

Problem A. 关于时间复杂度

Time limit: 1 second

Memory limit: 256 megabytes

在比赛中你可能会遇见复杂度上的问题，例如一个算法程序运行时间过长，所以在实现算法前，我们会估算算法的时间复杂度。

关于复杂度，前人有很丰富的理论，这里我们只用最简洁的方式介绍时间复杂度。同一个算法在不同的计算机上运行的速度会有一定的差别，并且实际运行速度难以在理论上进行计算，实际去测量又比较麻烦，所以我们通常考虑的不是算法运行的实际用时，而是算法运行所需要进行的基本操作的数量。

影响时间复杂度的最主要因素是循环，我们通常可以通过估算指令循环进行的次数来得到一种时间复杂度。如以下程序

```
1 int k=0;
2 for (int i=0;i<N;++i)
3   for (int j=0;j<M;++j)
4     ++k;
```

根据乘法原理，其中 $++k$ 的指令被执行了 $N * M$ 次，我们用 $O(NM)$ 表示其时间复杂度，其中 $O(x)$ 表示近似。

通常来说，如果时间复杂度达到了 $O(10^8)$ 则实际评测程序会运行约 $1s$ 。本次新手赛的题目大多是以 $1s$ 为时间复杂度上限，如果你的算法达到了 $O(10^8)$ 则仍有机会通过，但是如果达到了 $O(10^9)$ 及以上则估计要考虑重新设计算法或者改善常数部分了。

注意数据的输入输出会消耗比较大的时间，例如 C 语言中使用 `scanf` 能读入约 10^6 个整数，`printf` 类似。

C++ 的 `cin/cout` 与 Python 的 `input/print` 开销会更大，所以如果这些方法输入可能会超时，则请转用 C 语言的 `scanf` 输入。

Input

本题没有输入数据。

本题需要你阅读下列代码，在已知变量 n 的范围为 $[1, 10^5]$ 后，判断哪些代码不会超出 $1s$ 的时间复杂度限制。

```
1 // 此代码编号为 A
2 int k=0;
3 for (int i=1;i<n;++i)
4 for (int j=i+1;j<=n;++j)
5 ++k;
```

```
1 // 此代码编号为 B
2 int k=0;
3 for (int i=1;i<n;++i)
4 for (int j=1;j*j<=n;++j)
5 ++k;
```

```
1 // 此代码编号为 C
2 int k=0;
3 for (int i=1;i<n-1;++i)
4 for (int j=1;i*j<=n;++j)
5 ++k;
```

```
1 // 此代码编号为 D
2 for (int i=1;i<n;++i) {
3     int j=i;
4     while (j>0) {
5         if (j%2==0)
6             j=j/2;
7         else
8             j-=1;
9     }
10 }
```

Output

你的程序需要输出一行字符串，按编号顺序升序输出不会超时的代码的编号。

例如如果你认为编号 A,B,D 不会超时，则输出

```
1 ABD
```

其中编号为大写字母，编号间没有用来间隔的字符，字符串前后没有空格。

Problem B. 老实点 每次都有你

Time limit: 1 second

Memory limit: 256 megabytes

众所周知, Xcw 是 pcr 中的人气角色 (Xcw: 骑士君哥哥可以再叫我一次镜华吗), 同时她也受 Lanly 委托, 来到新手赛咨询群中处理事务。

Lanly 不总是有时间教 Xcw 做更多的事——这也给了群友们时不时整出些让人两眼一黑的活的机会——Lanly 不仅忙于自己的工作, 还要抽出时间到 pcr 中刷装备。

这一天, Lanly 在 pcr 中进行了 n 次扫荡, 每次扫荡都获得了一定数量的金币, 其中第 i 次扫荡获得的金币数量为 c_i 。

Lanly 发现每次扫荡得到的金币数量不同, 但是由于今天是 k 倍掉落的日子, 所以第 i 次扫荡实际得到的金币数量 c_i 为这次扫荡中初始得到的金币数量 a_i 乘 k 。其中 k 与任意的 a_i 都是正整数。

Lanly 忘记看今天是几倍掉落了, 但 Lanly 有这 n 次扫荡的数据。请问你能否通过分析这 n 次扫荡的数据, 得到今天的掉落倍数 k 可能是哪些值呢?

Input

输入数据分为多组数据。其中第一行输入一个整数 T , 表示数据组数。

接下来 T 组数据, 每组数据格式如下:

第一行输入一个整数 n , 表示 Lanly 今天进行了 n 次扫荡。

第二行输入 n 个整数, 其中第 i 个整数 c_i 表示第 i 次扫荡得到的金币数量。

Output

每组数据输出一行, 为将 k 的所有可能取值从小到大输出。

Example

standard input	standard output
3	1 3
6	1 2 4
6 27 30 18 39 9	1 7
7	
8 36 32 20 36 52 12	
6	
14 56 42 28 49 21	

Note

$1 \leq T \leq 5, 1 \leq n \leq 10^3$

对于任意 i 满足, $1 \leq c_i \leq 10^8$

提示: 存在与开方相关复杂度的求因数算法。

Problem C. 木桶效应

Time limit: 1 second

Memory limit: 256 megabytes

爷爷说：“你和我去把几只破木桶拆下来吧。”

杂物房角落的破木桶，我仍有印象，记得木桶的形状是不太规则的，听说用来造这几只木桶的每块板的高度都不一致，所以也没有太能派上用场。小时候许多承载了珍贵回忆的小物件被堆积在那几只破木桶里，兴许已经蒙上了厚厚的尘。

爷爷说：“木桶的拆除要从板的角度拆，一个桶先拆下最长的那块板，然后拆第二长的，这样子去拆，最省力。然后木桶要一个一个去拆，因为你还要收拾桶里的东西，收拾好一个就给我拆一个。对了，你先去收拾东西最少的那桶，加快速度。”

东西最少？我怎么会记得哪桶里放的多少东西呢？

爷爷说：“木桶效应，还记得吧？我记得你小时候我就是指着这几只桶和你讲的木桶效应，一只桶能装多少，取决于桶的最短板。短板最短的桶，就是东西最少的桶。”

好吧，也许是这样的。我开始翻箱倒柜找卷尺准备量木板的长度。

爷爷开始说：“额，我记得，好像当初造的时候，最短的那块板好像是和第二长的那块板在一起的，造同一个桶.....是第二长吗？额.....”

我来兴趣了，如果木桶的构成真如爷爷的记忆的一致，那么到最后，各块木板会按怎样的顺序被拆下来呢？

Input

输入数据分为多组数据。第一行输入一个整数 T ，表示数据组数。

接下来输入 T 组数据，每组数据格式如下：

第一行输入两个整数 n, m ，分别表示木板的总数，以及爷爷记忆中板与板的匹配数。

接下来输入 m 行，每行输入两个整数 x, y ，表示爷爷记得第 x 短的木板与第 y 短的木板被用来造同一个桶。

一个桶可以由多块木板构成，例如如果爷爷记得第 x 短与第 y 短的木板被用来造同一个桶，第 x 短与第 z 短的木板也被用来造同一个桶，说明 x, y, z 三块板都被用来造同一个桶。

当然有许多块木板没有被爷爷提及，那么我们不妨假设这些木板单块木板就能构成一个木桶。

Output

每组数据输出一行，含 n 个整数，表示按顺序输出木板被拆卸的顺序，其中第 x 短的木板用整数 x 表示。

木板的拆卸简单来讲就是找到未被拆下的最短木板所在的木桶，将此木桶最长的木板拆下，以此类推直到全部拆完。

Example

standard input	standard output
3	9 5 1 7 4 3 2 6 8 10
10 5	10 9 8 7 5 4 3 2 1 6
3 2	6 1 2 10 9 8 7 3 4 5
2 7	
4 2	
5 9	
1 5	
10 8	
7 2	
1 10	
2 3	
2 8	
3 9	
1 2	
5 1	
4 1	
10 5	
7 10	
8 7	
7 9	
1 6	
3 7	

Note

数据满足 $1 \leq T \leq 5, 1 \leq m \leq n \leq 10^5, 1 \leq x, y \leq n$

Problem D. 究极手

Time limit: 1 second

Memory limit: 256 megabytes

《塞尔达传说：王国之泪》将会在 2023.05.12 发售，Windberu 表示这个 4 月份是一天也呆不下去了。

前阵子，Windberu 看见了《王国之泪》的实机演示视频，激动无比，对其中林克的各项新能力非常感兴趣。

其中林克获得的一项能力是「究极手」，它让林克能够将许多材料连接拼装成一个整体，只需要敲击这个整体的任意一个位置，就会有一个信号被传递，启动连接在这个整体里的风扇机。如下图展示的这样



Windberu 想要写程序实现「究极手」的功能——在 Windberu 的设计中，一共有 N 个材料被组成一个整体，不同材料有不同数量的接口，第 i 个材料的接口数量为 p_i 。两个不同的材料 i, j 间可以各消耗一个接口被「究极手」连接起来，用 (i, j) 表示，要求 $i < j$ 。

Windberu 希望你能帮忙测试一下已经写好的程序。Windberu 会告诉你材料的数量 N 以及各个材料的接口数量 p_i ，希望你给 Windberu 一个使用「究极手」连接材料的测试数据，测试数据含使用「究极手」的次数以及分别在哪两个材料间使用。要求所有材料到最后都被连接起来成为一个整体，且每个材料的所有接口都被用上。

但是 Windberu 在尝试实现信号传递的功能时发现程序卡住了，检查后发现当材料的连接结构出现环的时候会出问题，例如 $(1, 2)(1, 3)(2, 3)$ 或者 $(1, 2)(1, 2)$ 这样的结构出现时，信号会在材料间无止境地循环。

Windberu 发现自己修不好这个 bug，于是反其道而行之，Windberu 要求你给的测试数据里不能出现这样的结构。

得知这样的奇怪要求的你依然愿意给 Windberu 测试数据，但是你也降下神罚，惩戒 Windberu 明年才能玩到《王国之泪》。

Input

输入数据包含多个数据点，每个数据点格式如下

第一行输入一个正整数 N ，表示材料的数量，材料从 1 开始从小往大编号。

第二行输入 N 个整数，其中第 i 个整数表示第 i 个材料的接口数量 p_i 。

当输入的 N 为 0 时表示数据输入结束。

Output

对于每个数据点，若不可能得到能满足输入数据的测试数据，则仅输出一行，内容为 -1 。若能得到能满足输入数据的测试数据，则输出测试数据。如果包含多个能满足的测试数据，输出任意一个即可。其中测试数据的格式为

第一行输出一个整数 M ，表示「究极手」的使用次数。

接下来 M 行，每行包含两个整数 i, j ，表示使用一次「究极手」将材料 i 与材料 j 连接起来，要求 $1 \leq i < j \leq N$ ，可以按任意顺序输出。要求所有材料到最后都被连接起来成为一个整体，且每个材料的所有接口都被用上，且不能出现环结构。

Example

standard input	standard output
3	-1
2 2 2	5
6	5 6
1 1 4 1 2 1	3 5
13	3 4
1 9 1 1 3 2 1 1 1 1 1 1 1	2 3
6	1 3
1 1 1 1 1 5	12
2	6 13
1 1	5 6
0	5 12
	2 5
	2 11
	2 10
	2 9
	2 8
	2 7
	2 4
	2 3
	1 2
	5
	5 6
	4 6
	3 6
	2 6
	1 6
	1
	1 2

Note

输入数据满足：所有数据点的 N 的总和 $1 \leq \sum N \leq 10^6$ ，每个数据点满足所有 $1 \leq p_i \leq 10^8$ 。

输出数据要求： $0 \leq M \leq 10^8$ ，任意 (i, j) 要求 $1 \leq i < j \leq N$ 。

(PS: 事实上 Windberu 买不起 Nintendo Switch，所以即使《王国之泪》发售也玩不到《王国之泪》)

Problem E. Password

Time limit: 1 second

Memory limit: 256 megabytes

— H002 Observation Record #78 —

最后一个人类死去了。

按照人类的习俗，我开始整理你的遗物。

你的物品总是很整洁，似乎总是那几件衬衫，以及有些旧了的白大褂。每当我问起时，“真理总是很简洁哦。”，你会微敛双眸，这样淡淡地回答道。

“什么是真理呀？”

“存在没有意义。很简洁吧？”

人类最后的科学家，那个永远站在宇宙的尽头，用冷冰冰的真理藐视一切荒谬的研究者，现在却给系统上了密码。

“Hint: $0 \sim n$ 中，使得 $\text{sqrt}(5 * x^2 + 4)$ 为整数的最大的 x ”

我皱起了眉头，好复杂，一点也不像你。

所以，其实这一道题的答案，也一定很简洁吧？这样想着，我输入了密码。

Input

第一行输入一个 T ，表示数据的组数。

接下来 T 行，每行输入一个 n 。

Output

对于每一个 n ，输出一个 x ，表示 $0 \sim n$ 中，使得 $\text{sqrt}(5 * x^2 + 4)$ 为整数的最大的 x

Example

standard input	standard output
4	8
12	3
4	1
2	0
0	

Note

输入数据满足： $0 \leq n < 10^{18}$ 。

好想知道，这样的你，也会有在乎的事情吗？

让我来看一看吧。

— EOF —

WARNING：偏执或错误的人类思想

本记录中蕴含的思想可能偏执或错误，任何分析者不应认同或模仿其中的观点，并严格遵循《人类档案分析规范》。

Problem F. 熊爷别笑了

Time limit: 1 second

Memory limit: 256 megabytes

FNaF 是 Windberu 非常喜欢的游戏之一，但 Windberu 在多数时间里是云玩家，因为玩偶们的 Jump Scare 实在让人心脏受不了。

终于有一天，Windberu 试图自己上手玩 FNaF 的时候被 Freddy 吓得心脏停跳半拍，于是 Windberu 忍不了了，恨不得自己跳进披萨店里建个围栏把玩偶们围起来。

披萨店可以被看作一个二维的平面，每只玩偶正处于披萨店里的某些点上，第 i 只玩偶的坐标为 (x_i, y_i) 。每只玩偶都具有对身体四周的感知能力，由于玩偶们是同一批一起生产的，所以它们能感知到的区域统一为以自身坐标 (x_i, y_i) 为圆点，以 R 为半径的一个开区域圆，玩偶们的感知区域可以有交集。

Windberu 需要将玩偶们围起来，不过 Windberu 清楚，要是围围栏过程中稍有不慎就得接受玩偶的 Jump Scare。因此围栏需要满足以下要求：

- 围栏要将所有玩偶围进密闭的区域里。即从玩偶的坐标出发，不存在路径使其能不跨过围栏到达无穷远。
- 围栏要确保所有玩偶是连通的。即对于任意两只玩偶，存在一条路径不跨过围栏连接两只玩偶的坐标。
- 围栏过程要确保不进入玩偶的感知区域。即不存在围栏上的点在任一玩偶的感知区域内。
- 在满足以上要求前提下，要求围栏长度尽可能短短。不然围栏不及时是会遇上 Jump Scare 的。

Windberu 想知道，在得知所有玩偶的坐标，以及它们感知区域的半径 R 以后，围栏的最短长度是多少。

Input

输入数据分为多组数据，第一行输入一个整数 T ，表示数据数量。

紧接着 T 组数据，每组数据格式如下：

第一行为一个整数 n 与一个实数 R ，分别表示玩偶的数量与玩偶的感知区域的半径。

接下来 n 行每行包含两个实数 x_i, y_i ，表示披萨店里有一只玩偶的坐标为 (x_i, y_i) 。

Output

每组数据输出一行，含一个实数，表示将玩偶按题目要求围起来后，围栏的最小长度，要求与最终答案精度差不超过 10^{-6} 。

Example

standard input	standard output
2	12.39822971502571
4 0.7	4.82842712474619
1 1	
-1 1	
-1 -1	
1 -1	
3 0.0	
1 0	
0 1	
0 -1	

Note

数据满足 $1 \leq T \leq 10, 1 \leq \sum N \leq 10^5, 0 \leq R \leq 10^8$

对于每只玩偶满足 $-10^8 \leq x_i, y_i \leq 10^8$

Problem H. Pause! Heal! Explosion!

Time limit: 1 second

Memory limit: 256 megabytes

由于你博学，审问，慎思，明辨，笃行，并且常年在咨询群里调戏 Xcw，所以 Xcw 开着电子大卡车将你送到了异世界当龙/凤傲天。

这是由 Hitass_SY 创造的剑与魔法的异世界，这个设定完备的异世界要游戏性有游戏性，要美术有游戏性，要剧情有游戏性，要音乐有游戏性。你非常喜欢这个世界，并且选择转职成为一位大魔法师。

这一天，你在单刷副本时遇见了一位凶恶的 Boss，与其展开了激烈的战斗。战斗为回合制，每回合由你进入行动轮行动，随后到 Boss 进入行动轮。Boss 有 A 点血量，有 B 点攻击力，每回合轮到 Boss 的行动轮的时候，如果 Boss 可以行动，则会对你造成 B 点无法避免的真实伤害，即扣除 B 点 HP。

作为大魔法师，你拥有无穷大的 HP 上限与 MP 上限，但是在前面的战斗中，你只剩下了 C 点 HP 与 0 点 MP。当你在 Boss 的攻击下 HP 小于等于 0 的时候，你就会被判定为战斗失败。

你装备的 Xcw 赠送的精力戒指使得每回合轮到你的行动轮的时候，你会自动回复 1 点 MP，随后你可以使用现有的 MP 释放技能与 Boss 战斗。

你并不是那种只会释放爆裂魔法的红魔族魔法师，你掌握了 n 种技能。各个技能的代价与效果不尽相同，但是主要被分为三类：

1. 这类技能为攻击技能，若第 i 种技能为攻击技能，则发动此技能会消耗 x_i 点 MP，并对 Boss 造成 y_i 点伤害。
2. 这类技能为控制技能，若第 i 种技能为控制技能，则发动此技能会消耗 x_i 点 MP，并控制 Boss y_i 回合 (包含本回合)，使其在其行动轮中不能发动攻击。
3. 这类技能为治疗技能，若第 i 种技能为治疗技能，则发动此技能会消耗 x_i 点 MP，并回复自身 y_i 点 HP。

每回合你的行动轮中你能释放技能，但是不能预支 MP，即释放技能后 MP 不能小于 0。当你使用攻击魔法将 Boss 的血量削减至小于等于 0 则被判定为胜利。

由于你接下来还有其它区域需要探索，你希望战胜 Boss 后你剩余的 HP 尽可能多——你不怎么在意 MP 会剩下多少。你想知道你是否能击败 Boss，以及如果你击败了 Boss 被判定为胜利时，你能剩余 HP 最大为多少。

Input

输入数据分为多组数据，第一行输入一个整数 T ，表示数据数量。

紧接着 T 组数据，每组数据格式如下：

第一行为四个整数 A, B, C, n ，分别表示 Boss 的血量与攻击力，你的 HP 与掌握的技能数量。

接下来 n 行每行输入三个整数 $type, x_i, y_i$ ，分别表示你掌握的第 i 种技能的类型，MP 消耗与效果，其中 $type$ 取值为 $[1, 3]$ ，分别对应题面所指的三类技能。

Output

每组数据仅输出一行，表示你击败 Boss 时能剩下的最大 HP。

若你不能击败 Boss 则输出

```
1 0
```

若你能在击败 Boss 前使 HP 趋于无穷大，则输出

```
1 inf
```

Example

standard input	standard output
3	5
10 2 9 3	0
1 1 1	inf
2 3 4	
1 3 5	
7 3 9 3	
1 1 1	
3 7 9	
1 3 5	
6 2 9 3	
1 1 1	
3 1 3	
1 3 5	

Note

$1 \leq T \leq 50$

$1 \leq A, B, C, n \leq 10^3$ ，对于任意 i 满足 $1 \leq x, y \leq 10^6$ 。

你发现由于 Hitass_SY 由于空闲时间缺乏，选择专攻异世界的游戏性，把异世界的美术、音乐交给了 Windberu 实现。而 Windberu 很懒，总是摸鱼，赶 DDL 完成了自己的工作，导致效果很差。可是你感到奇怪，为什么艺术性如此糟糕的异世界还能吸引那么多人转生到来呢？

哦，原来是异世界原住民每天都通过照片与影像做日记记录生活，而 Xcw 神教导人民用广州光锥元信息科技有限公司的产品，完善照片与影像对异世界美好生活的重现效果，使这些日记上传到社交平台后异世界饱受好评。

广州光锥元信息科技有限公司是一家面向海外市场的互联网公司，致力于用前沿技术和创意设计，打造世界领先的视频、图像创作 APP，风很大的 Oldroll、Proccd、PrettyUp、UnidreamAI 绘画等 APP 就是我们的作品！目前我们正在快速发展，全球用户超过 3 亿，多款 APP 在欧美、东南亚等地的应用市场摄影视频榜单排行前列。

我们提供有竞争力的薪酬福利、开放年轻的工作氛围、快速成长的职业机会。我们有良好的应届生文化和完善的新人培养机制，在这里同学们将有机会由浅入深，快速学习成长，我司也有很多中大毕业的学长学姐，和我们成长为独当一面的大牛，详情请戳官网 <https://www.guangzhuiyuan.com/job.html>

Problem I. 小学生也能秒解的题

Time limit: 1 second

Memory limit: 256 megabytes

Lanly 告知 Windberu，在《名侦探柯南》的动画第 1135 集（对应漫画的第 1062 话）中，柯南在诸多 FBI 面前淡定地破解了暗号，令众人佩服。



这是柯南的破译过程，为将密文的底部 3 行进行左右翻转，随后发现从左往右每 5*5 个字符表示原文的一个字符，借此破译出了原文的各个字符。

现在 Windberu 也给你一段密文，告诉你原文仅含小写字母，希望你能帮忙破译得到原文。

第一行输入一个整数 n ，代表原文长度。接下来 5 行每行输入一个长度为 $5 * n$ 的仅含 .|- 三种字符的字符串，这 5 行字符串代表着加密后的文字。

输出一行长度为 n 的字符串，表示破译后的原文，用小写字母表示。

[illegible]

你可能需要依靠随题目提供的大数据完成本题。

Problem J. Abyssal Echoes

Time limit: 1 second

Memory limit: 256 megabytes

我们并排坐在深深的海底，看着波澜的海浪把阳光切碎。

—佚名

你醒来，丝缕光线透过沉沉的海水，游曳在你和她的身边。

迷茫地起身，一只章鱼惊慌地从她的手上跃起，舞动着长长的腕足，消失在了深海里。“啊！真是的... 你把它吓跑了...” 她直勾勾地盯着你，落寞地说，“我们在聊很重要的事情呢”。

什么呀，这样的，“我听到了哦，是在说我的坏话吧。”

一瞬，偌大的海床只剩下了深海的回响。终于，她抿着唇，咯咯地笑起来，“你又不知道它的频率，你怎么知道呢？”

深海里的生物们，都有着自己的频率。它们说出的所有话，都会异或上一个频率常数。假设章鱼的频率是 x ，人鱼的频率是 y ，为了传输消息 t ，他们要说三句话：

1. 章鱼对人鱼说出： $x \oplus t$
2. 人鱼对章鱼说出： $y \oplus x \oplus t$
3. 章鱼对人鱼说出： $y \oplus t$

现在人鱼就能够知道 t 究竟是什么了。

作为原人类的你，当然知道这一些知识。刚刚，你也偷听到了 $x \oplus t$ 、 $y \oplus x \oplus t$ 和 $y \oplus t$ 。你决心告诉人鱼小姐 x 、 y 和 t ，来证明你有读心术。

但是，你已经很久很久没有算过数了。“你能帮我算一算吗？”你对着趴在肩上的小海星问道。

请你给小海星写一个程序，让它按照程序，帮忙计算 x 、 y 和 t 。

Input

输入数据分为多组数据。第一行输入一个整数 T ，代表数据组数。

接下来 T 组数据，每组数据输入一行，含三个整数，代表 $x \oplus t$ 、 $y \oplus x \oplus t$ 和 $y \oplus t$ 。

Output

对于每一组数据，输出三个整数， x 、 y 和 t 。

Example

standard input	standard output
3	79 203 132
203 0 79	22 135 145
135 0 22	115 15 124
15 0 115	

Note

输入数据满足： $0 \leq x, y, t < 2^{64}$ 。

其中 \oplus 表示按位异或， $x \oplus y$ 即在二进制下， x 与 y 按对应位进行异或。异或的运算规则为相同为 0，不同为 1。

例如 $3 \oplus 6 = [0011]_2 \oplus [0110] = [0101]_2 = 5$ ，在 C/C++/Python 语言中用脱字符实现。例如

```
1 a^b
```

注意，由于数据较大，你需要使用 unsigned long long。在 linux 下使用 C 语言的你需要使用 %llu 输入整数：

```
1 unsigned long long a;  
2 scanf("%llu", &a);
```


Problem K. 晚餐铃

Time limit: 1 second

Memory limit: 256 megabytes

晚餐铃响起，像梦里那样呼唤着我。

因为不能选择，所以才不情愿地称之为爱吗？长围裙也是有温度的人。

无论什么时候我都会乖乖的，从前妈妈定下的规矩，某个寒夜里在路边弄丢了，打不开的门的剪影、白铃铛。

我不想吃的东西数也数不完，但如果一直这样的话我会慢慢死去，变得渺小而悲惨。喝下玩具厨师煮的热汤，用那份温暖熬到天亮。抱紧自己好好睡一觉吧。

美味的晚餐被分成 n 碗汤，连续一排摆在餐桌上。每碗汤都有一个温暖值，第 i 碗汤的温暖值为 a_i ，如果我一次喝下一段从第 L 到第 R 碗汤，那么带来的温暖会是第 L 到第 R 碗汤中温暖值最大的一碗汤的温暖值 $\max_{i=L}^R a_i$ 。

一段一段地喝完所有的汤，每段都会给我带来温暖，这些温暖加起来会是无比幸福的吧？我可以将汤分成任意段，每段为连续的任意长度，该怎样划分，使得最后带来的温暖的和**最大**呢？

像世俗那般，和大家一样，治好我的发烧吧，温柔地治好我吧，晚餐铃。



Input

输入数据分为多组数据，第一行输入一个整数 T ，表示数据数量。

紧接着 T 组数据，每组数据格式如下：

第一行为一个整数 n ，表示美味的晚餐被分成 n 碗汤。

第二行为 n 个整数，第 i 个整数为 a_i ，表示第 i 碗汤的温暖值。

Output

每组数据输出一行，含一个整数，表示这 n 碗汤的各种划分方案中，带来的温暖的和的最大值。

Example

standard input	standard output
4	16
4	16
7 2 2 5	12
4	10
7 5 3 1	
4	
5 5 -1 2	
4	
5 0 5 0	

Note

$1 \leq T \leq 5, 1 \leq n \leq 10^5, -10^9 \leq a_i \leq 10^9$

样例后两组数据的合适的划分方法分别为

(5)(5, -1)(2)

(5, 0)(5, 0)

Problem L. 晚餐铃

Time limit: 1 second

Memory limit: 256 megabytes

晚餐铃响起，像寻常那样呼唤着大家。

各种我所不拥有的东西散落一地，大家理所当然拿着的餐具，触碰就消失不见了。

在梦里我能飞翔，用以前魔女给的翅膀，回家路上透过窗口看到的温暖聚会，长围裙的剪影、白铃铛。

我没有的东西数也数不完，回家路上透过窗口看到的温暖聚会，在魔术光里的喧闹。每次每次看到他们我都忍不住想，那份温暖是怎么来的由什么组成的？那里的孩子知道答案是什么。

苦涩的回忆被分成 n 勺雪，连续一排洒在雪地上。每勺雪都有一个痛苦值，第 i 勺雪的痛苦值为 a_i ，如果我一次喝下一段从第 L 到第 R 勺雪，那么带来的痛苦会是第 L 到第 R 勺雪中痛苦值最大的一勺雪的痛苦值 $\max_{i=L}^R a_i$ 。

一段一段地喝完所有的雪，每段都会给我带来痛苦，这些痛苦加起来会是难以承受的吧？我可以将雪分成任意段，每段为连续的任意长度，该怎样划分，使得最后带来的痛苦的和**最小**呢？

被银雪覆盖着，在我的心脏变冷之前，呼唤我的名字吧，晚餐铃。



Input

输入数据分为多组数据，第一行输入一个整数 T ，表示数据数量。

紧接着 T 组数据，每组数据格式如下：

第一行为一个整数 n ，表示苦涩的回忆被分成 n 勺雪。

第二行为 n 个整数，第 i 个整数为 a_i ，表示第 i 勺雪的痛苦值。

Output

每组数据输出一行，含一个整数，表示这 n 勺雪的各种划分方案中，带来的痛苦的和的最小值。

Example

standard input	standard output
3	0
3	1
5 1 -5	-1
3	
5 -5 1	
4	
5 -7 3 -2	

Note

$$1 \leq T \leq 5, 1 \leq n \leq 10^5, -10^9 \leq a_i \leq 10^9$$

样例前两组数据的合适的划分方法分别为

$(5, 1)(-5)$

$(5)(-5)(1)$

Problem M. 梦与公交站

Time limit: 1 second

Memory limit: 256 megabytes

小 Zo 正在公交站等车，可是深夜里公交车已经不会再来了。在小 Zo 开始哭泣之前，空中出现了一位小精灵。小精灵告诉小 Zo，沿路一共有 n 处公交站，编号从 1 到 n ，许多公交站都分布有数量不定的小精灵，当小 Zo 到达一处公交站的时候，该公交站的小精灵会用传送魔法帮助小 Zo 到想去的公交站。

不过每只小精灵都有自己的特点，第 i 只小精灵喜欢数字 w_i ，因此它只会将小 Zo 送到编号为 k_i 的倍数的公交站；此外它有法力的限制，只能将小 Zo 送到编号在 $[L_i, R_i]$ 间的公交站；同时传送会花费 w_i 的时间。

小精灵已经知道公交站的数量 n ，当前它与小 Zo 所在公交站编号 S ，小精灵的数量 q 以及它们的所有信息。小精灵想要知道，将小 Zo 送去某一个公交站，最短需要多长时间。

Input

输入数据分为多组数据。第一行输入一个整数 T ，表示数据组数。

接下来输入 T 组数据，每组数据输入格式如下：

第一行输入三个整数 n, q, S ，分别表示公交站的数量，小精灵的数量与当前小 Zo 所在公交站编号。

随后 q 行，每行输入五个整数 u_i, L_i, R_i, k_i, w_i ，分别表示一只小精灵位于编号为 u_i 的公交站，能将小 Zo 送到编号在 $[L_i, R_i]$ 间且为 k_i 倍数的公交站，传送花费时间为 w_i 。

Output

每组数据输出一行，含 n 个整数，为从小 Zo 出发点 S 出发到各个编号的公交站的最短距离，按编号从小到大输出。

Example

standard input	standard output
3	0 3 8 3 12
5 5 1	0 1 3 4 4
1 1 4 1 13	2 2 0 6
1 2 5 2 3	
2 3 5 1 9	
4 2 3 1 5	
5 1 5 1 1	
5 5 1	
1 1 3 2 1	
2 1 3 1 2	
3 3 5 5 1	
2 2 4 2 3	
5 1 5 1 1	
4 3 3	
1 1 4 1 7	
3 1 3 1 2	
2 2 4 2 4	

Note

$1 \leq T \leq 5, 1 \leq n \leq 10000, 1 \leq q \leq 30000$ 。

对于任意 i ，满足 $1 \leq u_i \leq n, 1 \leq L_i, R_i \leq n, 1 \leq k_i \leq n, 1 \leq w_i \leq 10^9$ 。

数据保证从编号为 S 的公交站出发，小 Zo 能直接或间接被送到 1 到 n 中任意编号的公交站。