

2025 0101 CCF 非专业级软件能力认证

CSP-J/S 2025 第二轮模拟认证003

时间：2025 年 10月7 日 10:30 ~ 12:00

题目名称	课间休息时间设置	游泳同步	评估	拆分数字	露营	删数
题目类型	传统型	传统型	传统型	传统型	传统型	传统型
目录	rest	swim	assess	split	camp	move
可执行文件名	rest	swim	assess	split	camp	move
输入文件名	rest.in	swim.in	assess.in	split.in	camp.in	move.in
输出文件名	rest.out	swim.out	assess.out	split.out	camp.out	move.out
每个测试点时限	1.0秒	1.0秒	1.0秒	1.0秒	1.0秒	1.0秒
内存限制	512 MiB	512 MiB	512 MiB	512 MiB	512 MiB	512 MiB
测试点数目	10	10	10	10	10	10
测试点是否等分	是	是	是	是	是	是

提交源程序文件名

- 对于C++语言：`rest.cpp` `swim.cpp` `assess.cpp` `split.cpp` `camp.cpp` `move.cpp`

编译选项

| 对于C++语言 | `-O2 -std=c++14 -static` |

注意事项（请仔细阅读）

- 文件名（程序名和输入输出文件名）必须使用英文小写。
- main 函数的返回值类型必须是 int，程序正常结束时的返回值必须是 0。
- 提交的程序代码文件的放置位置请参考各省的具体要求。
- 因违反以上三点而出现的错误或问题，申诉时一律不予受理。

5. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。
6. 选手提交的程序源文件必须不大于 100KB。
7. 程序可使用的栈空间内存限制与题目的内存限制一致。
8. 全国统一评测时采用的机器配置为：Intel(R) Core(TM) i7-8700K CPU @3.70GHz，内存 32GB。
9. 只提供 Linux 格式附加样例文件。
10. 评测在当前最新公布的 NOI Linux 下进行，各语言的编译器版本以此为准。

T1 课间休息时间设置 (rest)

题目描述

噢猴今天上的课 08:00 开始。课程时间 (n) 分钟。分为两小节，中间有一次课间休息，课间休息 10 分钟。

请问噢猴！在什么时间开始课间休息，可以保证两小节课程的时间尽可能接近。如果有多种方式，输出第一节课时间少一点的那种。

输入格式

一行一个整数 (n)。

输出格式

一行一个时间，表示开始休息的时间。

样例

输入样例1

120

输出样例1

08:55

输入样例2

121

输出样例2

08:55

输入样例3

122

输出样例3

08:56

样例解释

- 样例1的两节课时间分别为：55分钟（08:08-08:55）、55分钟（09:05-10:08）
- 样例2的两节课时间分别为：55分钟（08:08-08:55）、56分钟（09:05-10:01）
- 样例3的两节课时间分别为：56分钟（08:08-08:56）、56分钟（09:06-10:02）

数据规模与约定

对于 100% 的数据， $(10 < n < 240)$

- 子任务1（30分）：保证 $(n = 240)$
- 子任务2（30分）：保证 (n) 为偶数
- 子任务3（40分）：没有特殊限制

T2 游泳同步（swim）

题目描述

- 噢最近在学游泳。
- 噢游泳的速度非常稳定。每次从起点出发后都能在 (a) 分钟内折返回到起点。
- 唔的游泳的速度也非常稳定。每次从起点出发后都能在 (b) 分钟内折返回到起点。

他们一起开始，游了 (n) 分钟。请问他们有多少次同时出现在了起点。

输入格式

一行三个整数 (a, b, n) 。

输出格式

一行一个整数，表示两个人同时出现在起点的次数。

样例

输入样例1

3 4 15

输出样例1

2

输入样例2

4 6 33

输出样例2

3

输入样例3

6 8 48

输出样例3

3

样例解释

- 样例1的2次同时出现在起点的时间分别为：0时刻，12时刻
- 样例2的3次同时出现在起点的时间分别为：0时刻，12时刻，24时刻
- 样例3的3次同时出现在起点的时间分别为：0时刻，24时刻，48时刻

数据规模与约定

对于 100% 的数据， $1 \leq a, b \leq 10^3, 1 \leq n \leq 10^4$

- 子任务 1（30 分）：保证 $a = 1$ 。
- 子任务 2（30 分）：保证 b 是 a 的倍数。
- 子任务 3（40 分）：没有特殊限制。

T3 评估 (assess)

题目描述

小明同学是一家科技公司数据分析部门的员工。一天，他获取到了一组长度为 (n) 的整数数列 (a_i) ，这个数列代表着每个时间段产品的性能数据。为了更深入地了解产品性能的波动情况，他需要计算 $\sum_{i=1}^{n-1} \sum_{j=i+1}^n |a_i - a_j|^2$ 来评估整体的差异程度（数列从 1 开始编号）。

但小明同学并不想去计算，于是他想请你帮忙。

输入格式

输入的第一行包含一个正整数 (n) ，表示数列的长度。

输入的第二行包含 (n) 个整数 (a_i) ，表示每个时间段产品的性能数据。

输出格式

输出共一行，包含一个整数，表示数列整体的差异程度。

样例 1 输入

3
2 8 4

样例 1 输出

56

样例 1 解释

$|2 - 8|^2 + |2 - 4|^2 + |8 - 4|^2 = 36 + 4 + 16 = 56$

样例 2 输入

5
-5 8 9 -4 -3

样例 2 输出

958

数据规模与约定

- 对于 40% 的数据，保证 $n \leq 1000, |a_i| \leq 10$ 。
- 对于 100% 的数据，保证 $n \leq 1 \times 10^5, |a_i| \leq 1000$ 。

T4 拆分数字（split）

题目描述

小明同学探索到一个古老的数学遗迹，在遗迹的深处发现了若干道神秘的谜题。谜题中给出了整数 (n) 和 (k)，并有如下提示："在这个神秘的地方，存在着一类特殊的数字，它们的形式为 (3^m) ((m) 是非负整数)。现在需要判断能否通过恰好 (k) 个这样的特殊数字相加，得到整数 (n)。"

换言之，是否存在一个非负整数序列 { a_k }，使得 $n = 3^{a_1} + 3^{a_2} + ... + 3^{a_k}$ 。

不出意外的，小明同学又把这个任务交给你了。

输入格式

输入的第一行包含一个正整数 (T)，表示谜题的个数。

接下来 (T) 行，每行两个整数 (n, k)，表示一道谜题中的信息。

输出格式

输出共 (T) 行。对于每一道谜题，如果可以则输出 Yes，否则输出 No。

样例 1 输入

4
5 3
17 2
163 79
1000000000000000000 1000000000000000000

样例 1 输出

Yes
No
Yes
Yes

样例 1 解释

对于第一个测试案例， $5 = 3^1 + 3^0 + 3^0$ ，因此满足了相关条件。
对于第二个测试案例，没有非负整数序列 a_1, a_2 使得 $17 = 3^{a_1} + 3^{a_2}$ ，因此不满足有关条件。

数据规模与约定

- 对于 30% 的数据，保证 $n \leq 10, k \leq 5$ 。
- 对于另 30% 的数据，保证 $n \leq 1000, k \leq 2$ 。
- 对于 100% 的数据，保证 $1 \leq k \leq n \leq 1 \times 10^9, 1 \leq T \leq 1 \times 10^5$ 。

T5 露营（camp）

题目描述

小明和他的两位好朋友要在野外露营，经过认真的规划后，他们将整片地区划分为若干个由 (1 *1) 的小方格共同组成的平面直角坐标系。

经过一番商议后，他们决定在 (A, B, C) 三点出发扎寨（(A, B, C) 各占据了一个(1 *1)的小方格，且相互之间位置不同）。由于野外有许多枯枝败叶，他们希望能够清理一些小方格，以便他们相互拜访。

具体来说，开始时，所有的小方格都被未被清理（包括 (A, B, C) 三个点）。他们可以通过相邻的清理之后的小方格，从其他方格到达 (A, B, C) 中的任何一个。如果两个小方格有共同边，则它们相邻。

因此，请你帮忙计算一下，他们最少需要清理的小方格数量有多少？

输入格式

输入三行，每行各两个整数，分别表示三人安营扎寨的位置。

输出格式

输出共一行，表示最少需要清理的小方格数量。

样例 1 输入

0 0
1 1
2 2

样例 1 输出

5

样例 1 解释

一种最优方案为清理 $((0, 0), (0, 1), (1, 1), (1, 2), (2, 2))$ 。

样例 2 输入

0 0

2 0

1 1

样例 2 输出

4

数据规模与约定

- 对于 30% 的数据，保证 $0 \leq x, y \leq 2$ 。
- 对于 100% 的数据，保证 $0 \leq x, y \leq 1000$ 。

T6 删数 (move)

题目描述

有 N 个不同的正整数 $x_1, x_2, \dots, x_N$ 排成一排，我们可以从左边或右边去掉连续的 i ($1 \leq i \leq n$) 个数（只能从两边删除数），剩下 $N - i$ 个数，再把剩下的数按以上操作处理，直到所有的数都被删除为止。

每次操作都有一个操作价值，比如现在要删除从 i 位置到 k 位置上的所有的数。操作价值为 $|x_i - x_k| \times (k - i + 1)$ ，如果只去掉一个数，操作价值为这个数的值。

问如何操作可以得到最大值，求操作的最大价值。

输入格式

第一行为一个正整数 (N)；

第二行有 (N) 个用空格隔开的 (N) 个不同的正整数。

输出格式

一行，包含一个正整数，为操作的最大值

输入输出样例 #1

输入 #1

6

54 29 196 21 133 118

输出 #1

768

说明/提示

样例解释和说明

说明：经过 3 次操作可以得到最大值，第一次去掉前面 3 个数：54、29、196，操作价值为 426。第二次操作是在剩下的三个数（21, 133, 118）中去掉最后一个数 118，操作价值为 118。第三次操作去掉剩下的 2 个数：21 和 133，操作价值为 224。操作总价值为 (426 + 118 + 224 = 768)。

数据范围

$$3 \leq N \leq 100 , 1 \leq x_i \leq 1000$$