

[首页](#)[新闻](#)[博问](#)[专区](#)[闪存](#)[班级](#)[代码改变世界](#)[注册](#)[登录](#)

# Sweetness

lz?

[博客园](#)[首页](#)[新随笔](#)[联系](#)[管理](#)[订阅](#)

随笔- 194 文章- 0 评论- 45 阅读- 46106

## 贪心经典例题大赏

突然安利的网站：Usaco；

1.Chocolate Buying

[【题目连接】](#)

昵称： Sweetness

园龄： 2年7个月

粉丝： 8

关注： 16

[+加关注](#)

<	2021年10月						>
日	一	二	三	四	五	六	
26	27	28	29	30	1	2	
3	4	5	6	7	8	9	
10	11	12	13	14	15	16	
17	18	19	20	21	22	23	
24	25	26	27	28	29	30	
31	1	2	3	4	5	6	

搜索

找找看

# [BZOJ2014] [Usaco2010 Feb] Chocolate Buying

 谷歌搜索

🕒 2014年10月20日 📄 1,650 🗨 1

## Description

贝西和其他奶牛们都喜欢巧克力，所以约翰准备买一些送给她们。奶牛巧克力专卖店里  
有N种巧克力，每种巧克力的数量都是无限多的。每头奶牛只喜欢一种巧克力，调查显示，  
有Ci头奶牛喜欢第i种巧克力，这种巧克力的售价是Pi。

约翰手上有B元预算，怎样用这些钱让尽量多的奶牛高兴呢？下面举个例子：假设约翰  
有50元钱，商店里有5种巧克力：

巧克力品种	单价	高兴的奶牛数量
1	5	3
2	1	1
3	10	4
4	7	2
5	60	1

显然约翰不会去买第五种巧克力，因为他钱不够，不过即使它降价到50，也不该选择  
它，因为这样只会让一头奶牛高兴，实在太浪费了。最好的买法是：第二种买1块，第一种  
买3块，第四种买2块，第三种买2块，正好用完50元，可以让1+3+2+2=8头牛高兴。

首先看 $B \leq 10^{18}$ ，显然B很大了，因此如果我们写了一个复杂度带B的算法，显然会炸，一般复杂度在 $1e8$ 左右可以跑过，因此我们要设计一个时间复杂度不带B的算法；

SOLUTION：

把巧克力按价格从小到大排，然后分两种情况：1.剩余的钱可以使所有喜欢这种巧克力的牛都开心；2.没有足够余钱使喜欢这种巧克力的牛全都开心，只能使一部分开心；

严谨证明：

其实很好理解，花同样的钱，如果可以使更多的牛开心，我们显然选择花更少的钱使更多的牛开心；

贪心的三种证法：

1. 反证

## 常用链接

[我的随笔](#)[我的评论](#)[我的参与](#)[最新评论](#)[我的标签](#)[更多链接](#)

## 我的标签

[培训简 \(fan\) 摘\(44\)](#)[DP\(41\)](#)[test\(29\)](#)[图论\(17\)](#)[乱搞\(16\)](#)[字符串\(12\)](#)[数论\(11\)](#)[数据结构\(10\)](#)[搜索\(9\)](#)[dfs\(9\)](#)[更多](#)

## 随笔档案

[2020年5月\(3\)](#)[2020年4月\(4\)](#)[2020年2月\(6\)](#)

## 2. 直觉

## 3. 替换

## 2.Creative Accounting

**[BZOJ3544] [ONTAK2010] Creative Accounting**

© 2015年2月28日 1,560 2

**Description**

给定一个长度为N的数组a和M，求一个区间[l,r]，使得 $(\sum_{i=l}^r a_i) \bmod M$ 的值最大，求出这个值，注意这里的mod是数学上的mod

(使得： $\sum a_i \ (i: l \sim r)$ )

看到求区间和问题，应该条件反射前缀和；

然后将题目求原数组区间和最大转化为求原数组的前缀和差最大；

2020年1月(4)

2019年12月(6)

2019年11月(12)

2019年10月(2)

2019年9月(5)

2019年8月(16)

2019年7月(35)

2019年6月(42)

2019年5月(9)

2019年4月(23)

2019年3月(18)

2019年2月(8)

**相册**

背景(5)

**阅读排行榜**

1. 【洛谷p2142】高精度减法(2133)

2. 【总结】01背包问题(1653)

3. 【洛谷p1025】数的划分(1239)

4. 【洛谷p1981】表达式求值(1221)

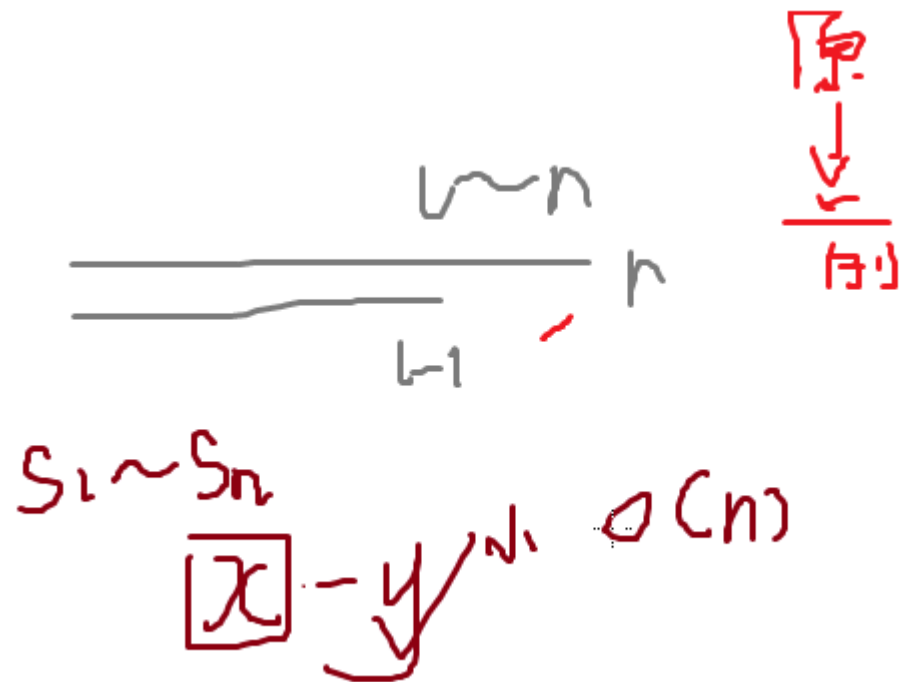
5. 【洛谷p1190】接水问题(1117)

**评论排行榜**

1. 2019清明期间qbxt培训qwq(9)

2. 【6.10校内test】noip模拟(4)

3. 【洛谷p1258】小车问题(4)



然后我们假设模M后差最大的两个前缀和为x和y;

我们枚举x从 $s_1 \sim s_n$ , 然后对于y, 有两种情况: (应该是要保证y在x前面)

1.  $x \geq y$ , 此时  $(x-y) \bmod M$  的值就为  $x-y$  所得结果再模M; 因此最大值可能是  $x-y$ , 那么找最小的y, 这样可以使得  $x-y$  的值最大;

2.  $y > x$ , 此时  $(x-y) \bmod M$  的值是  $x+M-y$  的结果再模M; 所以我们要使结果最大, 就要找到大于x的且最接近x的y;

4. 神仙网络最大流(3)

5. [寒假家里蹲培训]D1.1基础算法(2)

## 推荐排行榜

1. 可持久化数组 (1)

2. [HAOI2012] 音量调节(1)

3. [NOIp pj 2007][Luogu P1095] 守望者的逃离(1)

4. [Noip 2009 普及组 T4] [Luogu P1070] 道路游戏(1)

5. 跟风Manacher算法整理(1)

## 最新评论

1. Re:CSP-S 2019 杂记

%%%大佬

--Seaway-Fu

2. Re:几种简单的博弈 1

宁老了

--小蒟蒻皮皮鱼

3. Re:[寒假家里蹲培训]D1.1基础算法

@ 小蒟蒻皮皮鱼大早上的不困🐼...

--Sweetness

4. Re:[寒假家里蹲培训]D1.1基础算法

orz

--小蒟蒻皮皮鱼

5. Re:可持久化数组

orz

--小蒟蒻皮皮鱼

$$\left\{ \begin{array}{l} x-y \quad (y \leq x) \\ x+M-y \quad (y > x) \end{array} \right.$$

$1 \sim n$   $|S_i| = S_j$

突然插画：

```
#include <set>
#include <map>
```

比较重要的qwq ↑

好，都不会用。

### 3.会场安排

【连接】[http://ybt.ssoier.cn:8088/problem\\_show.php?pid=1422](http://ybt.ssoier.cn:8088/problem_show.php?pid=1422)

#### 会场安排问题

时间限制：3000 ms | 内存限制：65535 KB

难度：4

#### 描述

学校的小礼堂每天都有许多活动，有时这些活动的计划时间会发生冲突，需要选择出一些活动进行举办。小刘的工作就是安排学校小礼堂的活动，每个时间最多安排一个活动。现在小刘有一些活动计划的时间表，他想尽可能的安排更多的活动，请问他该如何安排。

然后之前是做过的，贪心的按照右端点从小到大排序，遍历每一个区间，如果没有重叠就选择这个活动；

这是我们的做法↑

然后这个是lyd的思路：

线段 $L[i], R[i]$

按 $R$ 把线段排序

$j$ ：当前枚举到哪一条线段

for ( $i = M; i \geq 0; i--$ )

枚举所有右端点比 $i$ 小的线段

找到其中左端点最大的，左端点为 $k$

$i = k$

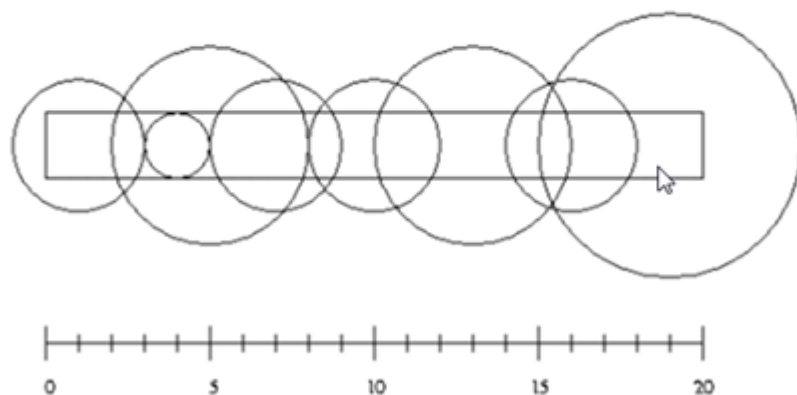


要注意，楼上↑的for循环枚举的是时间；

4.喷水装置：

$n$  sprinklers are installed in a horizontal strip of grass  $l$  meters long and  $w$  meters wide. Each sprinkler is installed at the horizontal center line of the strip. For each sprinkler we are given its position as the distance from the left end of the center line and its radius of operation.

What is the minimum number of sprinklers to turn on in order to water the entire strip of grass?



突然安利教辅：

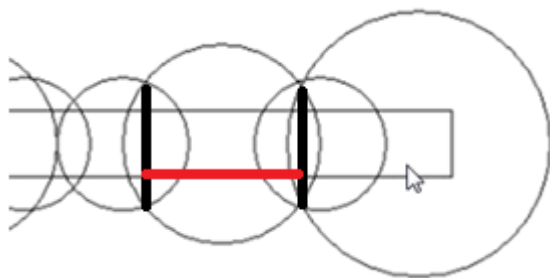
刘汝佳：算法竞赛入门经典训练指南（紫）算法

算法竞赛入门经典训练指南（蓝）刷题

黑书qwq；

算法导论nice~；

对于一个圆来说，真正有用的部分：



所以我们可以转化成线段覆盖问题啦；

处理每个圆代表一段线段；

求最小的线段条数，使之完全覆盖；

流程：读入=>处理成线段=>线段覆盖=>输出；

写程序=>结构化；



才鸿老大爷

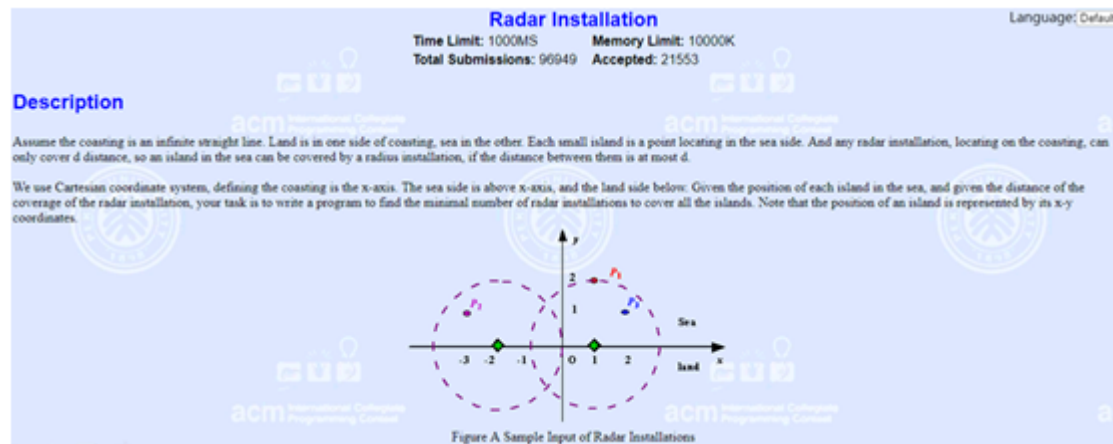
先抽象成线段，然后将所有的区间按左端点从小到大排序，处理每一个区间，每次选择覆盖点s的区间中右端点最小的一个



LCEZ 杨晔

半径小于等于矩形宽一半的圆就直接舍弃

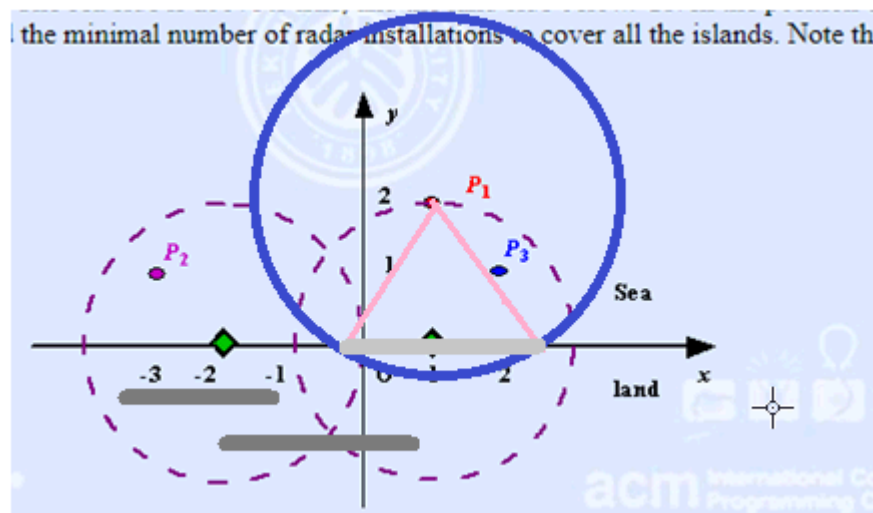
5.



SOLUTION:



以岛屿为 处理突破点，然后以 $d$ （给出的雷达半径）为半径，以岛屿为圆心画圆，找到与 $x$ 轴两个交点，这样每个岛屿对应一条线段；



转化为点覆盖线段问题；

然后将所有线段按照区间右端点从小到大排序，如果区间右端点相同的，按照区间左端点从大到小排，然后定义一个 `nowend`，表示当前最右的一个点放在了哪里，如果当前区间的左端点比 `nowend` 小（说明 `nowend` 也被覆盖在当前区间），`continue`；否则的我们在当前区间的最右放一个雷达，`nowend=ennd[i]`，答案+1；

[View Code](#)

6.

## Description

在一款电脑游戏中，你需要打败 $n$ 只怪物（从1到 $n$ 编号）。为了打败第 $i$ 只怪物，你需要消耗 $d[i]$ 点生命值，但怪物死后会掉落血药，使你恢复 $a[i]$ 点生命值。任何时候你的生命值都不能降到0（或0以下）。请问是否存在一种打怪顺序，使得你可以打完这 $n$ 只怪物而不死掉

贪心的想，显然我们应该先把杀完以后可以使血量增加的杀掉，这样可以积累更多的血，以备后面那些杀完之后使血量减少的怪物用。

因此我们可以分成两个部分，分别记录杀完以后血量增加的，和杀完以后血量减少的。

对于使血量增加的部分，很容易想我们应该先杀伤害小的（如果啥伤害大的可能出现gameover但实际还有更有解的可能[en.....感性理解一下]），所以对于使血量增加的部分，我们按照伤害d从小到大排序。

对于使血量减少的部分，我们按照回血量a从大到小排序：因为显然剩下的不管杀哪一个，血量都会在之前的血量下减少，因此我们尽量让回血多一些；

然后按照以上思路排序之后，直接模拟即可；

如果半路出现血量 $\leq 0$ 了，直接输出NIE，return 0；

然后一定要记得将当前血量z开long long

 View Code

首先，先把杀掉能回血的先杀了

显然杀的顺序按照消耗升序

杀完以后，不管用什么顺序杀掉剩下的怪，最后体力last是确定的

倒序来看，相当于将血药吐出来然后返还杀怪的消耗，那么显然也是按照损失体力（即血药回血量）升序，正回来即是降序。

即分为两部分，杀完能回血的按照消耗升序，剩余按血药回血量降序，然后模拟一遍判断是否合法即可



【树状数组】能可tfil\_retaw

杀的顺序按照消耗升序



【树状数组】能可tfil\_retaw

剩余按血药回血量降序

7.

## 阶乘之和

时间限制: 3000 ms | 内存限制: 65535 KB

难度: 3

### 描述

给你一个非负数整数 $n$ ，判断 $n$ 是不是一些数（这些数不允许重复使用，且为正数）的阶乘之和，如 $9=1!+2!+3!$ ，如果是，则输出Yes，否则输出No；

很好想的思路，首先打表1~k的阶乘（数据量应该很小的qwq）

然后我们从最大的阶乘减过来，如果最后可以减为0，就是某个数的阶乘，否则不是；

8.

## 最大排列(Perm)

### 问题描述

给一个长度为  $N$  的序列，每个元素都不相同。每次可以交换相邻两个元素，现在你最多可以交换  $K$  次，问你可通过交换得到的字典序最大的序列是什么。

### 输入格式

第一行两个数  $N$  和  $K$ 。

第二行  $N$  个数。

### 输出格式

一行  $N$  个数字，表示字典序最大的序列。

### 输入样例

```
7 1
10 20 30 40 50 60 70
```

### 输出样例

```
20 10 30 40 50 60 70
```

两种思路：

1. 可以把最大的提到最前：

把最大的往前提，然后再把次大的往前提；

2.不能把最大值提到最前面

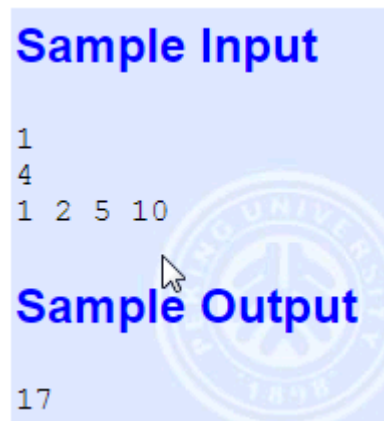
把前k个中最大的往前提

9.过河问题：

### 描述

在漆黑的夜里，N位旅行者来到了一座狭窄而且没有护栏的桥边。如果不借助手电筒的话，大家是无论如何也不敢过桥去的。不幸的是，N个人一共只带了一只手电筒，而桥窄得只够让两个人同时过。如果各自单独过桥的话，N人所需要的时间已知；而如果两人同时过桥，所需要的时间就是走得比较慢的那个人单独行动时所需的时间。问题是，如何设计一个方案，让这N人尽快过桥。

输入输出样例：



首先解释样例：

1~2过河，1送回来；

5~10过河，2送回来；

1~2过河；

直觉：最慢两人要搭伙过河；

设AB为最快的，CD为最慢的

把CD送过河的两种方案：

AC	A	AD	A
AB	A	CD	B

C	+	D	+	2	*	A
D	+	2	*	B	+	A

取楼上两种方法的最小值；

将CD送过河之后，变成子问题；人数每次减2；

减到最后，有两种情况：

1.四个人，直接在进楼上的操作；

2.三个人，A来回送叭：

3	A	B	C	
A	+	B	+	C

连接：[一个一定要好好提溜出来的贪心题](https://www.cnblogs.com/zhuier-xquan/p/11180406.html)