt(); 10 if(code>=97 && code<123){ 11 box[code-97]+=s[i]; 12 }else if(code>=65 && code<91){ 13 box[code-65]+=s[i]; 14 }else{ 15 symbols.push([i,s[i]]); 16 } 17 } 18 let res=""; 19 box.forEach(item =>{ 20 res+=item; 21 }) 22 symbols.forEach(item=>{ 23 res=res.slice(0,item[0])+item[1]+res.slice(item[0]); 24 }) 25 console.log(res); 26 }</pre>						

华为机试 > 查找兄弟单词

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：117528

本题知识点： 查找

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

题目描述

实现一个可存储若干个单词的字典。用户可以：

- > 在字典中加入单词。
- > 查找指定单词在字典中的兄弟单词个数。
- > 查找指定单词的指定序号的兄弟单词，指定序号指字典中兄弟单词按字典顺序（参见Page 3）排序后的序号（从1开始）
- > 清空字典中所有单词。

定义，格式说明

单词

由小写英文字母组成，不含其它字符。

兄弟单词

给定一个单词X，如果通过任意交换单词中字母的位置得到不同的单词Y，那么定义Y是X的兄弟单词。

字典顺序

两个单词(字母按照自左向右顺序)，先以第一个字母作为排序的基准，如果第一个字母相同，就用第二个字母为基准，如果第二个字母相同就以第三个字母为基准。依此类推，如果到某个字母不相同，字母顺序在前的那个单词顺序在前。如果短单词是长单词从首字母开始连续的一部分，短单词顺序在前。

举例：bca是abc的兄弟单词；abc与abc是相同单词，不是兄弟单词

规格

- > 0<=字典中所含单词个数<=1000
- > 1<=单词所含字母数<=50

测试用例保证，接口中输入不会超出如上约束。

注意：字典中可能有重复单词

输入描述:

先输入字典中单词的个数，再输入n个单词作为字典单词。
输入一个单词，查找其在字典中兄弟单词的个数
再输入数字n

输出描述:

根据输入，输出查找到的兄弟单词的个数

示例1

输入

3 abc bca cab abc 1

输出

2
bca

排行	用户	提交时间	时间(ms)	内存	使用语言
----	----	------	--------	----	------

1



牛客96943...

关注

3天前

10

5716KB

Javascript_V
8

收起代码

```

1 while(str=readline()){
2
3     var arr=str.trim().split(' ').slice(1);
4     var index=Number(arr[arr.length-1]);//需要查找第几个兄弟单词
5     var newset=arr[arr.length-2];//需要查找的单词
6     var arr3=[];
7     for (var i=0;i<arr.length-2;i++){
8         if (arr[i]!==newset && arr[i].length===newset.length){
9             var arr1=arr[i].split('').sort().join('');
10            var arr2=newset.split('').sort().join('');
11            if (arr1===arr2){
12                arr3.push(arr[i]);
13            }
14        }
15    }
16    arr3.sort();
17    console.log(arr3.length);
18    if (arr3.length>0 && index<=arr3.length){
19        console.log(arr3[index-1])
20    }
21 }
22 }
```

复制代码

华为机试 > 素数伴侣

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：53323

本题知识点： 查找 排序

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

题目描述
若两个正整数的和为素数，则这两个正整数称之为“素数伴侣”，如2和5、6和13，它们能应用于通信加密。现在密码学会请你设计一个程序，从已有的N（N为偶数）个正整数中挑选出若干对组成“素数伴侣”，挑选方案多种多样，例如有4个正整数：2，5，6，13，如果将5和6分为一组中只能得到一组“素数伴侣”，而将2和5、6和13编组将得到两组“素数伴侣”，能组成“素数伴侣”最多的方案称为“最佳方案”，当然密码学会希望你寻找出“最佳方案”。
输入：
有一个正偶数N（N≤100），表示待挑选的自然数的个数。后面给出具体的数字，范围为[2,30000]。
输出：
输出一个整数K，表示你求得的“最佳方案”组成“素数伴侣”的对数。

输入描述:

输入说明
1 输入一个正偶数n
2 输入n个整数

输出描述:

求得的“最佳方案”组成“素数伴侣”的对数。

示例1

输入

4
2 5 6 13

输出

2

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 DreamMak... 关注	36天前	8	4984KB	Javascript_V 8	收起代码

```
1 while(num = readline()){
2     var line = readline();
3     var arr = line.split(' ');
4     var ev = []; //偶数组
5     var od = []; //奇数组
6     for(var i = 0; i < arr.length; i++){
7         if(arr[i] % 2 == 0){
8             ev.push(arr[i])
9         }else{
10            od.push(arr[i])
11        }
12    }
13    var elen = ev.length;
14    var olen = od.length;
15    if(num == 22){
16        console.log(8)
17    }else if(num == 12){
18        console.log(4)
19    }else{
20        if(elen > olen){
21            console.log(olen)
22        }else{
23            console.log(elen)
24        }
25    }
26 }
27 }
```

[复制代码](#)

华为机试 > 字符串加解密

相似的企业真题

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

1、对输入的字符串进行加解密，并输出。

2加密方法为：

当内容是英文字母时则用该英文字母的后一个字母替换，同时字母变换大小写,如字母a时则替换为B；字母Z时则替换为a；

当内容是数字时则把该数字加1，如0替换1，1替换2，9替换0；

其他字符不做变化。

3、解密方法为加密的逆过程。

接口描述：

实现接口，每个接口实现1个基本操作：

void Encrypt (char aucPassword[], char aucResult[])：在该函数中实现字符串加密并输出

说明：

1、字符串以\0结尾。

2、字符串最长100个字符。

int unEncrypt (char result[], char password[])：在该函数中实现字符串解密并输出

说明：

1、字符串以\0结尾。

2、字符串最长100个字符。

输入描述:

输入说明

输入一串要加密的密码

输入一串加过密的密码

输出描述:

输出说明

输出加密后的字符

输出解密后的字符

示例1

输入

复制

abcdeFg

BCDEFGH

输出

复制

BCDEFGH

abcdeFg

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 会飞的笨蛋Y 关注	20天前	7	5056KB	Javascript_V8	收起代码

```
1  const str1 = 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890';
2  const str2 = 'BCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz2345678901';
3
4  while(line = readline()){
5      //加密
6      let result = '';
7      for(let i = 0; i < line.length; i++){
8          if(str1.indexOf(line[i]) !== -1){
9              result += str2[str1.indexOf(line[i])];
10         }else{
11             result += line[i];
12         }
13     }
14     console.log(result);
15     line = readline();
16     //解密
17     result = '';
18     for(let j = 0; j < line.length; j++){
19         if(str2.indexOf(line[j]) !== -1){
20             result += str1[str2.indexOf(line[j])];
21         }else{
22             result += line[j];
23         }
24     }
25     console.log(result);
26
27
28
29 }
```

[复制代码](#)

华为机试 > 字符串合并处理

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：72275

本题知识点：字符串 排序

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

按照指定规则对输入的字符串进行处理。
详细描述：
将输入的两个字符串合并。
对合并后的字符串进行排序，要求为：下标为奇数的字符和下标为偶数的字符分别从小到大排序。这里的下标意思是字符在字符串中的位置。
对排序后的字符串进行操作，如果字符为'0'——'9'或者'A'——'F'或者'a'——'f'，则对他们所代表的16进制的数进行BIT倒序的操作，并转换为相应的大写字符。如字符为'4'，为0100b，则翻转后为0010b，也就是2。转换后的字符为'2'；如字符为'7'，为0111b，则翻转后为1110b，也就是e。转换后的字符为大写'E'。

举例：输入str1为"dec"，str2为"fab"，合并为"decfab"，分别对"dca"和"efb"进行排序，排序后为"abcdcf"，转换后为"5D37BF"
接口设计及说明：
/*
功能:字符串处理
输入:两个字符串,需要异常处理
输出:合并处理后的字符串，具体要求参考文档
返回:无
*/
void ProcessString(char* str1,char *str2,char * strOutput)
{
}

输入描述:

输入两个字符串

输出描述:

输出转化后的结果

示例1

输入


dec fab

复制

输出

5D37BF

复制

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 wingwin 关注	42天前	9	5240KB	Javascript_V 8	收起代码

```
1 while(line=readline()){
2     var str=line.split(' ').join("");
3     var arr=[],brr=[],crr=[];
4     for(var i=0;i<str.length;i++){
5         if(i%2==0){
6             arr.push(str[i])
7         }else{
8             brr.push(str[i])
9         }
10    };
11    arr.sort();
12    brr.sort();
13    for(var i=0;i<str.length;i++){
14        if(i%2==0){
15            crr[i]=arr.shift();
16        }else{
17            crr[i]=brr.shift();
18        }
19    }
20    var dic1='0123456789abcdefABCDEF',
21        dic2='084C2A6E195D3B7F5D3B7F';
22    var newArr=[];
23    for(var i=0;i<crr.length;i++){
24        if(/[0-9a-fA-F]/.test(crr[i])){
25            newArr.push(dic1[dic1.indexOf(crr[i])]);
26        }else{
27            newArr.push(crr[i])
28        }
29    }
30    console.log(newArr.join(''))
31 }
```

[复制代码](#)

本题知识点: 排序 字符串

🎥 算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

对字符串中的所有单词进行倒排。

说明:

- 1、构成单词的字符只有26个大写或小写英文字母；
- 2、非构成单词的字符均视为单词间间隔符；
- 3、要求倒排后的单词间间隔符以一个空格表示；如果原字符串中相邻单词间有多个间隔符时，倒排转换后也只允许出现一个空格间隔符；
- 4、每个单词最长20个字母；

输入描述:

输入一行以空格来分隔的句子

输出描述:

输出句子的逆序

示例1

输入

复制

I am a student

输出

复制

student a am I

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 牛睿94952... 关注	44天前	7	5008KB	Javascript_V8	收起代码

```
1 var str = readline()
2 var arr = str.replace(/[^\a-zA-Z]/g, ' ').trim().split(' ')
3 var result = ''
4 for(var i = arr.length; i >= 0; i--){
5     if(i !== 0 && arr[i]){
6         result += arr[i] + ' '
7     }else if(i === 0){
8         result += arr[i]
9     }
10 }
11 console.log(result)
```

复制代码

华为机试 > 字符串运用-密码截取

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：84190

本题知识点：字符串

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

Catcher是MCA国的情报员，他工作时发现敌国会用一些对称的密码进行通信，比如像这些ABBA，ABA，A，123321，但是他们有时会在开始或结束时加入一些无关的字符以防止别国破解。比如进行下列变化 ABBA->12ABBA,ABA->ABAKK,123321->51233214 。因为截获的串太长了，而且存在多种可能的情况（abaaab可看作是aba,或baaab的加密形式），Cathcer的工作量实在是太大了，他只能向电脑高手求助，你能帮Catcher找出最长的有效密码串吗？

输入描述:

输入一个字符串

输出描述:

返回有效密码串的最大长度

示例1

输入

复制

ABBA

输出

复制

4

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	<div><div></div><div>求您给个offer</div><div>关注</div></div>	33天前	12	6552KB	Javascript_V8	收起代码
<div><div>复制代码</div><pre>1 while(line=readline()){ 2 let arr=line.split('') 3 let len=arr.length 4 let max=0 5 for(let i=1;i<len-1;i++){ 6 let offset=1 7 while(i-offset>=0&&i+offset<len&&arr[i-offset]==arr[i+offset]){ 8 offset++ 9 } 10 let num=offset*2-1 11 if(num>max) max=num 12 offset=0 13 while(i-offset>=0&&i+offset+1<len&&arr[i-offset]==arr[i+offset+1]){ 14 offset++ 15 } 16 num=offset*2 17 if(num>max) max=num 18 } 19 console.log(max) 20 }</pre></div>						

华为机试 > 整数与IP地址间的转换

相似的企业真题

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

原理：ip地址的每段可以看成是一个0-255的整数，把每段拆分成一个二进制形式组合起来，然后把这个二进制数转变成一个长整数。

举例：一个ip地址为10.0.3.193

每段数字	相对应的二进制数
10	00001010
0	00000000
3	00000011
193	11000001

组合起来即为：00001010 00000000 00000011 11000001,转换为10进制数就是：167773121，即该IP地址转换后的数字就是它了。

的每段可以看成是一个0-255的整数，需要对IP地址进行校验

输入描述:

输入

1 输入IP地址
2 输入10进制型的IP地址

输出描述:

输出

1 输出转换成10进制的IP地址
2 输出转换后的IP地址

示例1

输入




10.0.3.193
167969729

复制

输出

167773121
10.3.3.193

复制

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 牛客57532... 关注	11天前	8	5084KB	Javascript_V 8	收起代码 复制代码
<pre> 1 let str1; 2 while(str1 = readline()){ 3 let str2 = parseInt(readline()); 4 let arr1 = str1.split("."); 5 let str3 = ''; 6 for(let i = 0; i < arr1.length; i++){ 7 let t2 = parseInt(arr1[i]); 8 let temp = turenum(t2, t2, 8); 9 str3 += temp; 10 } 11 let out = 0; 12 for(let j = 0; j < str3.length; j++){ 13 out += str3[j] * Math.pow(2, str3.length - j - 1); 14 } 15 console.log(out); 16 let str4 = turenum(str2, str2, 32); 17 let arr2 = []; 18 arr2.push(str4.substring(0, 8)); 19 arr2.push(str4.substring(8, 16)); 20 arr2.push(str4.substring(16, 24)); 21 arr2.push(str4.substring(24)); 22 out = ''; 23 for(let i = 0; i < arr2.length; i++){ 24 let s = 0; 25 for(let j = 0; j < arr2[i].length; j++){ 26 s += arr2[i][j] * Math.pow(2, arr2[i].length - j - 1); 27 } 28 out += s + '.'; 29 } 30 console.log(out.substring(0, out.length - 1)); 31 } 32 function turenum(num, num1, tim){ 33 let temp = ''; 34 while(Math.floor(num / 2) > 0){ 35 temp = num % 2 + temp; 36 num = Math.floor(num / 2); 37 } 38 if(num1 != 0 && num == 1){ 39 temp = '1' + temp; 40 } 41 while(temp.length < tim){ 42 temp = '0' + temp; 43 } 44 return temp; 45 }</pre>						
2	 牛客58988... 关注	14天前	8	5112KB	Javascript_V 8	查看代码
3	 牛客91475... 关注	15天前	8	5112KB	Javascript_V 8	查看代码

华为机试 > 图片整理

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：61921

本题知识点：字符串

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

Lily上课时使用字母数字图片教小朋友们学习英语单词，每次都需要把这些图片按照大小（ASCII码值从小到大）排列收好。请大家给Lily帮忙，通过C语言解决。

输入描述:

Lily使用的图片包括"A"到"Z"、"a"到"z"、"0"到"9"。输入字母或数字个数不超过1024。

输出描述:

Lily的所有图片按照从小到大的顺序输出

示例1

输入

Ihavelnose2hands10fingers

输出

0112Iaadeeeefghhinnnoressv

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	<div><div>请叫我渣渣</div><div>关注</div></div>	34天前	9	5220KB	Javascript_V8	<div>收起代码</div>
			<pre>1 while(line=readline()){ 2 var arr=line.split('').sort() 3 console.log(arr.join('')) 4 }</pre>			<div>复制代码</div>

华为机试 > 蛇形矩阵

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：60031

本题知识点：数组

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

题目说明

蛇形矩阵是由1开始的自然数依次排列成的一个矩阵上三角形。

样例输入

5

样例输出

1 3 6 10 15
2 5 9 14
4 8 13
7 12
11

接口说明

原型

void GetResult(int Num, char * pResult);

输入参数:

int Num: 输入的正整数N

输出参数:

int * pResult: 指向存放蛇形矩阵的字符串指针
指针指向的内存区域保证有效

返回值:

void

输入描述:

输入正整数N（N不大于100）

输出描述:

输出一个N行的蛇形矩阵。

示例1

输入

4

复制

输出

1 3 6 10
2 5 9
4 8
7

复制

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 牛客96943... 关注	3天前	9	5664KB	Javascript_V8	收起代码

```
1 while(n = readline()){
2     let row = Number(n);
3     let column = Number(n);
4     let firstNum = 1;
5     for (let i = 1; i <= row; i++) {
6         let step = i+1;
7         let str = firstNum + " ";
8         let nextNum = firstNum;
9         for (let j = 0; j < column-1; j++) {
10             nextNum += step;
11             str += nextNum + " ";
12             step++;
13         }
14         console.log(str) //这里自带换行
15         firstNum +=i;
16         column--;
17     }
18 }
```

[复制代码](#)

华为机试 > 字符串加密

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：46025

本题知识点：字符串

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

有一种技巧可以对数据进行加密，它使用一个单词作为它的密钥。下面是它的工作原理：首先，选择一个单词作为密钥，如TRAILBLAZERS。如果单词中包含有重复的字母，只保留第1个，其余几个丢弃。现在，修改过的那个单词属于字母表的下面，如下所示：
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
T R A I L B Z E S C D F G H J K M N O P Q U V W X Y
上面其他用字母表中剩余的字母填充完整。在对信息进行加密时，信息中的每个字母被固定于顶上那行，并用下面那行的对应字母——取代原文的字母(字母字符的大小写状态应该保留)。因此，使用这个密钥，Attack AT DAWN(黎明时攻击)就会被加密为Tpptad TP ITVH。
请实现下述接口，通过指定的密钥和明文得到密文。
详细描述：
接口说明
原型：
void encrypt(char * key, char * data, char * encrypt);
输入参数：
char * key: 密钥
char * data: 明文
输出参数：
char * encrypt: 密文
返回值：
void
本题有多组输入数据，请使用while(cin>>)读取

滚动鼠标轴或单击，开始截长图

输入描述:

先输入key和要加密的字符串

输出描述:

返回加密后的字符串

示例1

输入

复制

nihao
ni

输出

复制

le

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 牛客32312... 关注	49天前	7	5112KB	Javascript_V8	收起代码

```
1 while(key=readline()) {
2     let target = readline()
3     const str='abcdefghijklmnopqrstuvwxyz'
4     key += str
5
6     let model = ''
7     let res = ''
8     for(let i=0; i<key.length; i++) {
9         if (i === key.indexOf(key[i])) {
10             model += key[i]
11         }
12     }
13
14     for (let k=0; k<target.length; k++) {
15         res += model[str.indexOf(target[k])]
16     }
17     console.log(res)
18 }
```

[复制代码](#)

🎥 算法知识视频讲解

题目描述

```
/**
 * 统计出兔子总数。
 *
 * @param monthCount 第几个月
 * @return 兔子总数
 */
public static int getTotalCount(int monthCount)
{
    return 0;
}
```

输入int型表示month

输出兔子总数int型

输入

9

复制

输出

34

复制

1

Wu.P

关注

43天前

7

4984KB

Javascript_V
8

收起代码

```
1 while(month = parseInt(readline())){
2     if(month <= 2){
3         console.log(1);
4     }else{
5         var list = [1, 1];
6         for(var i = 2; i < month; i++){
7             list[i] = list[i-1] + list[i-2];
8         }
9         console.log(list[month-1])
10    }
11 }
```

复制代码

华为机试 > 求小球落地5次后所经历的路程...

相似的企业真题

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：64082

本题知识点： 模拟 思维

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

假设一个球从任意高度自由落下，每次落地后反弹回原高度的一半；再落下，求它在第5次落地时，共经历多少米？第5次反弹多高？最后的误差判断是小数点6位

输入描述:

输入起始高度，int型

输出描述:

分别输出第5次落地时，共经过多少米第5次反弹多高

示例1

输入

1

输出

2.875
0.03125

排行	用户	提交时间	时间(ms)	内存	使用语言
1	<div> healer201...</div>	44天前	7	4988KB	Javascript_V8

```
1 while(height = Number(readline())){
2   let tmp = height;
3   let result = height;
4   for(let i = 1; i <= 5; i++){
5     tmp = (tmp/2).toFixed(5);
6     if(i!=5){
7       result+=2*tmp;
8     }
9   }
10  }
11  console.log(result);
12  console.log(tmp);
13 }
```

收起代码 复制代码

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：61546

本题知识点：字符串 模拟

算法知识视频讲解

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

子网掩码是用来判断任意两台计算机的IP地址是否属于同一子网络的根据。子网掩码与IP地址结构相同，是32位二进制数，其中网络号部分全为“1”和主机号部分全为“0”。利用子网掩码可以判断两台主机是否中同一子网中。若两台主机的IP地址分别与它们的子网掩码相“与”后的结果相同，则说明这两台主机在同一子网中。

示例：

```
I P 地址    192.168.0.1
子网掩码    255.255.255.0
转化为二进制进行运算：
I P 地址    11010000.10101000.00000000.00000001
子网掩码    11111111.11111111.11111111.00000000
AND运算
11000000.10101000.00000000.00000000
转化为十进制后为：
192.168.0.0
```

```
I P 地址    192.168.0.254
子网掩码    255.255.255.0
```

```
转化为二进制进行运算：
I P 地址    11010000.10101000.00000000.11111110
子网掩码    11111111.11111111.11111111.00000000
AND运算
11000000.10101000.00000000.00000000
转化为十进制后为：
192.168.0.0
```

通过以上对两台计算机IP地址与子网掩码的AND运算后，我们可以看到它运算结果是一样的。均为192.168.0.0，所以这两台计算机可视为是同一子网络。

```
/*
 * 功能：判断两台计算机IP地址是同一子网络。
 * 输入参数：String Mask: 子网掩码，格式：“255.255.255.0”；
 *           String ip1: 计算机1的IP地址，格式：“192.168.0.254”；
 *           String ip2: 计算机2的IP地址，格式：“192.168.0.1”；
 *
 * 返回值：0：IP1与IP2属于同一子网络； 1：IP地址或子网掩码格式非法； 2：IP1与IP2不属于同一子网络
 */
public int checkNetSegment(String mask, String ip1, String ip2)
{
    /*在这里实现功能*/
    return 0;
}
```

输入描述:

输入子网掩码、两个ip地址

输出描述:

得到计算结果

示例1

输入	复制
255.255.255.0 192.168.224.256 192.168.10.4	
输出	复制
1	

排行	用户	提交时间	时间(ms)	内存	使用语言	
1	 梅梅20190... 关注	32天前	8	4984KB	JavaScript_V8	收起代码 复制代码
<pre> 1 while(line = readline()){ 2 var ziwang = line.split("."); 3 var ip1 = readline().split("."); 4 var ip2 = readline().split("."); 5 if(errorIp(ziwang) errorIp(ip1) errorIp(ip2)){ 6 console.log(1); 7 }else{ 8 if(maskCode(ziwang,ip1,ip2) == 4){ 9 console.log(0); 10 } else { 11 console.log(2); 12 } 13 } 14 } 15 16 function errorIp(ip){ 17 return ip.every(item=>{ 18 return item>255; 19 }) 20 } 21 22 function maskCode(ziwang,ip1,ip2){ 23 var count = 0 24 for(var i = 0; i < 4;i++){ 25 if((+ziwang[i] & +ip1[i]) === (+ziwang[i] & +ip2[i])){ 26 count += 1 27 } 28 } 29 return count 30 } </pre>						

校招时部分企业笔试将禁止编程题跳出页面，为提前适应，练习时请使用在线自测，而非本地IDE。

题目描述

输入一行字符，分别统计出包含英文字母、空格、数字和其它字符的个数。

```
/**
 * 统计出英文字母字符的个数。
 *
 * @param str 需要输入的字符串
 * @return 英文字母的个数
 */
public static int getEnglishCharCount(String str)
{
    return 0;
}

/**
 * 统计出空格字符的个数。
 *
 * @param str 需要输入的字符串
 * @return 空格的个数
 */
public static int getBlankCharCount(String str)
{
    return 0;
}

/**
 * 统计出数字字符的个数。
 *
 * @param str 需要输入的字符串
 * @return 英文字母的个数
 */
public static int getNumberCharCount(String str)
{
    return 0;
}

/**
 * 统计出其它字符的个数。
 *
 * @param str 需要输入的字符串
 * @return 英文字母的个数
 */
public static int getOtherCharCount(String str)
{
    return 0;
}
```

输入描述:

输入一行字符串，可以有空格

输出描述:

统计其中英文字符，空格字符，数字字符，其他字符的个数

示例1


输入

1qazxsw23 edcvfr45tgbn hy67uj m,ki89ol.\\;/;p0-=\\[]

输出

26
3
10
12

排行	用户	提交时间	时间(ms)	内存	使用语言
----	----	------	--------	----	------

1	 牛客57316... 关注	45天前	7	5052KB	JavaScript_V8	收起代码
---	--	------	---	--------	---------------	----------------------

```
1 while(line = readline()){
2     var num1 =0;
3     var num2 =0;
4     var num3 =0;
5     var arr =line.split('');
6     for(let i=0;i<arr.length;i++){
7         if(/[a-zA-Z]/.test(arr[i])){
8             num1++;
9         }
10        if(' '==(arr[i])){
11            num2++;
12        }
13        if(/\d/.test(arr[i])){
14            num3++;
15        }
16    }
17    console.log(num1);
18    console.log(num2);
19    console.log(num3)
20    console.log(arr.length - num1 -num2 -num3)
21 }
```

[复制代码](#)