

模拟赛

(请选手务必仔细阅读本页内容)

一、题目概况

中文题目名称	负二进制	旋转减半数	进入太戈	上机体验
英文题目与子目录名	negbin	half	tiger	line
可执行文件名	negbin	half	tiger	line
输入文件名	negbin.in	half.in	tiger.in	line.in
输出文件名	negbin.out	half.out	tiger.out	line.out
每个测试点时限	1 秒	1 秒	1 秒	1 秒
测试点数目	10	10	10	10
每个测试点分值	10	10	10	10
附加样例文件	有	有	有	有
结果比较方式	全文比较，过滤末行后空行，不过滤中间行行末空格			
题目类型	传统	传统	传统	传统
运行内存上限	256M	256M	256M	256M

二、提交源程序文件名

对于 C++语言	negbin.cpp	half.cpp	tiger.cpp	line.cpp
----------	------------	----------	-----------	----------

三、编译命令 (不包含任何优化开关)

对于 C++语言	<code>g++ -o negbin negbin.cpp -lm</code>	<code>g++ -o half half.cpp -lm</code>	<code>g++ -o tiger tiger.cpp -lm</code>	<code>g++ -o line line.cpp -lm</code>
----------	---	---------------------------------------	---	---------------------------------------

负二进制

(negbin.cpp)

【问题描述】

除了二进制，-2 也可以作为进制的基数，我们称这种新的表示方法为负二进制。这种特殊进制的优点是：

- 不论正负，任何整数都有且只有一种表示方法；
- 表示负数不用负号 -

负二进制以 -2 作为基数，从最低位开始，每位的权重依次为 1, -2, ~4, -8, 16, ... 例如：

$$(111)_{-2} = (-2)_2 + (-2)_1 + (-2)_0 = 3$$

$$(1011)_{-2} = (-2)_3 + (-2)_1 + (-2)_0 = -9$$

$$(11010)_{-2} = (-2)_4 + (-2)_3 + (-2)_1 = 6$$

给定一个以十进制表示的整数 n ，请输出 n 的负二进制表示，头部不要出现多余的 0。

【输入格式】输入文件 **negbin.in**

单个整数，表示 n 。

【输出格式】输出文件 **negbin.out**

单个字符串，表示 n 的负二进制表示。

【输入输出样例 1】

negbin.in	negbin.out
-13	110111

【样例说明】 $-32 + 16 - 0 + 4 - 2 + 1 = -13$

【数据规模】

对于 50% 的数据， $-100 \leq n \leq 100$ ；

对于 100% 的数据， $-10^9 \leq n \leq 10^9$ 。

旋转减半数

(half.cpp)

时空限制：1s/256M，测试数据共 10 组

【问题描述】

842105263157894736 是一个很有趣的数字，因为如果将第一位数字 8 移动到最后，可以得到 421052631578947368 恰好是原来数字的一半。

这样旋转后恰好减半的数字在其他进制中也存在。比如在二进制下，10101010 就是一个旋转后减半的数字：

$$(1010)_2 = (0101)_2 + (0101)_2$$

给定一个正整数 b ，请求出 b 进制下最小的旋转减半数，所谓旋转减半数，就是将第一位数字移动到最后一位后，得到的新数字恰好在 b 进制下为原数字大小的一半。

如果不存在这样的数字，输出 None

【输入格式】输入文件 **half.in**

单个整数表示 b

【输出格式】输出文件 **half.out**

若干个自然数，每个数字表示一位数字，每位数字在 0 到 $b-1$ 之间，第一位不能为 0，每个数字间有一个空格。若不存在这样的数字，输出 None。

【输入输出样例 1】

half.in	half.out
2	1 0

【样例说明】 $10 = 01 + 01$

【输入输出样例 2】

half.in	half.out
10	1 0 5 2 6 3 1 5 7 8 9 4 7 3 6 8 4 2

【样例说明】 $105263157894736842 = 52631578947368421 + 52631578947368421$

【数据规模】

对于 30% 的数据， $2 \leq b \leq 10$ ；

对于 60% 的数据， $2 \leq b \leq 200$ ；

对于 100% 的数据， $2 \leq b \leq 2000$ 。

进入太戈

(tiger.cpp)

【问题描述】

太戈空间是一个令人向往的空间。但是在进入太戈空间之前，你需要先通过无人区，一块 $M \times N$ 格构成的区域，每个格子上有一个正整数。行的编号从 $1, 2, \dots, M$ ，列的编号从 $1, 2, \dots, N$ 。格子 (r, c) 表示第 r 行第 c 列的格子。

你现在位于左上角的格子 $(1, 1)$ ，太戈的入口在右下角 (M, N) 。你所在的格子上是正整数 x ，你可以跳跃到任意格子 (a, b) ，只要满足 $x = a \times b$ 的条件。例如现在所在格子数字是 6，那么可以跳到 $(2, 3)$ 。需要注意的是跳跃不能超过区域的边界，如果区域尺寸是 5×6 ，从数字 6 的格子可以跳跃到 $(2, 3)$, $(3, 2)$, $(1, 6)$, $(6, 1)$ ，但是总共 5 行，所以 $(6, 1)$ 是无法到达的。

【输入格式】输入文件 tiger.in

第一行一个整数 M ($1 \leq M \leq 1000$)。第二行一个整数 N ($1 \leq N \leq 1000$)。接下来 M 行，每行 N 个正整数，用空格分开，表示区域每个格子上的数字，每个数字均不超过 1000000。

【输出格式】输出文件 tiger.out

如果可以到达入口，输出 yes，否则输出 no。

【输入输出样例 1】

tiger.in	tiger.out
3 4 3 10 8 14 1 11 12 12 6 2 3 9	yes

【样例说明】起点 $(1, 1)$ 上的数字是 3，可以到达 $(1, 3)$ 。 $(1, 3)$ 上的数字是 8，可以跳到 $(2, 4)$ 。 $(2, 4)$ 上的数字是 12，可以跳到 $(3, 4)$ 。

【数据规模】

对于 20% 数据， $1 \leq m \leq 10, 1 \leq n \leq 10$;

对于 50% 数据， $1 \leq m \leq 50, 1 \leq n \leq 50$;

对于 70% 数据， $1 \leq m \leq 200, 1 \leq n \leq 200$;

对于 100% 数据， $1 \leq m \leq 1000, 1 \leq n \leq 1000$;

上机体验

(line.cpp)

时空限制：1s/256M，测试数据共 10 组

【问题描述】

冬令营上某公司赞助了一台超级计算机。每位与会人员都想要到超级计算机上体验一番。

由于只有一台超级计算机，人们体验时难免会排队等待。参会共 n 人($1 \leq n \leq 10^5$)，每个人都有个安排，在 t_i 时间来体验，体验共持续 l_i 时间。由于入会时间不等，先入会的人资历老，在体验时有优先权。

现场是这样的，如果计算机有人使用，那么新来的人需要等待。如果计算机当前无人使用，那么前来的人可以直接上机体验，并且持续 l_i 分钟，然后走人，后面的人继续上机体验。如果等待的人不止 1 个，那么按照资历先后，由资历最老的人先上机，其他人继续等待。

请问所有人体验完，排队队伍里等待时间最长的人等了多久？等待时间计算规则为这个人上机时间减去到达时间。

【输入格式】输入文件 line.in

输入的第一行包含 n 。接下来 n 行按资历顺序给出了 n 个人的信息（按照资历从大到小给出）。每行包含一个人的到达时间和持续时间。持续时间为不超过 10^4 的正整数，到达时间为不超过 10^9 的正整数。

【输出格式】输出文件 line.out

所有人中最长等待时间。

【输入输出样例 1】

line.in	line.out
5	10
25 3	
105 30	
20 50	
10 17	
100 10	

【样例说明】有 5 个人（按输入中的顺序编号为 1..5）。4 号最先到达（时间 10），在结束之前（时间 27）1 号和 3 号都到达了。由于 1 号资历老，所以先轮到他，他从到达开始共等待了 2 个单位时间。1 号在时间 30 结束，随后 3 号开始，从他到达开始共计等待了 10 单位时间。在一段没有人后，5 号到达，在他正在使用的时间里 2 号也到达了，在 5 个单位时间之后轮到 2 号。过程中等待时间最长的是 3 号。

【数据规模】

对于 10% 数据, $1 \leq n \leq 10$

对于 30% 数据, $1 \leq n \leq 10^4$

对于 100%的数据, $1 \leq n \leq 10^5$

太戈编程
www.etiger.vip