

## Bài 1. Văn nghệ

Tên file: vn.cpp hoặc vn.py

Đội văn nghệ xung kích của trường đại học X được cử đi diễn giao lưu ở các huyện trong tỉnh Y. Khi đi đoàn có  $n$  bạn nam và  $m$  bạn nữ. Mỗi tổ sẽ được giao nhiệm vụ biểu diễn tại một địa điểm phục vụ các em nhỏ. Biết rằng số lượng nam và nữ phải chia đều giữa các tổ. Hỏi người quản lý có thể chia số sinh viên ra tối đa được bao nhiêu tổ? Mỗi tổ có bao nhiêu nam và bao nhiêu nữ.

**Dữ liệu vào:**

- Hai số nguyên  $n, m$  cách nhau một khoảng trắng ( $1 < n, m < 10^9$ )

**Dữ liệu ra:**

- Dòng 1 ghi số lượng tổ tối đa có thể chia
- Dòng 2 ghi 2 số  $a, b$  tương ứng là số nam và số nữ của mỗi tổ

Ví dụ:

Vn.inp	Vn.out
48 72	24 2 3

**Bài 2. Trục nhật**Tên file: `tn.cpp` hoặc `tn.py`

Ở một lớp học có  $n$  học sinh. Mỗi bạn đều phải trục nhật và cứ sau một số  $y$  ngày nhất định bạn đó mới phải trục nhật lại. Biết rằng xuất phát điểm ban đầu tất cả sẽ đều trục nhật vào ngày đầu tiên. Bạn hãy giúp lớp trưởng tính xem sau bao nhiêu ngày thì tất cả các bạn mới lại cùng nhau trục nhật và khi đó mỗi bạn đã trục nhật bao nhiêu lần.

**Dữ liệu vào:**

- Dòng đầu chứa số nguyên  $n$  ( $2 \leq n < 100$ )
- Dòng thứ hai chứa  $n$  số nguyên  $y$ . ( $1 \leq y < 100$ )

**Dữ liệu ra:**

- Dòng đầu tiên ghi ra số ngày mà tất cả cùng nhau trục nhật lại.
- Dòng thứ hai chứa  $n$  số là số lần một bạn đã trục nhật cho tới lúc tất cả cùng trục nhật

**Ví dụ:**

<code>tn.inp</code>	<code>tn.out</code>
3	12
2 3 4	6 4 3

**Bài 3. Song nguyên tố****Tên file: CPRIME.PY hoặc CPRIME.CPP**

Số nguyên tố là một số nguyên dương có 2 ước dương là 1 và chính nó.

Ví dụ: 7, 13, 17, ... là những số nguyên tố; còn các số 14, 8, 25, ... không phải là những số nguyên tố.

Số đảo ngược của một số là số được viết theo thứ tự ngược lại của số đó.

Ví dụ: 13 đảo ngược của nó là 31; 145 đảo ngược của nó là 541.

Một số  $n$  được gọi là song nguyên tố nếu như  $n$  là số nguyên tố và số đảo ngược của nó cũng là số nguyên tố.

Ví dụ: 7, 13 là các số song nguyên tố còn các số 8, 41 không phải là các số song nguyên tố.

Cho một số nguyên dương  $n$ . Em hãy kiểm tra xem số  $n$  có phải là song nguyên tố không?

INPUT

- Một dòng duy nhất chứa số nguyên dương  $n$  ( $n \leq 2 \cdot 10^9$ )

OUTPUT

- In ra số **1** nếu đó là số song nguyên tố, và in ra số **0** nếu đó không phải là số song nguyên tố.

Ví dụ:

Ví dụ 1		Ví dụ 2	
CPRIME.INP	CPRIME.out	CPRIME.INP	CPRIME.out
7	1	8	0

## Bài 4. GCD lớn nhất

Tên file: GCDMAX.CPP hoặc GCDMAX.PY

Hãy xem xét tất cả các số nguyên trong phạm vi từ 1 đến  $n$ .

Trong số tất cả các cặp số nguyên phân biệt trong phạm vi này, hãy tìm ước số chung lớn nhất có thể có của các số nguyên trong cặp. Về mặt hình thức, hãy tìm giá trị lớn nhất của  $\gcd(a, b)$  (ước chung lớn nhất của  $a$  và  $b$ ), trong đó  $1 \leq a < b \leq n$ .

Input: GCDMAX.INP:

- ✓ Dòng đầu tiên chứa số nguyên dương  $t$  ( $t \leq 100$ ), số lượng các trường hợp cần xét
- ✓  $t$  dòng tiếp theo, mỗi dòng chứa 1 số nguyên dương  $n$  ( $2 \leq n \leq 10^6$ )

Output: GCDMAX.OUT:

- gồm  $t$  dòng, mỗi dòng chứa số nguyên dương là ước số chung lớn nhất của ước số chung lớn nhất của các cặp số thuộc đoạn  $[1, n]$

Ví dụ:

GCDMAX.INP	GCDMAX.OUT
2	1
3	2
5	

Giải thích:

- ở trường hợp 1:  $n = 3$ , ước chung lớn nhất của mọi cặp  $(a, b) \leq n$  là:  $\gcd(1, 2) = \gcd(1, 3) = \gcd(2, 3) = 1$  nên kết quả là 1.

- ở trường hợp 2:  $n = 5$ , ước chung lớn nhất của mọi cặp  $(a, b) \leq n$  là:  $\gcd(1, 2) = \gcd(1, 3) = \gcd(1, 4) = \gcd(1, 5) = \gcd(2, 3) = \gcd(2, 5) = \gcd(3, 4) = \gcd(3, 5) = \gcd(4, 5) = 1$ ,  $\gcd(2, 4) = 2$  nên kết quả là 2.

## Bài 5. Số có 3 ước

Tên file: TNUM.CPP hoặc TNUM.PY

Một số nguyên dương có đúng 3 ước số nguyên dương khác nhau được gọi là số TNUM. Cho trước một dãy  $N$  ( $1 \leq N \leq 10^5$ ) số nguyên dương, xác định các số đã cho có phải là số TNUM hay không?

**Input:** Cho trong tệp TNUM.INP có cấu trúc như sau:

- Dòng đầu tiên ghi số  $N$
- Dòng tiếp theo ghi  $N$  số nguyên  $a_1, a_2 \dots a_n$  cách nhau bởi một dấu cách ( $1 \leq a_i \leq 10^{12}$ )

**Output:** Ghi ra tệp TNUM.OUT gồm  $N$  dòng, dòng thứ  $i$  ghi YES nếu số thứ  $i$  là số TNUM, ngược lại thì ghi NO.

Ví dụ:

TNUM.INP	TNUM.OUT
3	YES
4 5 6	NO
	NO

**Bài 6. Đi chợ**Tên file: **BMARKET.PY** hoặc **BMARKET.CPP**

Để chuẩn bị cho lễ kết nạp đoàn viên mới, thầy **T** nhờ bạn Tèo đi ra chợ mua bim bim để các bạn liên hoan. Tèo mua được 2 túi bim bim. Túi thứ nhất có  $x$  ( $x \geq 1$ ) gói và túi thứ 2 có  $y$  ( $y \geq 1$ ) gói. Tèo đang rất lo lắng vì cậu đã lỡ làm mất túi thứ nhất, và không nhớ trong túi thứ nhất có bao nhiêu gói bim bim. Cậu chỉ nhớ được tổng số gói không vượt quá  $n$  và chia hết cho  $k$ .

Thầy **T** nhờ các bạn hãy giúp Tèo tính có thể có bao nhiêu gói bim bim có trong túi thứ nhất để thầy mua bổ sung. Nếu có nhiều khả năng thì in ra tất cả theo thứ tự tăng dần.

Nếu không thể biết số gói bim bim thì in ra -1.

**INPUT**

- Một dòng duy nhất chứa 3 số nguyên  $y, k, n$ . ( $1 \leq y, k, n \leq 10^9, n/k \leq 10^5$ ).

**OUTPUT**

- In ra kết quả của bài toán.

Ví dụ:

BMARKET.INP	BMARKET.OUT
10 1 10	-1
10 6 40	2 8 14 20 26

## Bài 7. Đếm số 0 bên phải

Tên file: BZERO.PY hoặc BZERO.CPP

Cho một số nguyên  $n$ . Hãy đếm xem trong kết quả của số  $n!$  ( $n$  giai thừa) có bao nhiêu chữ số 0 liên tiếp tính từ hàng đơn vị (hay bao nhiêu số 0 liên tiếp bên phải ngoài cùng).

**INPUT:** Là số nguyên  $n$  ( $1 \leq n \leq 1.000$ )

**OUTPUT:** Là số lượng chữ số 0 liên tiếp tính từ hàng đơn vị của  $n!$ .

Ví dụ:

INPUT	OUTPUT
4	0
8	1
20	4

**Bài 8. Số siêu nguyên tố**

Tên file: SPRIME.CPP hoặc SPRIME.PY

Một số tự nhiên  $N$  được gọi là siêu nguyên tố nếu bản thân nó là một số nguyên tố và tất cả các số thu được bằng cách xóa lần lượt các chữ số bên phải của nó đều là số nguyên tố.

Ví dụ:

Số 317 là một số siêu nguyên tố vì: 317 là 1 số nguyên tố

Xóa 1 chữ số bên phải: 31 là 1 số nguyên tố

Xóa 2 chữ số bên phải: 3 là 1 số nguyên tố

Cho 2 số nguyên  $a, b$ . Hãy liệt kê tất cả các số siêu nguyên tố thuộc đoạn  $[a, b]$ .

**Input:** Tập SPRIME.INP gồm một dòng ghi 2 số nguyên dương  $a, b$  ( $0 < a, b \leq 10^7$ )

**Output:** Tập SPRIME.OUT liệt kê theo thứ tự tăng các số siêu nguyên tố thuộc đoạn  $[a, b]$ , mỗi số trên một dòng, hoặc ghi “NO” trong trường hợp không có số nào thuộc đoạn đó.

Ví dụ

SPRIME.INP	SPRIME.OUT
3 57	3 5 7 23 29 31 37 53



**Bài 9. MUA CỎ**Tên file: **FORAGE.PY** hoặc **FORAGE.CPP**

Rút kinh nghiệm trong việc chăm sóc đàn bò của mình từ năm ngoái, bác nông dân John mua quá ít loại cỏ khác nhau cho chúng ăn dẫn đến sữa của các cô bò không có được chất lượng thực sự tốt.

Năm nay, bác quyết định sẽ mua  $n$  loại cỏ khác nhau về bồi dưỡng cho đàn bò trước kỳ vắt sữa. Bác John mang theo  $p$  dollars sang nông trại hàng xóm là nông dân Tom. Bác Tom có trồng  $n$  loại cỏ, bác Tom đồng ý bán cho bác John với giá như sau:

- Loại cỏ thứ 1 bán với giá  $k$  dollars
- Loại cỏ thứ 2 bán với giá  $2k$  dollars
- Loại cỏ thứ 3 bán với giá  $3k$  dollars
- ....
- Loại cỏ thứ  $n$  bán với giá  $nk$  dollars

Em hãy cho biết, với  $p$  dollars của mình mang theo liệu nông dân John có thể mua được  $n$  loại cỏ hay không.

**INPUT**

- Một dòng duy nhất chứa 3 số nguyên  $p, n, k$  ( $1 \leq p, n \leq 2 \cdot 10^8, 1 \leq k \leq 20$ )

**OUTPUT**

- Nếu nông dân có đủ tiền mua được  $n$  loại cỏ của bác Tom thì đưa ra thông báo là "YES".
- Trong trường hợp không đủ tiền thì:
  - Dòng 1: thông báo là "NO"
  - Dòng 2: thông báo số tiền mà bác nông dân John phải nợ lại bác nông dân Tom

**Ví dụ:**

INPUT	OUTPUT	INPUT	OUTPUT
17 4 3	NO 13	33 4 3	YES

**Bài 10. Biểu diễn  $N!$** Tên file: `analyse.py` hoặc `analyse.cpp`

Giai thừa  $N$ , ký hiệu  $N!$  là tích tất cả các số nguyên từ 1 đến  $N$ . Giai thừa  $N$  tăng rất nhanh, ví dụ  $5!=120$ ,  $10!=3628800$ . Một cách để xác định số lớn như vậy, người ta chỉ ra số lần xuất hiện các số nguyên tố trong phân tích của nó ra thừa số nguyên tố. Ví dụ, số 825 có thể xác định bởi dãy (0 1 2 0 1) có nghĩa là  $825 = 2^0 \cdot 3^1 \cdot 5^2 \cdot 7^0 \cdot 11^1$ .

Cho một số nguyên dương  $N \leq 1000$ . Hãy tìm cách biểu diễn số  $N!$  dưới dạng số lần xuất hiện các số nguyên tố trong phân tích  $n!$  ra các thừa số nguyên tố.

**Dữ liệu** vào từ file `ANALYSE.INP` gồm nhiều dòng, mỗi dòng chứa 1 số nguyên  $N$  ( $2 \leq N \leq 1000$ ).

**Kết quả** Ghi ra file `ANALYSE.OUT` mỗi dòng tương ứng với 1 dòng trong file dữ liệu vào là dãy số thể hiện biểu diễn dưới dạng phân tích thành số nguyên tố của  $n!$  (phần tử cuối cùng của dãy phải là số dương).

*Ví dụ:*

ANALYSE.INP	ANALYSE.OUT
10	8 4 2 1
3	1 1