# 2026 年 ACM 结训赛

2026 HGNU ACM Graduation Contest







## 题目目录


**A** 贪吃的乐乐
**B** 入场票价
**C** 刘学长开播啦
**D** 迟到登记
**E** 简单字符串
**F** 纸杯回收站
**G** 小新再次来袭
**H** 证书评定
**I** 参加比赛
**J** 字符串反转

# Problem A. 贪吃的乐乐

Input file: standard input
Output file: standard output

乐乐是个十足的小吃货，今天他来到了限时自助餐厅。餐厅规定，每位顾客的用餐时长最多只有 120 分钟，时间一到就会立刻停止用餐。

开餐前乐乐一次性取好了 $m$ 份爱吃的食物。乐乐吃东西十分专注，吃完一份才会接着吃下一份，中间没有任何休息停顿，每份食物都需要固定 $t$ 分钟才能完整吃完。

餐厅为了杜绝浪费，设置了惩罚规则：用餐时间结束之后，如果**剩下没吃完的食物份数严格大于** $k$，就判定为严重浪费，需要支付额外罚金；如果剩余份数小于或者等于 $k$，则不会被扣钱。

乐乐会一刻不停地吃，直到下面任意一种情况发生就停下：

1. 120 分钟的用餐时限耗尽；
2. 他把自己取的所有食物全部吃光。

请你帮乐乐计算两件事：用餐计时结束之后，还剩下多少份没有吃完的食物；并且判断乐乐最终会不会被扣钱。

注意特殊情况：如果剩余时间不足以完整吃完当前这一份，这份食物就会算作剩下的食物。

## Input

一行三个非负整数 $m, t, k$（$1 \le m \le 100$，$0 \le t < 150$，$0 \le k \le 100$），依次代表取的食物总份数、吃完一份食物需要的分钟数、允许剩余且不被扣钱的食物上限份数。

## Output

第一行输出一个整数：时限结束后剩余的食物份数。

第二行输出判断结果：会被扣钱输出 `1`，不会被扣钱输出 `0`。

## Examples

| standard input | standard output |
| --- | --- |
| 50 10 20 | 38<br>1 |
| 20 10 20 | 8<br>0 |

### 样例解释

样例 1 最多吃完 120/10 = 12 份，还剩 50 – 12 = 38 份，大于 20，输出 1。样例 2 还剩 20 – 12 = 8 份，不大于 20，输出 0。

# Problem B. 入场票价

Input file: standard input
Output file: standard output

某活动按以下顺序计算票价。

- 基础票价：未满 6 岁免费；6 至 17 岁 40 元；18 至 59 岁 80 元；60 岁及以上 30 元。
- 学生优惠：持学生证且年龄在 18 至 59 岁时，基础票价改为 50 元。其他年龄的学生证不影响票价。
- 高峰加价：入场时刻为 17、18、19 或 20 时，非免费票加收 20 元。免费票不加价。
- 优惠券：最后减 15 元；若不足 15 元，最终票价为 0 元。

请输出最终应付金额。

## Input

一行四个整数 $a, h, s, c$，分别表示年龄、入场时刻、是否持学生证、是否持优惠券。

$0 \le a \le 120$，$0 \le h \le 23$，$s, c \in \{0, 1\}$。

## Output

输出一个整数，表示应付金额（元）。

## Examples

| standard input | standard output |
| --- | --- |
| 18 17 1 1 | 55 |

### 样例解释

18 岁且持学生证，基础票价 50 元；17 时加收 20 元；优惠券再减 15 元，最后支付 55 元。

# Problem C. 刘学长开播啦

Input file: standard input
Output file: standard output

刘学长打瓦这款游戏很强，他最拿手的英雄就是尚博勒，因为这个英雄和他一样“优雅”。尚博勒是一名以“经济流”打法著称的哨卫。他在每一回合都需要根据当前手头金币，决定购买多少发猎头子弹、购买多少个陷阱，并把剩余金币保留下来。由于不同购买方案会影响本回合能够提供的火力，因此他希望在不超过当前金币数的前提下，选择一种最合适的购买方案，使本回合的“火力值”尽可能大。

本题只考察购买数量与预算约束，不考虑护甲、技能冷却、地图位置、命中率、敌人护甲等游戏内复杂因素。题目中的“火力值”是为了描述购买方案价值而定义的抽象加权值，不与游戏实际伤害结算完全等价，道具的数量上也略有区别，仅作题目背景使用。



## Problem Description

给定尚博勒当前拥有的金币数 $M$。当前回合，他可以购买两类物品：

- 猎头子弹：每发价格 150 金币，最多可以购买 8 发；
- 陷阱：每个价格 200 金币，最多可以购买 2 个。

设他最终购买的猎头子弹数量为 $h$，陷阱数量为 $t$。由于购买数量必须是非负整数，因此有 $0 \le h \le 8$，$0 \le t \le 2$，并且 $h, t$ 都必须是整数。购买总花费为 $150h + 200t$。尚博勒不能赊账，因此总花费不能超过当前金币数：$150h + 200t \le M$。本题不要求把金币全部花完，也不考虑购买护甲或其他物品。

本回合火力值定义如下：每发猎头子弹提供 55 点火力；每个陷阱提供 30 点火力。因此，购买 $h$ 发猎头子弹和 $t$ 个陷阱时，本回合总火力值为

$$F(h, t) = 55h + 30t.$$

需要特别说明：本题只按照购买数量计算陷阱提供的火力值。也就是说，购买 $t$ 个陷阱，就计算 $30t$ 点火力。现在，请在所有满足预算约束和购买数量限制的方案中，选择使本回合火力值 $F(h, t)$ 最大的方案。

如果存在多组购买方案取得相同的最大火力值，则按照以下规则依次比较：优先选择猎头子弹数量 $h$ 更多的方案；如果仍然并列，则选择陷阱数量 $t$ 更多的方案。

## Input

输入一行一个非负整数 $M$（$0 \le M \le 10^9$），表示尚博勒当前拥有的金币数。

## Output

输出一行三个整数 $F_{\max}, h, t$，三个整数之间用一个空格分隔，依次表示在最优购买方案下取得的最大火力值、购买的猎头子弹数量和陷阱数量。

## Examples

| standard input | standard output |
| --- | --- |
| 500 | 165 3 0 |

### 样例解释

预算 500 时，购买 3 发猎头子弹花费 450，火力值为 165；枚举所有合法的 $h, t$ 后，这一方案的火力值最大，故输出 165 3 0。

# Problem D. 迟到登记

Input file: standard input
Output file: standard output

签到从时刻 0 开始，规定最晚到达时刻为 $t$。给出 $n$ 名同学的到达时刻，求迟到人数和所有迟到同学的迟到分钟数之和。恰好在 $t$ 到达不算迟到。

## Input

第一行 $n, t$（$1 \le n \le 10^5$，$0 \le t \le 10^9$）。第二行 $n$ 个到达时刻 $a_i$（$0 \le a_i \le 10^9$）。

## Output

输出两个整数。

## Examples

| standard input | standard output |
| --- | --- |
| 4 10<br>9 10 13 15 | 2 8 |

### 样例解释

到达时刻 10 恰好达标；时刻 13 和 15 分别迟到 3、5 分钟，因此输出 2 人和 8 分钟。

# Problem E. 简单字符串

Input file: standard input
Output file: standard output

Vasya 对许多人在网络上将大写字母和小写字母混用在同一个单词里感到十分不满。因此，他打算为自己最喜欢的浏览器开发一个扩展程序，将每个单词的字母变为全小写或全大写，以保证一个单词只包含一种字母大小写。并且，整个单词中应尽可能少地更改字母。

例如，单词 `HoUse` 应被替换为 `house`，而单词 `ViP` 应被替换为 `VIP`。如果一个单词中大写和小写字母数量相等，则应将所有字母改为小写。例如，`maTRIx` 应被替换为 `matrix`。

你的任务是对给定的单词，按照上述方法进行处理。

## Input

第一行包含一个单词 $s$，它只包含大写和小写的拉丁字母，长度为 1 到 100。

## Output

输出纠正后的单词 $s$。如果原单词 $s$ 中大写字母数量严格多于小写字母，则将单词全部转换为大写；否则，全部转换为小写。

## Examples

| standard input | standard output |
|---|---|
| HoUse | house |
| ViP | VIP |
| maTRIx | matrix |

### 样例解释

HoUse 中小写字母较多，转为 house；ViP 中大写字母较多，转为 VIP；maTRIx 中大小写各 3 个，按规则转为 matrix。

# Problem F. 纸杯回收站

Input file: standard input
Output file: standard output

活动结束后有 $n$ 个回收箱。第 $i$ 个箱子里有 $a_i$ 个纸杯。每满 $k$ 个纸杯可以兑换一张贴纸；每个箱子单独兑换，不合并余数。求一共能兑换多少张贴纸，以及剩余纸杯总数。

## Input

第一行两个整数 $n, k$（$1 \le n \le 10^5$，$1 \le k \le 10^9$）。第二行 $n$ 个整数 $a_i$（$0 \le a_i \le 10^9$）。

## Output

输出两个整数：贴纸总数、剩余纸杯总数。

## Examples

| standard input | standard output |
| --- | --- |
| 3 4<br>3 8 6 | 3 5 |

### 样例解释

三个箱子分别兑换 $0, 2, 1$ 张贴纸，剩余 $3, 0, 2$ 个纸杯；总计 3 张贴纸、5 个剩余纸杯。各箱余数不能合并兑换。

# Problem G. 小新再次来袭

Input file: standard input
Output file: standard output

给定长度为 $n$ 的整数数组 $a$。你可以执行任意次操作（包括零次）。每次操作都必须**同时**将数组中的每个数 $a_i$ 变为 $|a_i - 2|$。

选择合适的操作次数，使最终数组中某个整数的出现次数尽可能多。输出这个最大次数。

## Input

第一行一个整数 $n$（$1 \le n \le 2 \times 10^5$）。

第二行 $n$ 个整数 $a_i$（$1 \le a_i \le 10^9$）。**只有一组测试数据。**

## Output

输出一个整数，表示最终数组中单个整数出现次数的最大值。

## Examples

| standard input | standard output |
| --- | --- |
| 2<br>1 3 | 2 |
| 4<br>1 1 1 2 | 3 |
| 3<br>6 7 8 | 1 |
| 5<br>1 10 100 1000 100000 | 3 |

### 样例解释

样例 1 操作一次后得到 $[1, 1]$。样例 3 中，6 与 8 虽然都是偶数，但无论操作多少次也不会变成同一个数；7 始终是奇数。因此三个数始终两两不同，答案为 1。

# Problem H. 证书评定

Input file: standard input
Output file: standard output

组委会按照以下优先级评定证书。只执行最先满足的一条规则：

- 若违规标志 $v = 1$，输出 INVALID。
- 否则，若出勤次数 $a < 8$，输出 INELIGIBLE。
- 否则，若成绩 $p = 100$ 且出勤 $a = 10$，输出 PERFECT。
- 否则，若 $p \geq 90$，输出 GOLD。
- 否则，若 $75 \leq p < 90$，输出 SILVER。
- 否则，若 $60 \leq p < 75$，输出 BRONZE。
- 其余情况输出 NONE。

## Input

一行三个整数 $p, a, v$：成绩、出勤次数、违规标志。

$0 \leq p \leq 100$，$0 \leq a \leq 10$，$v \in \{0, 1\}$。

## Output

输出一个大写英文单词，表示证书结果。

## Examples

| standard input | standard output |
| --- | --- |
| 100 10 0 | PERFECT |

### 样例解释

没有违规，出勤 10 次且成绩 100 分，最先满足 PERFECT 条件，输出 PERFECT。

# Problem I. 参加比赛

Input file: standard input
Output file: standard output

三位选手要去参加编程竞赛，比赛共有 $N$ 道题目，他们商定：对于每一道题目，只有两位及以上的选手确认找到解法时才做。

试问，他们一共能做多少题？

## Input

第一行一个整数 $N$（$1 \le N \le 10^5$），表示题目的数量。

第二行到第 $N+1$ 行，每行三个整数（0 或者 1），其中 0 表示这位选手无法解出此题，1 表示这位选手想到了解法。

## Output

输出一行一个整数，表示他们所能做的题目的数量。

## Examples

| standard input | standard output |
| --- | --- |
| 3<br>1 1 0<br>1 1 1<br>1 0 0 | 2 |
| 2<br>1 0 0<br>0 1 1 | 1 |

### 样例解释

样例 1 的前两道题各有至少两位选手会做，因此答案为 2。样例 2 只有第二道题满足条件，因此答案为 1。

# Problem J. 字符串反转

Input file: standard input
Output file: standard output

给定一个长度小于 50、仅由数字字符组成的字符串，请你将整个字符串反转后输出。

在反转之前，先统计原字符串里下面三项，再做反转输出：

1. 字符 `'3'` 出现的次数；
2. 字符 `'6'` 出现的次数；
3. 连续子串 `"18"` 出现的次数（重叠也计算，例如 `"1818"` 包含两处 `18`）。

## Input

输入只有一行，为一个仅包含数字的字符串，字符串长度 len < 50。

## Output

一共四行：

第 1 行：字符 `3` 出现的次数；

第 2 行：字符 `6` 出现的次数；

第 3 行：子串 `18` 出现的次数；

第 4 行：反转之后的字符串。

## Examples

| standard input | standard output |
| --- | --- |
| 361886 | 1<br>2<br>1<br>688163 |

### 样例解释

原串 361886 中，字符 3 出现 1 次、字符 6 出现 2 次、子串 18 出现 1 次；整体反转得到 688163。