

# BÀI TẬP TIN HỌC ÔN LUYỆN OLIMPIC





Làm trai phải lạ ở trên đời,  
Há để càn khôn tự chuyển dời

## MỤC LỤC

|                              |    |
|------------------------------|----|
| ĐỒNG HỒ BÁO THỨC .....       | 5  |
| TỔNG NHỎ NHẤT .....          | 6  |
| ĐẦU GIÁ .....                | 7  |
| DÂY SỐ TRUNG BÌNH CỘNG ..... | 8  |
| THU NHẬP BÓNG .....          | 9  |
| SỐ Ô ĐEN TRÊN BÀN CỜ .....   | 10 |
| SỐ ĐẸP .....                 | 11 |
| XÂU KÝ TỰ NGOẶC.....         | 12 |
| KHÔI PHỤC NGOẶC.....         | 13 |
| LỊCH BYTELAND .....          | 14 |
| HỘP KEO.....                 | 15 |
| BÀN CỜ .....                 | 16 |
| MÃ HÓA ĐA LỚP.....           | 17 |
| CÁC ĐỒNG XU .....            | 19 |
| DÂY LIÊN TIẾP.....           | 20 |
| GIẢI MÃ SỐ .....             | 21 |
| KHOẢNG CÁCH SỐ .....         | 22 |
| BẦU CỬ .....                 | 23 |
| EQUATION .....               | 24 |
| XÂY DỰNG HÀNG RÀO .....      | 25 |
| TẶNG HOA .....               | 26 |
| ĐẶT QUẦY PHỤC VỤ.....        | 27 |
| THỨ SÁU NGÀY 13.....         | 28 |
| HÀM SỐ.....                  | 29 |
| QUÀ TẶNG .....               | 30 |
| BÀI TẬP VỀ NHÀ.....          | 31 |
| SỐ NGUYÊN TỐ.....            | 32 |
| THẦN TƯỢNG.....              | 33 |
| TỐI GIẢN PHÂN SỐ .....       | 34 |
| TÁO QUÂN .....               | 35 |
| HIỆN SỐ BẰNG ĐÈN LED.....    | 36 |
| THÀNH PHỐ MAY MẮN .....      | 37 |
| LŨY THỪA CỦA 2.....          | 38 |

|                                     |    |
|-------------------------------------|----|
| CÁC THỎI NAM CHÂM.....              | 39 |
| TẦN SỐ XUẤT HIỆN NHIỀU NHẤT .....   | 40 |
| HỖN HỢP .....                       | 41 |
| MODULO.....                         | 42 |
| TIỀN .....                          | 43 |
| KHẢM TRANH .....                    | 44 |
| NTFS .....                          | 45 |
| SỐ THÂN THIỆN.....                  | 46 |
| NUMPOS .....                        | 47 |
| TRÒ CHƠI VỚI DÂY SỐ .....           | 48 |
| CON SỐ BÍ ẨN.....                   | 49 |
| LUYỆN TẬP DỰ THI HỌC SINH GIỎI..... | 50 |
| SỐ ĐỐI XỨNG .....                   | 51 |
| TRÔNG XE.....                       | 52 |
| ĐỒ XE .....                         | 53 |
| DÒ TÌM MẬT KHẨU .....               | 54 |
| ĐA GIÁC.....                        | 55 |
| ROBOT DI CHUYỂN .....               | 56 |
| QUAY BẢNG .....                     | 57 |
| THAM QUAN THÀNH PHỐ .....           | 58 |
| DÂY SỐ VÔ HẠN.....                  | 59 |
| DÂY SỐ ĐẢO NGƯỢC .....              | 60 |
| ỐC SÊN .....                        | 61 |
| TỔNG .....                          | 62 |
| MUA VÉ XE .....                     | 63 |
| XÂY THÁP.....                       | 64 |
| QUAY XÂU KÝ TỰ .....                | 65 |
| DÃ NGOẠI .....                      | 66 |
| SỐ SINH ĐÔI .....                   | 67 |
| TÔ MÀU .....                        | 68 |
| DÂY DẪN.....                        | 69 |
| TỪ DÀI NHẤT .....                   | 70 |

## ĐỒNG HỒ BÁO THỨC

An rất mê đồng hồ loại hiển thị bằng số điện tử sử dụng 7 đèn LED để biểu diễn các số từ 0 đến 9 như hình bên dưới.



An thường mê mê chinh chiếc đồng hồ xinh xắn của mình để đặt báo thức vào mỗi tối. Đêm qua cô bé đã mơ về chiếc đồng hồ yêu quý của mình, nhưng không may khi tỉnh dậy lại quên thời gian đã hiển thị trên đồng hồ mà chỉ còn nhớ số vạch LED hiển thị trên đồng hồ.



Thời gian hiển thị trên đồng hồ của An được biểu diễn bởi 4 chữ số, 2 chữ số cho giờ và 2 chữ số cho phút, và được thiết lập hiển thị ở chế độ 24h. Ví dụ hình bên biểu diễn cho 9h30 (có số 0 ở đầu).

**Dữ liệu:** vào từ tập tin văn bản **ALARM.INP** số nguyên  $n (0 \leq n \leq 30)$  là số vạch hiển thị trên đồng hồ.

**Kết quả:** xuất ra tập tin văn bản **ALARM.OUT** 5 kí tự hiển thị theo định dạng “hh:mm” là thời gian hợp lệ hiển thị trên đồng hồ ( $0 \leq hh < 24; 0 \leq mm < 60$ ).

- Nếu có nhiều kết quả thì in ra kết quả bất kỳ
- Nếu không tìm được kết quả thì in ra thông báo “Impossible”

**Ví dụ:**

| ALARM.INP |
|-----------|
| 23        |

| ALARM.OUT |
|-----------|
| 09:30     |

| ALARM.INP |
|-----------|
| 28        |

| ALARM.OUT  |
|------------|
| Impossible |

## TỔNG NHỎ NHẤT

Cho hai dãy số nguyên  $A = (a_1, a_2, \dots, a_m)$  và  $B = (b_1, b_2, \dots, b_n)$  hãy tìm một phần tử  $a_i$  trong dãy  $A$  và một phần tử  $b_j$  trong dãy  $B$  có  $|a_i + b_j|$  là nhỏ nhất có thể ( $1 \leq i \leq m; 1 \leq j \leq n$ ).

**Dữ liệu:** vào từ tập tin văn bản **ASUMMIN.INP**

- Dòng 1 chứa hai số nguyên dương  $m, n \leq 10^5$
- Dòng 2 chứa  $m$  số nguyên  $a_1, a_2, \dots, a_m (\forall i: |a_i| < 2^{31})$
- Dòng 3 chứa  $n$  số nguyên  $b_1, b_2, \dots, b_n (\forall j: |b_j| < 2^{31})$

**Kết quả:** ghi ra tập tin văn bản **ASUMMIN.OUT** hai chỉ số  $i$  và  $j$  của hai phần tử tương ứng tìm được.

**Ví dụ**

| ASUMMIN.INP   | ASUMMIN.OUT |
|---------------|-------------|
| 4 5           | 2 4         |
| 1 8 2 9       |             |
| -5 -6 3 -7 -4 |             |

Giải thích:  $|a_2 + b_4| = |8 + (-7)| = 1$

## ĐẤU GIÁ

Sở giao thông Hà Nội quyết định bán đấu giá các biển số xe đẹp để lấy tiền ủng hộ đồng bào lũ lụt miền Trung. Một biển số xe được gọi là đẹp nếu nó thỏa mãn các điều kiện sau:

- Là một số nguyên dương  $T$  mà  $A \leq T \leq B$  trong đó  $A, B$  là hai số nguyên dương cho trước;
- $T$  là một số nguyên tố;
- $T$  là một số đối xứng (đọc  $T$  từ trái qua phải thu được kết quả giống như đọc  $T$  từ phải qua trái).

**Yêu cầu:** Cho hai số nguyên dương  $A$  và  $B$ , hãy tìm số lượng các biển số xe đẹp.

**Dữ liệu:** vào từ tập tin văn bản **AUCTION.INP** chứa hai số nguyên  $A, B$  ( $10^4 \leq A < B \leq 10^5$ )

**Kết quả:** ghi ra tập tin văn bản **AUCTION.OUT** số lượng biển số xe đẹp tìm được.

**Ví dụ:**

| AUCTION.INP | AUCTION.OUT |
|-------------|-------------|
| 11111 22222 | 23          |

## DÃY SỐ TRUNG BÌNH CỘNG

Mirko vừa nghĩ ra một cách luyện tập các phép toán số học mà cậu cho là thú vị như sau: trước tiên Mirko viết một dãy gồm các số  $A$ . Sau đó, bên dưới mỗi phần tử của dãy số đầu tiên, Mirko viết một con số là giá trị trung bình cộng các phần tử của  $A$  tính từ đầu dãy đến vị trí hiện tại.

Chẳng hạn, dãy  $A$  có giá trị 1,3,2,6,8 thì giá trị của dãy  $B$  sẽ là

$$\left\{ \frac{1}{1}, \frac{1+3}{2}, \frac{1+3+2}{3}, \frac{1+3+2+6}{4}, \frac{1+3+2+6+8}{5} \right\} = \{1, 2, 2, 3, 4\}$$

**Yêu cầu:** cho giá trị các phần tử của dãy  $B$ . Hãy tìm dãy  $A$  ban đầu phù hợp với cách tính của Mirko.

**Dữ liệu:** vào từ tập tin văn bản **AVGSEQ.INP**

- Dòng đầu tiên chứa số nguyên dương  $n$  ( $1 \leq n \leq 100$ )
- Dòng tiếp theo chứa dãy số nguyên  $b_1, b_2, \dots, b_n$  ( $1 \leq b_i \leq 10^9$ )

**Kết quả:** ghi ra tập tin văn bản **AVGSEQ.OUT** gồm 1 dòng chứa dãy số  $a_1, a_2, \dots, a_n$  tìm được. Dữ liệu vào được cho đảm bảo dãy  $A$  tìm được là dãy số nguyên và có giá trị không vượt quá  $10^9$ .

**Ví dụ:**

| AVGSEQ.INP |
|------------|
| 4          |
| 3 2 3 5    |

| AVGSEQ.OUT |
|------------|
| 3 1 5 11   |



## THU NHẬT BÓNG

Trong một trò chơi vận động, đội chơi sẽ cử người thực hiện một số lượt để lấy hết tất cả  $n$  quả bóng của ban tổ chức theo luật chơi. Lượt thứ nhất, người chơi chỉ được lấy tối đa 1 quả, lượt thứ 2 chỉ được lấy tối đa  $1 + 2 = 3$  quả, lượt thứ 3 chỉ được lấy tối đa  $1 + 2 + 3 = 6$  quả. Tổng quát, lượt thứ  $k$  người chơi chỉ được lấy tối đa  $1 + 2 + \dots + (k - 1) + k$  quả bóng.

**Yêu cầu:** cho số nguyên dương  $n$ , hỏi đội chơi cần thực hiện ít nhất là bao nhiêu lượt để lấy hết tất cả  $n$  quả bóng của ban tổ chức.

**Dữ liệu:** vào từ tập tin văn bản **BALLOONS.INP** số nguyên dương  $n (n \leq 10^6)$ .

**Kết quả:** ghi ra tập tin văn bản **BALLOONS.OUT** số lượt ít nhất mà đội chơi cần thực hiện để lấy hết tất cả  $n$  quả bóng.

**Ví dụ:**

| BALLOONS.INP |
|--------------|
| 16           |

| BALLOONS.OUT |
|--------------|
| 4            |

## SỐ Ô ĐEN TRÊN BÀN CỜ

Một bàn cờ gồm  $n$  dòng,  $m$  cột, các dòng và cột được đánh thứ tự từ 1 (hình minh họa). Mỗi ô được sơn đen hoặc trắng đan xen nhau tương tự bàn cờ vua. Ô ở dòng  $r$  ( $1 \leq r \leq n$ ) cột  $c$  ( $1 \leq c \leq m$ ) của bàn cờ được sơn màu đen. Hãy xác định có bao nhiêu ô được sơn màu đen.

|   | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 |   |   |   |   |   |
| 2 |   |   |   |   |   |
| 3 |   |   |   |   |   |
| 4 |   |   |   |   |   |

**Dữ liệu:** vào từ tập tin văn bản **BCELLS.INP** chứa bốn số nguyên  $n, m, r, c$  ( $1 \leq n, m \leq 10^6$ ) trên cùng dòng và cách nhau khoảng trắng

**Kết quả:** ghi ra tập tin văn bản **BCELLS.OUT** số ô được sơn màu đen.

**Ví dụ:**

| BCELLS.INP |
|------------|
| 4 5 3 4    |

| BCELLS.OUT |
|------------|
| 10         |

## SỐ ĐẸP

Một số nguyên dương được gọi là số đẹp nếu tổng các chữ số của nó (trong hệ thập phân) chia hết cho số chữ số. Các số được xét không chứa số 0 không có nghĩa. Ví dụ, 15 là một số đẹp vì  $1+5$  chia hết cho 2.

Các số đẹp được đánh số từ 1 trở đi theo thứ tự tăng dần của giá trị.

**Yêu cầu:** Cho số nguyên dương  $n$  ( $1 \leq n \leq 100\,000$ ). Hãy tìm số đẹp thứ  $n$ .

**Dữ liệu:** Vào từ file văn bản **BEAUTY.INP** gồm nhiều tests, mỗi test ghi trên một dòng chứa một số nguyên  $n$ .

**Kết quả:** Đưa ra file văn bản **BEAUTY.OUT**, kết quả mỗi test đưa ra trên một dòng.

**Ví dụ:**

| BEAUTY.INP |
|------------|
| 1          |
| 15         |

| BEAUTY.OUT |
|------------|
| 1          |
| 20         |

## XÂU KÝ TỰ NGOẶC

Xét xâu chỉ chứa các ký tự ngoặc tròn  $(, )$ , ngoặc vuông  $[, ]$  và ngoặc nhọn  $\{, \}$ . Để ngắn gọn, ta gọi nó là xâu ngoặc.

Định nghĩa xâu ngoặc đúng:

- Xâu rỗng được coi là xâu ngoặc đúng,
- Nếu  $a$  là xâu ngoặc đúng thì  $(a)$ ,  $[a]$ ,  $\{a\}$  cũng là các xâu ngoặc đúng,
- Nếu  $a$  và  $b$  là các xâu ngoặc đúng thì  $ab$  cũng là xâu ngoặc đúng.

Cho xâu  $S$  độ dài  $n$ . Xâu  $s_k s_{k+1} s_{k+2} \dots s_n s_1 s_2 \dots s_{k-1}$  được gọi là xâu đầy vòng của  $S$ . Bản thân  $S$  cũng là một xâu đầy vòng của  $S$ .

**Yêu cầu:** Cho xâu ngoặc  $S$  có độ dài không quá 1000. Hãy xác định có tồn tại một xâu đầy vòng của  $S$  là xâu ngoặc đúng hay không và đưa ra câu trả lời *Yes* hoặc *No*.

**Dữ liệu:** Vào từ file văn bản **BRACKETS.INP** gồm một dòng chứa xâu  $S$ .

**Kết quả:** Đưa ra file văn bản **BRACKETS.OUT** câu trả lời *Yes* hoặc *No*.

**Ví dụ:**

| BRACKETS.INP |
|--------------|
| } } { }      |

| BRACKETS.OUT |
|--------------|
| Yes          |

## KHÔI PHỤC NGOẶC

Một dãy dấu ngoặc hợp lệ là một dãy các ký tự "(" và ")" được định nghĩa như sau:

- Dãy rỗng (không có ký tự nào) là một dãy dấu ngoặc hợp lệ
- Nếu A là một dãy dấu ngoặc hợp lệ thì (A) là dãy dấu ngoặc hợp lệ. Dấu ngoặc mở và dấu ngoặc đóng hai bên dãy A được gọi là tương ứng với nhau
- Nếu A và B là hai dãy dấu ngoặc hợp lệ thì AB là dãy dấu ngoặc hợp lệ.

Ví dụ: ((( )))(( ))( ) là một dãy dấu ngoặc hợp lệ. các dấu mở ngoặc ở các vị trí: 1, 2, 3, 7, 8, 11, 13 tương ứng lần lượt với các dấu đóng ngoặc ở các vị trí: 6, 5, 4, 10, 9, 12, 14.

Ban đầu có một dãy dấu ngoặc hợp lệ, người ta viết vào dưới mỗi dấu ngoặc mở một số là số dấu ngoặc (cả đóng và mở) nằm giữa dấu ngoặc mở đó và dấu ngoặc đóng tương ứng:

(   (   (   )   )   )   (   (   )   )   (   )   (   )  
4   2   0                      2   0                      0                      0

Sau đó xóa đi dãy ngoặc.

**Yêu cầu:** cho biết dãy số còn lại, hãy khôi phục lại dãy ngoặc ban đầu

**Dữ liệu:** vào từ tập tin văn bản **BRACKETS.INP**

- Dòng 1: Ghi số  $n$  là số phần tử của dãy số còn lại ( $n \leq 10^5$ )
- Dòng 2: Ghi lần lượt các số trong dãy

**Kết quả:** xuất ra tập tin văn bản **BRACKETS.OUT** dãy dấu ngoặc khôi phục được.

**Ví dụ:**

| BRACKETS.INP  | BRACKETS.OUT    |
|---------------|-----------------|
| 7             | (( ))(( ))(( )) |
| 4 2 0 2 0 0 0 |                 |

## LỊCH BYTELAND

Xứ sở *Byteland* sử dụng lịch tương tự với dương lịch của Trái Đất. Mỗi năm ở *Byteland* có  $n$  tháng bắt đầu từ tháng 1. Tháng chẵn có 30 ngày và tháng lẻ có 31 ngày, như vậy tháng 1 có 31 ngày, tháng 2 có 30, tháng 3 có 31 ngày, ...

Mỗi tuần ở *Byteland* có 7 ngày: thứ 1, thứ 2, thứ 3, thứ 4, thứ 5, thứ 6 và thứ 7.

**Yêu cầu:** cho biết ngày đầu năm mới ở *Byteland* là thứ  $w$  ( $1 \leq w \leq 7$ ) trong tuần. Hãy xác định ngày  $d$  tháng  $m$  ở *Byteland* là thứ mấy trong tuần.

**Dữ liệu:** vào từ tập tin văn bản **CALENDAR.INP**

- Dòng đầu tiên chứa số nguyên dương  $w$ .
- Dòng tiếp theo chứa hai số nguyên  $d, m$  ( $1 \leq d \leq 31; 1 \leq m \leq 10^9$ ) cách nhau khoảng trắng là một ngày hợp lệ ở *NDPlanet*.

**Kết quả:** ghi ra tập tin văn bản **CALENDAR.OUT** một số nguyên là kết quả tìm được.

**Ví dụ:**

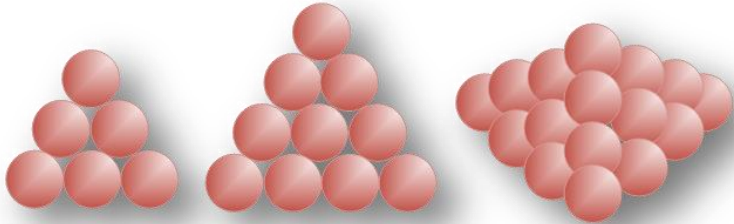
| CALENDAR.INP |
|--------------|
| 3            |
| 10 1         |

| CALENDAR.OUT |
|--------------|
| 5            |

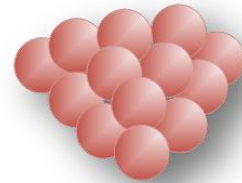
## HỘP KẸO

Đến mừng sinh nhật, một người bạn mang gói quà đến, đặt cẩn thận giữa bàn. Khi mở ra mọi người đều trầm trồ thán phục: quá ngon và đẹp. Các viên bi sô cô la được xếp thành một hình chóp tứ diện đều. Mọi người đều đồng ý với ý kiến của bạn chủ nhà: *”Còn hấp dẫn hơn cả kim tự tháp Giza!”*. Đáy của tứ diện đều là một tam giác đều lấp ráp từ các viên sô cô la. Mỗi lớp tiếp theo ở trên cũng là một tam giác đều với cạnh ít hơn một viên.

Bỗng một bạn nam, nổi tiếng là tinh nghịch, kêu lên *”Ai đấy? Sao đến muộn thế!”*. Trong khi mọi người quay ra phía cửa chờ đợi, bạn nam này bốc và ăn hết



các viên ở một số lớp trên của tứ diện. Trò tinh quái này lập tức bị phát hiện. Mọi người lao vào *”tay non faireplayer”* để trừng phạt. Kết quả là bàn tiệc bị chao đảo, khối kẹo (bây giờ chỉ còn là hình chóp cụt) tóe ra, các viên kẹo lăn lung tung khắp nơi. Tất cả tập trung vào việc đi tìm và nhặt kẹo. Số viên kẹo tìm lại được là  $n$ . Không ai biết là có còn sót viên nào hay không. Mọi quyết định sẽ kiểm tra bằng cách xếp chúng lại thành hình chóp cụt tứ diện đều. Nếu xếp được – có thể yên tâm là đã nhặt đủ.



**Yêu cầu:** Cho số nguyên  $n$  ( $0 \leq n \leq 10^{17}$ ). Hãy xác định xem có thể xếp được một hình chóp cụt tứ diện đều hay không.

**Dữ liệu:** Vào từ file văn bản **CANDIES.INP** gồm một dòng chứa số nguyên  $n$ .

**Kết quả:** Đưa ra file văn bản **CANDIES.OUT** thông báo **YES** hoặc **NO**.

**Ví dụ:**

| CANDIES.INP |
|-------------|
| 16          |

| CANDIES.OUT |
|-------------|
| YES         |

## BÀN CỜ

Một bàn cờ gồm  $n \times m$  ô, mỗi ô vuông được sơn màu đen hoặc trắng đan xen nhau. Ô ở góc dưới bên trái bàn cờ được sơn màu đen. Hãy xác định có bao nhiêu ô được sơn màu đen trên bàn cờ.

**Dữ liệu:** vào từ tập tin văn bản **CHESSBOARD.INP** 2 số nguyên dương  $n, m (n, m \leq 30000)$

**Kết quả:** ghi ra tập tin văn bản **CHESSBOARD.OUT** số ô vuông được sơn màu đen

**Ví dụ:**

| CHESSBOARD.INP |
|----------------|
| 3 4            |

| CHESSBOARD.OUT |
|----------------|
| 6              |



## MÃ HÓA ĐA LỚP

Trong mật mã học, một ngành toán học ứng dụng cho công nghệ thông tin, mã hóa là phương pháp để biến thông tin (phim ảnh, văn bản, hình ảnh...) từ định dạng bình thường sang dạng thông tin không thể hiểu được nếu không có phương tiện giải mã. Giải mã là phương pháp để đưa từ dạng thông tin đã được mã hóa về dạng thông tin ban đầu, quá trình ngược của mã hóa. Một hệ thống mã hóa bao gồm các thành phần:

- Thông tin trước khi mã hóa, kí hiệu là  $P$
- Thông tin sau khi mã hóa, kí hiệu là  $C$
- Chìa khóa, kí hiệu là  $K$
- Phương pháp mã hóa/giải mã, kí hiệu là  $E/D$ .

Quá trình mã hóa được tiến hành bằng cách áp dụng hàm toán học  $E$  lên thông tin  $P$ , vốn được biểu diễn dưới dạng số, để trở thành thông tin đã mã hóa  $C$ . Quá trình giải mã được tiến hành ngược lại: áp dụng hàm  $D$  lên thông tin  $C$  để được thông tin đã giải mã  $P$ . Trong lúc học môn này ở trường đại học, Nam nghĩ ra phương pháp mã hóa mới như sau :

0. Mã hóa văn bản gồm  $n$  kí tự la tinh  $[A-Z]$ , khoảng trắng, dấu chấm, dấu phẩy, dấu hai chấm, dấu chấm phẩy, dấu nháy kép.
1. Mỗi kí tự của văn bản được biểu diễn dưới dạng thập phân như sau:  
 $\langle \text{khoảng trắng} \rangle = 0$ ,  $A=1$ ,  $B=2$ ,  $C=3$ , ...,  $Y=25$ ,  $Z=26$ ,  $\langle \text{dấu chấm} \rangle = 27$ ,  $\langle \text{dấu phẩy} \rangle = 28$ ,  
 $\langle \text{dấu hai chấm} \rangle = 29$ ,  $\langle \text{dấu chấm phẩy} \rangle = 30$ ,  $\langle \text{dấu nháy kép} \rangle = 31$ .
2. Mỗi số dạng thập phân lại được biểu diễn một nhóm 5 số ở dạng nhị phân ,ví dụ :  
 $A=1=00001$ ,  $C=3=00011$ ,  $M=13=01101$ .
3. Dãy số nhị phân nhận được do các số kết hợp với nhau được viết lại vào mảng hai chiều kích thước  $5 \times n$  theo chiều từ trên xuống dưới và từ trái sang phải, bắt đầu từ góc trên trái (như hình bên dưới). Ví dụ : Văn bản ACM sẽ biểu diễn thành dãy 000010001101101 với độ dài 15 và được điền vào mảng hai chiều  $5 \times 3$  như sau:

|   |   |   |
|---|---|---|
| 0 | 0 | 0 |
| ↓ | ↓ | ↓ |
| 0 | 0 | 1 |
| ↓ | ↓ | ↓ |
| 0 | 0 | 1 |
| ↓ | ↓ | ↓ |
| 0 | 1 | 0 |
| ↓ | ↓ | ↓ |
| 1 | 1 | 1 |

5. Từ các bit của mảng ta viết theo từng dòng từ trái sang phải, từ trên xuống dưới, bắt đầu từ góc trên trái, ta sẽ được dãy số nhị phân mới. Ví dụ trên sẽ có dãy nhị phân mới như sau 000001001010111. Sau đó ta nhóm dãy nhị phân mới nhận được thành từng nhóm 5 bit, bắt đầu từ trái sang phải của dãy nhị phân. Với mỗi nhóm 5 bit ở hệ đếm nhị phân ta chuyển về hệ đếm thập phân và xem chúng tương ứng với kí tự nào ở mục 1 thì lần lượt viết ra theo thứ tự của nhóm 5 bit trong dãy. Chuỗi kí tự nhận được chính là đoạn mã hóa. Ví dụ trên sẽ có đoạn mã hóa như sau:

$$00000_2 = 0_{10} = \text{<khoảng trắng>}, 10010_2 = 18_{10} = \text{R}, 10111_2 = 23_{10} = \text{W}.$$

**Yêu cầu:** Hãy thực hiện phương pháp mã hóa này.

**Dữ liệu:** vào từ tập tin văn bản **CODE.INP** gồm một dòng chứa văn bản gồm các kí tự la tinh, khoảng trắng, dấu chấm, dấu phẩy, dấu hai chấm, dấu chấm phẩy, dấu nháy kép. Văn bản dài không quá  $10^4$  kí tự.

**Kết quả:** ghi ra tập tin văn bản **CODE.OUT** ghi chuỗi kí tự nhận được từ kết quả mã hóa.

*Các tập tin dữ liệu mẫu:*

| CODE.INP |
|----------|
| ACM      |

| CODE.OUT |
|----------|
| RW       |

## CÁC ĐỒNG XU

Steve ngồi bên cửa sổ rất lâu quan sát lũ quạ. Chúng là loài chim thông minh, rất thích các vật lóng lánh và hay tha những thứ này về tổ. Hôm nay, không biết kiếm được ở đâu, chúng tha về các đồng xu. Có tất cả  $n$  con quạ. Con quạ thứ  $i$  đã mang về  $a_i$  đồng xu. Nhìn vào vị trí của tổ trên cành Steve hiểu rằng nếu có  $b_i$  thì tổ của quạ thứ  $i$  sẽ bị lật, rơi xuống đất và Steve sẽ nhặt được hết xu trong tổ. Steve đang có trong túi  $m$  đồng xu và có tài ném đầu trúng đó. Bây giờ chính là lúc cái tài lẻ này phát huy tác dụng.

**Yêu cầu:** Hãy xác định số tiền tối đa mà Steve sẽ có được.

**Dữ liệu:** Vào từ file văn bản **COINS.INP**:

- Dòng đầu tiên chứa 2 số nguyên  $n$  và  $m$  ( $1 \leq n \leq 1\,000$ ,  $0 \leq m \leq 1\,000$ ),
- Dòng thứ 2 chứa  $n$  số  $a_1, a_2, \dots, a_n$  ( $0 \leq a_i \leq 1\,000$ ,  $i = 1 \div n$ ),
- Dòng thứ 3 chứa  $n$  số  $b_1, b_2, \dots, b_n$  ( $a_i < b_i \leq 1\,000$ ,  $i = 1 \div n$ ).

**Kết quả:** Đưa ra file văn bản **COINS.OUT** một số nguyên – số tiền tối đa mà Steve sẽ có.

**Ví dụ:**

| COINS.INP |
|-----------|
| 2 3       |
| 1 2       |
| 4 6       |

| COINS.OUT |
|-----------|
| 6         |

## DÃY LIÊN TIẾP

Gọi  $S(n)$  là số chữ số của số nguyên dương  $n$ . Ta viết một dãy các số nguyên liên tiếp bắt đầu từ  $m$ , nghĩa là ta có dãy  $m, m + 1, m + 2, \dots$ . Biết rằng, khi thêm số nguyên  $n$  vào dãy thì ta sẽ phải tốn một chi phí là  $S(n) * k$ . Chẳng hạn với  $m = 7$  và  $k = 2$  thì dãy gồm 6 phần tử:

7,8,9,10,11,12 sẽ tốn chi phí 18.

**Yêu cầu:** cho biết chiều dài lớn nhất của dãy gồm các số nguyên liên tiếp bắt đầu từ  $m$  và với chi phí ban đầu  $w$ .

**Dữ liệu:** vào từ tập tin văn bản **CONSEQ.INP** gồm 3 số nguyên dương  $w(1 \leq w \leq 10^{16})$ ,  $m(1 \leq m \leq 10^{16})$ ,  $k(1 \leq k \leq 10^9)$

**Kết quả:** xuất ra tập tin văn bản **CONSEQ.OUT** gồm 1 số nguyên dương là chiều dài lớn nhất của dãy tạo được.

**Ví dụ:**

| CONSEQ.INP |
|------------|
| 9 1 1      |

| CONSEQ.OUT |
|------------|
| 9          |

## GIẢI MÃ SỐ

Giả sử các chữ số từ 1 đến 9 được mã hoá dưới dạng một chuỗi chỉ gồm các ký tự a, b và c theo quy tắc sau:

| Chữ số | 1        | 2        | 3         | 4          | 5          | 6          | 7          | 8          | 9          |
|--------|----------|----------|-----------|------------|------------|------------|------------|------------|------------|
| Mã     | <i>a</i> | <i>b</i> | <i>cc</i> | <i>bbc</i> | <i>cbc</i> | <i>abc</i> | <i>bac</i> | <i>aac</i> | <i>cac</i> |

Ví dụ số 132 sẽ mã hóa thành ac**cb**.

**Yêu cầu:** từ xâu ký tự cho trước, hãy giải mã để tìm số nguyên dương tương ứng.

**Dữ liệu:** Vào từ file văn bản **DECODE.INP** xâu chứa không quá 100 ký tự a,b, c.

**Kết quả:** Đưa ra file văn bản **DECODE.OUT** số tương ứng hoặc -1 nếu xâu không tương ứng với một số nguyên nào.

**Ví dụ:**

| DECODE.INP |
|------------|
| abcac      |

| DECODE.OUT |
|------------|
| 129        |

## KHOẢNG CÁCH SỐ

Người ta định nghĩa khoảng cách giữa 2 số nguyên dương  $a, b$  là tổng độ lệch của các chữ số cùng hàng tương ứng trên  $a$  và  $b$ . Chẳng hạn với  $a = 2015$  và  $b = 48$  thì tổng độ lệch là  $d(a, b) = |2 - 0| + |0 - 0| + |1 - 4| + |5 - 8| = 8$ .

**Yêu cầu:** Cho 2 số nguyên dương  $a, b (1 \leq a, b \leq 10^9)$ . Hãy tính khoảng cách giữa  $a$  và  $b$  theo định nghĩa trên.

**Dữ liệu:** vào từ tập tin văn bản **DIST.INP** chứa hai số nguyên dương  $a, b$  trên cùng dòng cách nhau khoảng trắng.

**Kết quả:** ghi ra tập tin văn bản **DIST.OUT** giá trị khoảng cách giữa  $a, b$ .

**Ví dụ:**

| DIST.INP | DIST.OUT |
|----------|----------|
| 2015 48  | 8        |

## BẦU CỬ

Chính quyền đất nước Metacity đang tổ chức lấy phiếu bầu cho  $n$  ứng viên, được đánh thứ tự từ 1 đến  $n$ . Megacity có  $m$  thành phố và hệ thống bầu cử ở đất nước này được chia thành 2 giai đoạn khá lạ lùng như sau.

Giai đoạn đầu của đợt bầu cử, các lá phiếu chỉ tính cho từng thành phố, nghĩa là ứng viên được cử tri của thành phố nào bỏ nhiều phiếu nhất sẽ chiến thắng tại thành phố đó. Nếu có nhiều ứng viên cùng số phiếu cao nhất của cùng một thành phố thì ứng viên nào có thứ tự nhỏ hơn sẽ chiến thắng.

Ở giai đoạn tiếp theo, người chiến thắng được xác định theo nguyên tắc như sau: người thắng cử là người chiến thắng tại nhiều thành phố nhất. Nếu có nhiều ứng viên cùng thắng tại nhiều thành phố nhất thì ứng viên có thứ tự nhỏ hơn sẽ thắng cử.

**Yêu cầu:** hãy xác định ứng viên thắng cử

**Dữ liệu:** vào từ tập tin văn bản **ELECTION.INP**

- Dòng đầu tiên chứa 2 số nguyên  $n, m$  ( $1 \leq n, m \leq 1000$ )
- Dòng thứ  $i$  trong  $m$  dòng tiếp theo chứa  $n$  số nguyên  $a_{i1}, a_{i2}, \dots, a_{in}$  ( $0 \leq a_{ij} \leq 10^9$ ) cho biết số phiếu bầu của các ứng viên thứ  $j$  ở thành phố thứ  $i$ .

**Kết quả:** ghi ra tập tin văn bản **ELECTION.OUT** chỉ số của ứng viên thắng cử.

**Ví dụ:**

| ELECTION.INP |
|--------------|
| 3 3          |
| 1 2 3        |
| 2 3 1        |
| 1 2 1        |

| ELECTION.OUT |
|--------------|
| 2            |

## EQUATION

Hãy giải phương trình nghiệm nguyên dương sau:

$$x - s(x)^a * b - c = 0$$

trong đó  $s(x)$  là tổng các chữ số của  $x$ ;  $a, b, c$  nguyên ( $1 \leq a \leq 5$ ;  $1 \leq b \leq 10^4$ ;  $-10^4 \leq c \leq 10^4$ )

**Dữ liệu:** vào từ tập tin văn bản **EQUATION.INP**

- 3 số nguyên  $a, b, c$  trên cùng dòng và cách nhau khoảng trắng.

**Kết quả:** ghi ra tập tin văn bản **EQUATION.OUT**

- Tất cả nghiệm nguyên dương không vượt quá  $10^9$  của phương trình, mỗi nghiệm trên một dòng, theo thứ tự tăng dần. Nếu phương trình vô nghiệm xuất ra “No solution”.

*Ví dụ:*

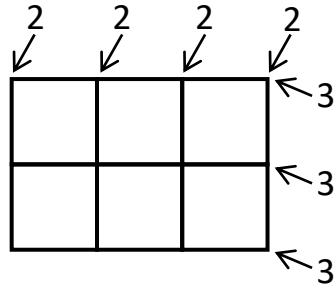
| EQUATION.INP |
|--------------|
| 3 2 8        |

| EQUATION.OUT |
|--------------|
| 10           |
| 2008         |
| 13726        |



## XÂY DỰNG HÀNG RÀO

Một khu đất hình chữ nhật kích thước  $m \times n$  (gồm  $m$  hàng, mỗi hàng gồm  $n$  ô vuông độ dài cạnh là 1). Người ta cần làm hàng rào để ngăn cách từng ô vuông riêng biệt (độ dày hàng rào là không đáng kể). Đường biên xung quanh khu đất cũng cần được rào lại.



**Yêu cầu** cho  $m, n$  là kích thước khu đất, hãy tính tổng độ dài cần rào.

Hình minh họa khu đất với kích thước  $m = 2, n = 3$  và tổng độ dài cần rào là 17 (4 hàng rào dọc độ dài 2; 3 hàng rào ngang độ dài 3).

**Dữ liệu:** vào từ tập tin văn bản **FENCES.INP** hai số nguyên  $m, n (1 \leq m, n \leq 10^4)$ .

**Kết quả:** ghi ra tập tin văn bản **FENCES.OUT** tổng độ dài cần rào.

**Ví dụ:**

| FENCES.INP |
|------------|
| 2 3        |

| FENCES.OUT |
|------------|
| 17         |

## TẶNG HOA

Nhân ngày phụ nữ Việt Nam 20/10, Hoàng muốn mua một bó hoa thật đẹp có giá trị trong phạm vi số tiền mình có là  $c$  đồng để tặng bạn gái. Vì bạn gái chỉ thích hoa lan và hoa hồng nên Hoàng chỉ tìm mua 2 loại hoa này. Hoa hồng có giá  $a$  đồng một bông, hoa lan có giá  $b$  đồng một bông ( $a < b$ ). Hoàng muốn mua được càng nhiều hoa càng tốt và giá trị của bó hoa phải là lớn nhất (dĩ nhiên, không vượt quá số tiền mình hiện có).

**Yêu cầu:** cho 3 số nguyên  $a, b, c$  ( $1 \leq a < b \leq 10^4, 0 \leq c \leq 10^6$ ). Hãy xác định giá trị của bó hoa mua được.

**Dữ liệu:** vào từ tập tin văn bản **FLOWERS.INP** 3 số nguyên  $a, b, c$ .

**Kết quả:** ghi ra tập tin văn bản **FLOWERS.OUT** giá trị lớn nhất của bó hoa mua được.

**Ví dụ:**

| FLOWERS.INP |
|-------------|
| 2 3 11      |

| FLOWERS.OUT |
|-------------|
| 11          |

## ĐẶT QUẦY PHỤC VỤ

Tí đang điều hành một tiệm thức ăn nhanh mở cửa 24/7, tiệm có nhiều quầy để phục vụ khách ăn uống. Một hôm, bằng cách nào đó Tí biết được ngày hôm sau sẽ có  $n$  khách ghé tiệm của mình.

Tí cũng biết chính xác thời điểm mà người khách thứ  $i$  sẽ ghé tiệm mình là vào lúc  $h_i$  giờ  $m_i$  phút ( $0 \leq h_i \leq 23; 0 \leq m_i \leq 59$ ). Thời gian để phục vụ cho một khách hàng là không đáng kể nhưng Tí hiểu rằng nếu khách đến mà thấy không còn quầy phục vụ nào trống thì khách sẽ bỏ đi.

Tí là người tham lam nên muốn phục vụ cho tất cả  $n$  khách hàng. Hãy giúp Tí tính số quầy phục vụ tối thiểu để phục vụ cho tất cả  $n$  khách.

**Dữ liệu:** vào từ tập tin **FREECASH.INP**

- Dòng đầu tiên chứa số nguyên dương  $n$  ( $n \leq 10^5$ ) là số khách sẽ ghé tiệm
- Dòng thứ  $i$  trong  $n$  dòng tiếp chứa 2 số nguyên  $h_i, m_i$  cho biết thời điểm người khách thứ  $i$  sẽ ghé tiệm theo thứ tự thời gian.

**Kết quả:** ghi ra tập tin văn bản **FREECASH.OUT** số quầy tối thiểu để phục vụ cho tất cả khách hàng.

**Ví dụ:**

| FREECASH.INP | FREECASH.OUT |
|--------------|--------------|
| 4            | 2            |
| 8 0          |              |
| 8 10         |              |
| 8 10         |              |
| 8 45         |              |

## THỨ SÁU NGÀY 13

Hành tinh Cyberplanet sử dụng lịch cũng tương tự với lịch của hành tinh chúng ta. Mỗi năm ở Cyberplanet có  $n$  tháng, mỗi tháng có đúng 30 ngày, mỗi tuần có 7 ngày và ngày không may mắn ở đây cũng là thứ sáu ngày 13. Biết rằng ngày đầu năm mới ở hành tinh Cyberplanet là ngày thứ  $k$  ( $1 \leq k \leq 7$ ) trong tuần ( $k = 1$ : thứ hai;  $k = 2$ : thứ ba, ...,  $k = 7$ : chủ nhật)

**Yêu cầu:** hãy xác định có bao nhiêu ngày không may mắn trong năm ở Cyberplanet

**Dữ liệu:** vào từ tập tin văn bản **FRIDAY.INP** 2 số nguyên dương

$n, k$  ( $1 \leq n \leq 10^9; 1 \leq k \leq 7$ )

**Kết quả:** xuất ra tập tin văn bản **FRIDAY.OUT** số ngày rơi vào thứ sáu ngày 13 trong năm.

**Ví dụ:**

| FRIDAY.INP | FRIDAY.OUT |
|------------|------------|
| 12<br>1    | 2          |

## HÀM SỐ

Cho số nguyên dương  $n (n \leq 10^6)$ , hàm số  $f$  được định nghĩa như sau:

$$f(n) = \begin{cases} n, & n < 10 \\ f(s(n)), & n \geq 10 \end{cases}$$

Trong đó  $s(n)$  là tổng các chữ số của  $n$ . Hãy tính  $f(n!)$

**Dữ liệu** vào từ tập tin văn bản **FUNCTION.INP** số nguyên dương  $n$

**Kết quả** ghi ra tập tin văn bản **FUNCTION.OUT** giá trị của  $f(n!)$

**Ví dụ:**

| FUNCTION.INP |
|--------------|
| 5            |

| FUNCTION.OUT |
|--------------|
| 3            |

## QUÀ TẶNG

Trong buổi tiệc chia tay cuối năm, các bạn học sinh đã mang theo  $n(n \leq 10^6)$  món quà để tặng cho nhau (buổi tiệc có  $n$  học sinh tham dự, mỗi học sinh chỉ mang một món quà).

Để tạo tính bất ngờ thú vị cho các học sinh tham dự, giáo viên chủ nhiệm tập hợp các món quà lại và đánh số thứ tự các hộp quà từ 1 đến  $n$  tương ứng với số thứ tự của chủ nhân hộp quà. Tiếp theo, cô tạo  $n$  lá thăm, trên mỗi lá thăm ghi một con số từ 1 đến  $n$ . Các học sinh sẽ lần lượt bốc ngẫu nhiên 1 lá thăm và sẽ nhận món quà có số thứ tự tương ứng với con số được ghi trên lá thăm mà học sinh bốc trúng.

Các học sinh tham dự rất háo hức để xem món quà mình sắp nhận là gì nhưng cũng rất muốn biết ai sẽ là người nhận được món quà mà mình đã chuẩn bị từ nhiều ngày trước đó.

**Dữ liệu:** vào từ tập tin văn bản **GIFTS.INP**:

- Dòng đầu tiên chứa số  $n$  là số học sinh tham dự
- Dòng thứ  $i + 1 (1 \leq i \leq n)$  chứa 1 số nguyên là số thăm mà học sinh thứ  $i$  bốc được

**Kết quả:** xuất ra tập tin văn bản **GIFTS.OUT** gồm  $n$  dòng, dòng thứ  $i$  ghi số thứ tự của học sinh nhận được quà của học sinh thứ  $i$ .

**Ví dụ:**

| GIFTS.INP |
|-----------|
| 4         |
| 2         |
| 3         |
| 4         |
| 1         |

| GIFTS.OUT |
|-----------|
| 4         |
| 1         |
| 2         |
| 3         |

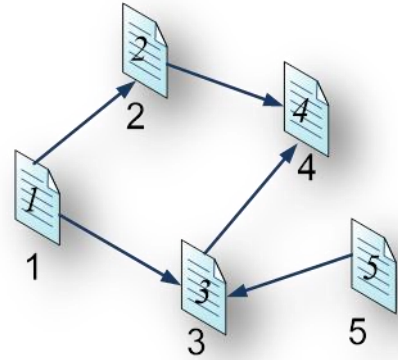
## BÀI TẬP VỀ NHÀ

Steve rất không thích làm bài tập ở nhà. Nhưng trong giờ Tin học thầy giáo cho tới  $n$  bài tập về nhà, trong đó có những bài chỉ có thể giải được sau khi làm một số bài khác.

Steve đọc đầu bài, ước lượng thời gian giải cho từng bài và thấy rõ rằng mình không kịp làm được hết tất cả các bài. Khi đó Steve quyết định sẽ bỏ một

bài. Hy vọng rằng nếu chỉ thiếu có một bài thầy giáo sẽ không mắng nhiều. Vấn đề là phải chọn bài nào để lại, không làm sao cho tổng thời gian làm các bài còn lại là nhỏ nhất.

Ví dụ, với  $n = 5$ , thời gian làm bài thứ  $i$  là  $i$  phút và các bài 2, 3 phải làm sau khi đã làm xong bài 1, bài 3 phải làm sau bài 5. Như vậy Steve có thể bỏ bài 4 và thời gian là các bài còn lại sẽ là  $1+2+3+5=11$  phút.



**Yêu cầu:** Cho các số nguyên  $n, m, t_i$  – thời gian làm bài thứ

$i, i = 1 \div n$  và  $m$  cặp quan hệ dạng  $(a, b)$  cho biết bài  $b$  phải làm sau bài  $a$ . Hãy xác định thời gian tối thiểu cần thiết để Steve thực hiện được kế hoạch của mình.

**Dữ liệu:** Vào từ file văn bản **HOMEWORK.INP**:

- Dòng đầu tiên chứa 2 số nguyên  $n$  và  $m$  ( $1 \leq n \leq 100, 0 \leq m \leq 1000$ ),
- Dòng thứ 2 chứa  $n$  số nguyên  $t_1, t_2, \dots, t_n$  ( $1 \leq t_i \leq 1000, i = 1 \div n$ ),
- Mỗi dòng trong  $m$  dòng sau chứa 2 số nguyên  $a$  và  $b$  ( $1 \leq a, b \leq n, a \neq b$ ).

**Kết quả:** Đưa ra file văn bản **HOMEWORK.OUT** một số nguyên – thời gian tối thiểu tìm được.

**Ví dụ:**

| HOMEWORK.INP |
|--------------|
| 5 5          |
| 1 2 3 4 5    |
| 1 2          |
| 5 3          |
| 1 3          |
| 3 4          |
| 2 4          |

| HOMEWORK.OUT |
|--------------|
| 11           |

## SỐ NGUYÊN TỐ

Cho hai số nguyên dương  $n$  và  $h$ , hãy liệt kê tất cả các số nguyên tố trong phạm vi từ 1 tới  $n$  có tổng các chữ số bằng  $h$ .

**Dữ liệu:** vào từ tập tin văn bản **HPRIME.INP** dòng chứa hai số nguyên dương

$n, h (n \leq 10^6, h \leq 10^9)$

**Kết quả:** ghi ra tập tin văn bản **HPRIME.OUT**

- Dòng 1 ghi số  $k$  là số những số nguyên tố trong phạm vi từ 1 tới  $n$  có tổng các chữ số bằng  $h$
- $k$  dòng tiếp theo, mỗi dòng ghi một số nguyên tố thỏa mãn yêu cầu đề ra, các số nguyên tố phải được liệt kê theo thứ tự tăng dần.

**Ví dụ:**

| HPRIME.INP | HPRIME.OUT                                       |
|------------|--|
| 500 16     | 7<br>79<br>97<br>277<br>349<br>367<br>439<br>457 |



## THẦN TƯỢNG

Có  $n$  người nổi tiếng đánh số từ 1 tới  $n$ . Người thứ  $i$  coi  $k_i$  người khác làm “thần tượng” của mình. Chú ý mối quan hệ “thần tượng” không phải quan hệ đối xứng, tức là có thể người  $a$  coi người  $b$  làm thần tượng nhưng người  $a$  không phải thần tượng của người  $b$ .

**Yêu cầu:** Liệt kê những người là thần tượng của nhiều người khác nhất.

**Dữ liệu:** vào từ tập tin văn bản **IDOLS.INP**

- Dòng 1: Chứa số nguyên dương  $n$
- $n$  dòng tiếp theo, dòng thứ  $i$  chứa số nguyên  $k_i$ , tiếp theo là  $k_i$  số nguyên đôi một khác nhau tương ứng với số hiệu những người được người  $i$  coi làm thần tượng

**Kết quả:** ghi ra tập tin văn bản **IDOLS.OUT** chỉ số của những người được nhiều người khác coi là thần tượng nhất. Các chỉ số tìm được cần ghi trên một dòng theo thứ tự tăng dần.

**Ràng buộc:**  $n \leq 10^5$ ;  $\sum_{i=1}^n k_i \leq 10^6$ ; Không người nào tự nhận mình làm thần tượng.

**Ví dụ:**

| IDOLS.INP | IDOLS.OUT |
|-----------|-----------|
| 5         | 2 4       |
| 2 2 4     |           |
| 3 5 1 4   |           |
| 2 2 5     |           |
| 1 1       |           |
| 2 2 4     |           |

## TỐI GIẢN PHÂN SỐ

Cho 4 số nguyên dương  $a, b, c, d$  ( $1 \leq a, b, c, d \leq 10^6$ ). Hãy tìm  $x, y$  là tử và mẫu tối giản tương ứng của phân số tổng

$$\frac{x}{y} = \frac{a}{b} + \frac{c}{d}$$

**Dữ liệu** vào từ tập tin văn bản **IRD.INP** 4 số nguyên dương  $a, b, c, d$

**Kết quả** ghi ra tập tin văn bản **IRD.OUT** 2 số nguyên dương  $x, y$

**Ví dụ:**

| IRD.INP |
|---------|
| 2 6 1 3 |

| IRD.OUT |
|---------|
| 2 3     |

## TÁO QUÂN

Có  $m$  ông táo vào  $n$  bà táo được Ngọc Hoàng phân công nhiệm vụ trong năm mới. Đầu tiên Ngọc Hoàng chọn  $k$  táo (ông hoặc bà) làm những nhiệm vụ đặc biệt tại các Bộ/Ngành, sau đó Ngọc Hoàng sẽ chọn ra các nhóm, mỗi nhóm gồm đúng 2 ông táo và 1 bà táo để phân xuống các gia đình dưới hạ giới.

**Yêu cầu:** Hãy giúp Ngọc Hoàng xác định số nhóm nhiều nhất để phân xuống các gia đình dưới hạ giới. Ví dụ có  $m = 12$  ông táo và  $n = 7$  bà táo, có  $k = 6$  táo phải làm nhiệm vụ đặc biệt. Ngọc Hoàng có thể chọn 4 ông táo và 2 bà táo làm nhiệm vụ đặc biệt, còn lại chia làm 4 nhóm (1 bà táo không được phân việc). Cách khác là chọn 3 ông táo và 3 bà táo làm nhiệm vụ đặc biệt, còn lại chia làm 4 nhóm (1 ông táo không được phân việc).

**Dữ liệu:** vào từ tập tin văn bản **LARES.INP** gồm 1 dòng chứa 3 số nguyên dương  $m, n, k \leq 10^9$  cách nhau ít nhất một dấu cách

**Kết quả:** ghi ra tập tin văn bản **LARES.OUT** một số nguyên duy nhất là số nhóm nhiều nhất chọn được để phân xuống các gia đình dưới hạ giới.

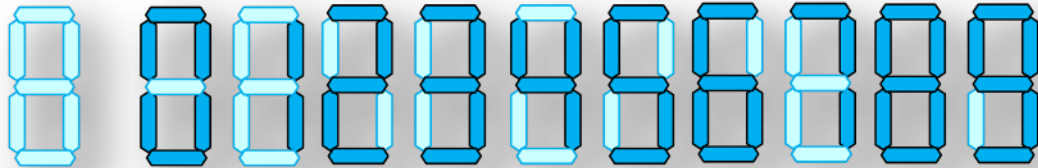
**Ví dụ:**

| LARES.INP |
|-----------|
| 12 7 5    |

| LARES.OUT |
|-----------|
| 4         |

## HIỆN SỐ BẰNG ĐÈN LED

Quà sinh nhật mà Mai nhận được là một máy tính bấm tay màn hình tinh thể lỏng. Màn hình có thể hiển thị được  $n$  chữ số. Có 7 đèn LED tạo thành 7 vạch để hiển thị một chữ số. Mỗi chữ số sẽ tương ứng với một số đèn LED được kích hoạt và vạch tương ứng sẽ có màu đen. Cách hiển thị các số là như sau:



Như vậy, để hiển thị số 0 cần 6 vạch đen, số 1 cần 2 vạch đen, ... Là người ham hiểu biết, Mai tự hỏi không biết số gồm  $n$  chữ số có nghĩa nào nhỏ nhất và lớn nhất có thể hiển thị với đúng  $k$  vạch đen trên màn hình.

**Yêu cầu:** cho  $n$  và  $(1 \leq n \leq 100, 1 \leq k \leq 700)$ . Hãy xác định số gồm  $n$  chữ số có nghĩa nhỏ nhất và lớn nhất, mỗi số được hiển thị với đúng  $k$  vạch đen trên màn hình.

**Dữ liệu:** vào từ tập tin văn bản **LED.INP** chứa 2 số nguyên  $n$  và  $k$

**Kết quả:** ghi ra tập tin văn bản **LED.OUT**

- Dòng thứ nhất ghi số nguyên nhỏ nhất
- Dòng thứ hai ghi số nguyên lớn nhất
- Nếu không có nghiệm thì đưa ra thông báo NO SOLUTION

**Ví dụ:**

| LED.INP | LED.OUT        |
|---------|----------------|
| 5 15    | 10117<br>97111 |

## THÀNH PHỐ MAY MẮN

Tháp Giga là tòa tháp cao nhất và sâu nhất thành phố Cyberland với tổng cộng có 17,777,777,777 tầng và được đánh thứ tự từ tầng -8,888,888,888 đến tầng 8,888,888,888 (có tầng 0 nằm giữa tầng -1 và tầng 1).

Người dân sống ở thành phố Cyberland tin rằng số 8 là một con số may mắn và một số nguyên được gọi là may mắn nếu nó có chứa ít nhất một chữ số 8. Chẳng hạn số 18, 380, -808 là các số may mắn.

Bờm là một du khách lần đầu đến Cyberland và cậu ta muốn viếng thăm Giga để tìm kiếm sự may mắn. Hiện tại Bờm đang đứng ở tầng thứ  $a$  của tòa tháp và cậu ta muốn tìm số nguyên dương  $b$  nhỏ nhất sao cho khi lên cao thêm  $b$  tầng tính từ tầng  $a$  thì cậu ta sẽ đến được tầng có số thứ tự là một số may mắn.

**Dữ liệu:** vào từ tập tin văn bản **LUCKY.INP** số nguyên  $a$  ( $-10^9 \leq a \leq 10^9$ )

**Kết quả:** ghi ra tập tin văn bản **LUCKY.OUT** số nguyên dương  $b$  nhỏ nhất tìm được

**Ví dụ:**

| LUCKY.INP |
|-----------|
| 179       |

| LUCKY.OUT |
|-----------|
| 1         |

| LUCKY.INP |
|-----------|
| 18        |

| LUCKY.OUT |
|-----------|
| 10        |

## LŨY THỪA CỦA 2

Hãy tính  $2^n$  .

**Dữ liệu vào** từ tập tin LUYTHUA2.INP

Chứa một số tự nhiên  $n \leq 10000$

**Dữ liệu kết quả** ra tập tin LUYTHUA2.OUT

Số  $2^n$

Ví dụ:

| LUYTHUA2.INP | LUYTHUA2.OUT |
|--------------|--------------|
| 10           | 1024         |

## CÁC THỎI NAM CHÂM

Nhà khoa học lập dị Mike đã nghĩ ra trò sắp các quân cờ domino để tiêu khiển nhưng Mike lại không có bộ cờ domino nên đã thay bằng các thanh nam châm. Mỗi thanh nam châm đều có 2 cực dương (+) và âm (-). Nếu 2 thanh nam châm đặt gần nhau thì các cực cùng dấu sẽ đẩy nhau và các cực trái dấu sẽ hút nhau.



Mike đặt các thanh nam châm thành một hàng ngang trên bàn. Mỗi lần anh ta thêm một thỏi nam châm vào cuối phải dãy. Tùy vào cách anh ta bố trí các thanh nam châm chúng sẽ hút nhau tạo thành một nhóm hàng ngang hoặc đẩy nhau tách biệt thành nhóm khác.

**Yêu cầu:** với cách sắp đặt các thanh nam châm thành một hàng ngang như trên, hãy xác định số nhóm nam châm được hình thành.

**Dữ liệu:** đọc từ tập tin văn bản **MAGNETS.INP** gồm nhiều dòng

- Dòng đầu chứa số nguyên dương  $n (n \leq 10^5)$ .
- Dòng thứ  $i$  trong  $n$  dòng tiếp theo, mỗi dòng chứa 2 số 0, 1 mô tả cho 2 đầu của thanh nam châm thứ  $i$ . 0 tương ứng với cực âm, 1 tương ứng cực dương

**Kết quả:** ghi ra tập tin văn bản **MAGNETS.OUT** gồm 1 số nguyên là số nhóm nam châm được hình thành

**Ví dụ:**

| MAGNETS.INP |
|-------------|
| 6           |
| 10          |
| 10          |
| 10          |
| 01          |
| 10          |
| 10          |

| MAGNETS.OUT |
|-------------|
| 3           |

## TẦN SỐ XUẤT HIỆN NHIỀU NHẤT

Cho dãy số nguyên  $a_1, a_2, \dots, a_n$ , hãy cho biết giá trị xuất hiện nhiều nhất dãy.

**Dữ liệu:** vào từ tập tin văn bản **MAXFREQ.INP**

- Dòng 1 chứa số nguyên dương  $n (n \leq 10^5)$
- Dòng thứ  $i$  trong  $n$  dòng tiếp theo chứa số nguyên  $a_i (|a_i| \leq 10^9)$

**Kết quả:** ghi ra tập tin văn bản **MAXFREQ.OUT** số nguyên là giá trị xuất hiện nhiều nhất trong dãy, nếu có nhiều giá trị thỏa điều kiện thì xuất giá trị nhỏ nhất.

**Ví dụ:**

| MAXFREQ.INP | MAXFREQ.OUT |
|-------------|-------------|
| 10          | 3           |
| 5           |             |
| 3           |             |
| 2           |             |
| 5           |             |
| 5           |             |
| 3           |             |
| 4           |             |
| 6           |             |
| 8           |             |
| 3           |             |



## HỖN HỢP

Harry Potter được giao 3 bình dung dịch cùng thể tích và đều chứa đầy. Dung dịch trong mỗi bình là hỗn hợp của 3 chất A, B và C. Trong bình thứ  $i$  tỷ lệ các chất A, B và C là

$X_i, Y_i, Z_i (i = 1, 2, 3)$ . Ví dụ, mỗi bình có thể tích 12 lít và bình thứ nhất chứa 3 lít chất A, 6 lít chất B và 3 lít chất C thì tỷ lệ các chất trong bình 1 là 1, 2, 1. Nhiệm vụ của Harry Potter là đổ chung 3 bình này vào một thùng rộng lớn và tính tỷ lệ các chất A, B, C trong thùng. Hãy lập trình giúp Harry Potter tính tỷ lệ các chất trong thùng dựa theo tỷ lệ đã cho ở các bình.

**Dữ liệu** vào từ tập tin văn bản **MIX.INP**, gồm 3 dòng - dòng thứ  $i$  chứa 3 số nguyên dương  $X_i, Y_i, Z_i$ , các số cách nhau một dấu cách.

**Kết quả** xuất ra tập tin văn bản **MIX.OUT** 3 số nguyên  $P, Q, R$  - tỷ lệ các chất A, B, C trong thùng. Các số phải nguyên tố cùng nhau và cách nhau một dấu cách.

**Ví dụ:**

| MIX.INP |
|---------|
| 1 2 1   |
| 2 1 3   |
| 3 2 1   |

| MIX.OUT  |
|----------|
| 13 12 11 |

## MODULO

Cho dãy số nguyên dương  $a_1, a_2, \dots, a_n$ . Hãy tìm số dư của phép chia từng phần tử của dãy  $a_i$  cho số nguyên dương  $k$ .

Dữ liệu vào từ tập tin văn bản **MODULO.INP** gồm nhiều dòng:

- Dòng đầu tiên chứa 2 số nguyên dương  $n, k$  ( $n \leq 10^5, k \leq 10^9$ )
- Dòng thứ  $i$  trong  $n$  dòng tiếp theo chứa số nguyên dương  $a_i$  ( $a_i \leq 10^{100}$ )

Kết quả xuất ra tập tin văn bản **MODULO.OUT** gồm  $n$  dòng. Dòng thứ  $i$  chứa một số nguyên dương là giá trị của phép chia lấy phần dư  $a_i$  cho  $k$

Ví dụ:

| MODULO.INP | MODULO.OUT |
|------------|------------|
| 5 7        | 1          |
| 50         | 4          |
| 130        | 2          |
| 79         | 0          |
| 42         | 6          |
| 27         |            |

## TIỀN

Hóa ra ai cũng cần tiền, kể cả phù thủy. Họ sử dụng các đồng vàng, bạc và đồng, gọi tương ứng là Galeon, Sikel và Knat. Một Galeon ăn 17 sikel, một sikel ăn 29 knat. Mọi giá cả nêu sau đều theo các đơn vị kể trên. Trong mỗi giá số sikel không quá 16, số knat – không quá 28.

Trước khi vào nhập học ở Hogvard Harry Potter rút ở ngân hàng Gringot một số tiền để mua một số học cụ cần thiết như đầu thần, cú, chậu thiếc, áo choàng, . . . Số tiền Harry rút ra là  $g$  Galeo,  $s$  Sikel và  $k$  Knat. Harry cần mua tất cả là  $n$  thứ. Vật thứ  $i$  có giá là  $(p_i, q_i, r_i)$ ,  $i = 1 \div n$ , ( $0 \leq n \leq 10^5$ ).

**Yêu cầu:** Hãy xác định số tiền Harry còn lại sau khi sắm mọi thứ. Nếu Harry không đủ tiền thì đưa ra số -1.

**Dữ liệu:** Vào từ file văn bản MONEY.INP:

- Dòng đầu tiên chứa 3 số nguyên  $g, s$  và  $k$  ( $0 \leq g \leq 10^5$ ),
- Dòng thứ 2 chứa số nguyên  $n$ ,
- Dòng thứ  $i$  trong  $n$  dòng sau chứa 3 số nguyên  $p_i, q_i, r_i$  ( $0 \leq p_i \leq 10^5$ ).

**Kết quả:** Đưa ra file văn bản MONEY.OUT trên một dòng 3 số nguyên xác định số tiền còn lại của Harry hoặc số -1.

**Ví dụ:**

| MONEY.INP |
|-----------|
| 5 16 10   |
| 2         |
| 3 9 21    |
| 0 4 0     |

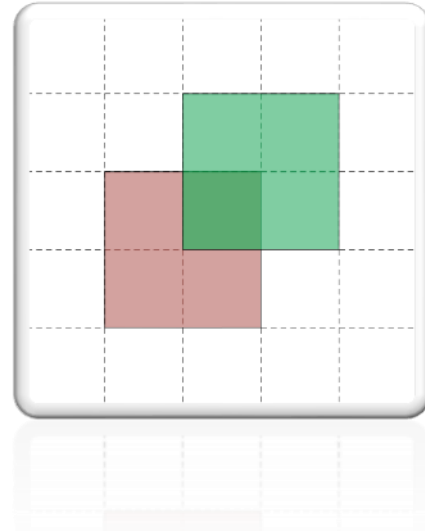
| MONEY.OUT |
|-----------|
| 2 2 18    |

## KHẮM TRANH

Một họa sỹ đã lấy cảm hứng từ các tranh khảm sành sứ nổi tiếng để tạo ra một tường phái nghệ thuật riêng – nghệ thuật khảm tranh bằng màu. Trên tấm vải nền màu trắng kích thước  $w \times h$  ô ông vẽ  $n$  hình chữ nhật có màu khác màu nền (màu trắng). Hình chữ nhật thứ  $i$  được xác định bởi cặp tọa độ đỉnh đối  $(x1_i, y1_i)$  và  $(x2_i, y2_i)$ ,  $0 \leq x1_i, x2_i \leq w$ ,  $0 \leq y1_i, y2_i \leq h$ ,  $1 \leq w, h \leq 100$ ,  $i = 1 \div n$ ,  $0 \leq n \leq 5\,000$ .

Hình bên ứng với trường hợp  $w = h = 5$ ,  $n = 2$ , hình chữ nhật thứ nhất được xác định bởi cặp điểm  $(1, 1)$  và  $(3, 3)$ , hình chữ nhật thứ 2 – cặp điểm  $(2, 4)$  và  $(4, 2)$ .

Phần vải còn để mộc (chưa bị tô) có diện tích là 18.



**Yêu cầu:** Cho  $w$ ,  $h$ ,  $n$  và tọa độ các điểm xác định từng hình chữ nhật. Hãy xác định phần vải còn được để mộc.

**Dữ liệu:** Vào từ file văn bản **MOSAIC.INP**:

- Dòng đầu tiên chứa 2 số nguyên  $w$  và  $h$ ,
- Dòng thứ 2 chứa số nguyên  $n$ ,
- Dòng thứ  $i$  trong  $n$  dòng còn lại chứa 4 số nguyên  $x1_i, y1_i, x2_i$  và  $y2_i$ .

**Kết quả:** Đưa ra file văn bản **MOSAIC.OUT** một số nguyên – diện tích phần để mộc.

**Ví dụ:**

| MOSAIC.INP |
|------------|
| 5 5        |
| 2          |
| 1 1 3 3    |
| 2 4 4 2    |

| MOSAIC.OUT |
|------------|
| 18         |

**NTFS**

Nếu đĩa cứng máy tính được format theo chế độ NTFS thì bộ nhớ được phân phối cho các file chứa trong đĩa cứng theo đơn vị cluster, mỗi cluster là một vùng nhớ có kích thước 4KB (1KB = 1024 byte). Như vậy dù tập tin có kích thước nhỏ hơn hoặc bằng 4KB thì nó cũng chiếm một vùng nhớ 4KB trên đĩa.

**Yêu cầu:** cho số nguyên dương  $n(n \leq 10^9)$  là kích thước của một tập tin theo đơn vị byte. Hãy xác định kích thước theo KB mà tập tin chiếm trên đĩa cứng NTFS.

**Dữ liệu:** Vào từ file văn bản **NTFS.INP** số nguyên dương  $n$ .

**Kết quả:** Đưa ra file văn bản **NTFS.OUT** kích thước theo KB mà tập tin chiếm giữ.

**Ví dụ:**

| NTFS.INP |
|----------|
| 4097     |

| NTFS.OUT |
|----------|
| 8        |

## SỐ THÂN THIỆN

Số tự nhiên có rất nhiều tính chất thú vị. Ví dụ với số 23, số đảo ngược của nó là 32. Hai số này có ước chung lớn nhất là 1. Những số như thế được gọi là số thân thiện, tức là số 23 được gọi là số thân thiện, số 32 cũng được gọi là số thân thiện.

**Yêu cầu:** cho 2 số nguyên  $a, b$  ( $10 \leq a \leq b \leq 10^5$ ). Đếm số lượng số thân thiện trong đoạn giá trị  $[a, b]$ .

**Dữ liệu:** vào từ tập tin văn bản **NUMFRE.INP** chứa 2 số  $a, b$ .

**Kết quả:** ghi ra tập tin văn bản **NUMFRE.OUT** số lượng số thân thiện tìm được.

**Ví dụ:**

| NUMFRE.INP |
|------------|
| 20 30      |

| NUMFRE.OUT |
|------------|
| 3          |

## NUMPOS

Phong được lớp trưởng phân công trực nhật vì vậy hôm nay bạn đến trường sớm, giặt khăn lau bảng và xóa bảng. Đang xóa bỗng Phong nhận thấy dãy số mà mình đã xóa một phần khá đặc biệt, phần đầu của nó là

1 2 3 2 3 4 3 4 5 4 5 6 5 6 7

Đáng tiếc, phần còn lại của dãy đã bị xóa mất. Cuối cùng Phong cũng xóa sạch bảng trước khi chuông reng vào lớp, nhưng dãy số trên vẫn cứ lớn vồn mãi trong đầu ☹

Buổi tối khi đi ngủ, Phong lại nghĩ về dãy số này. Phong nhận thấy số 1 xuất hiện 1 lần trong dãy, số 2 xuất hiện 2 lần và lần đầu ở vị trí thứ 2, số 3 xuất hiện 3 lần và lần đầu ở vị trí thứ 3, nhưng số 4 thì lần đầu tiên xuất hiện không ở vị trí thứ 4. Tổng quát hơn số  $k$  sẽ xuất hiện lần đầu ở vị trí thứ mấy trong dãy.

Bạn có thể giúp Phong tìm ra vị trí xuất hiện đầu tiên của số nguyên  $k$  trong dãy không?

**Dữ liệu:** vào từ tập tin văn bản **NUMPOS.INP** số nguyên  $k (1 \leq k \leq 10^9)$

**Kết quả:** ghi ra tập tin văn bản **NUMPOS.OUT** vị trí xuất hiện đầu tiên của số nguyên  $k$

**Ví dụ:**

| NUMPOS.INP |
|------------|
| 5          |

| NUMPOS.OUT |
|------------|
| 9          |

## TRÒ CHƠI VỚI DÃY SỐ

Tí rất thích môn Số học nên thường nghĩ ra các câu đố có liên quan đến những con số để đố Tèo là em trai mình. Lần này Tí có  $n$  số nguyên dương  $a_1, a_2, \dots, a_n$ , cậu ta đố em trai thực hiện các thao tác sau trên dãy số:

- Chọn 2 số bất kỳ có giá trị khác nhau trong dãy
- Giảm giá trị của số lớn hơn đi một đại lượng bằng giá trị của số nhỏ hơn còn lại.

Tèo có thể thực hiện các thao tác trên với số lần tùy thích sao cho tổng các phần tử của dãy là nhỏ nhất có thể có.

**Dữ liệu** vào từ tập tin văn bản **NUMQUIZ.INP**

- Dòng đầu chứa số nguyên dương  $n (n \leq 10^5)$
- Dòng tiếp theo chứa  $n$  số nguyên dương  $a_1, a_2, \dots, a_n (1 \leq a_i \leq 10^5)$

**Kết quả** xuất ra tập tin văn bản **NUMQUIZ.OUT** số nguyên duy nhất là giá trị tổng nhỏ nhất

**Ví dụ:**

| NUMQUIZ.INP |
|-------------|
| 2           |
| 1 2         |

| NUMQUIZ.OUT |
|-------------|
| 2           |



## CON SỐ BÍ ẨN

Cho 5 số nguyên dương  $a_1, a_2, a_3, a_4, a_5$ , trong đó 4 số có cùng giá trị  $x$ , số còn lại có giá trị  $y (x \neq y)$ . Ví dụ hình minh họa 4 số có cùng giá trị  $x = 3$  và giá trị còn lại là  $y = 7$ .



Nhiệm vụ của bạn là chỉ ra giá trị  $y$  và vị trí của số mang giá trị  $y$ .

**Dữ liệu:** vào từ tập tin văn bản **ODDNUM.INP** năm số nguyên dương theo thứ tự  $a_1, a_2, a_3, a_4, a_5$ .

**Kết quả:** ghi ra tập tin văn bản **ODDNUM.OUT** hai số nguyên dương: số đầu tiên là giá trị  $y$ ; số tiếp theo là vị trí của số mang giá trị  $y$ .

**Ví dụ:**

| ODDNUM.INP |
|------------|
| 3 7 3 3 3  |

| ODDNUM.OUT |
|------------|
| 7 2        |

## LUYỆN TẬP DỰ THI HỌC SINH GIỎI

Để chuẩn bị cho kỳ thi học sinh giỏi môn tin học, thầy giáo ra  $n$  bài tập ( $1 \leq n \leq 10^5$ ) được đánh số từ 1 đến  $n$ . Mỗi bài tập nhằm rèn luyện một số kỹ năng nào đó.

Nhằm định hướng cho quá trình tự luyện tập được hiệu quả, mỗi bài tập có một yêu cầu tối thiểu về trình độ kỹ năng. Để giải được bài thứ  $i$ , học sinh cần có trình độ kỹ năng tối thiểu là  $a_i$ . Điều này có nghĩa là học sinh có thể giải được bài thứ  $i$  khi và chỉ khi có trình độ kỹ năng bằng hoặc lớn hơn  $a_i$ . Nếu giải được bài thứ  $i$  trình độ kỹ năng của học sinh sẽ tăng thêm một lượng là  $b_i$  ( $1 \leq b_i \leq 10^9$ ). Giả sử ban đầu trình độ kỹ năng của học sinh trước khi làm bài tập là  $c$  ( $1 \leq c \leq 10^9$ ). Các bài tập có thể được làm theo trình tự bất kỳ tùy chọn.

Ví dụ, với trình độ kỹ năng ban đầu là  $c = 1, n = 4$  và các giá trị  $a_i, b_i$  tương ứng là (1, 10), (21, 5), (1, 10), (100, 100), học sinh sẽ giải bài 1, sau đó làm bài 3 và cuối cùng làm bài 2. Như vậy học sinh sẽ giải được tất cả 3 bài.

**Yêu cầu:** cho các số nguyên  $n, c$  và  $n$  cặp giá trị  $(a_i, b_i)$ . Hãy xác định số lượng bài tối đa có thể được giải.

**Dữ liệu:** vào từ tập tin văn bản **OLYMPIC.INP** có cấu trúc như sau:

- Dòng đầu tiên chứa 2 số nguyên  $n, c$
- Dòng thứ  $i$  trong  $n$  dòng tiếp theo ( $1 \leq i \leq n$ ) chứa 2 số nguyên  $a_i, b_i$

**Kết quả:** xuất tập tin văn bản **OLYMPIC.OUT** một số nguyên – số lượng bài tối đa có thể được giải.

**Ví dụ:**

| OLYMPIC.INP | OLYMPIC.OUT |
|-------------|-------------|
| 4 1         | 3           |
| 1 10        |             |
| 21 5        |             |
| 1 10        |             |
| 100 100     |             |

## SỐ ĐỐI XỨNG

Số đối xứng là số có thể viết từ trái sang phải các chữ số của nó ta vẫn được chính nó. Từ một số có 2 chữ số, ta có thể nhận được một số đối xứng theo cách sau: lấy số ban đầu cộng với số phản xạ gương của nó, tức là số nhận được bằng cách đọc các chữ số từ phải sang trái. Nếu chưa phải là số đối xứng, số đó lại cộng với ánh xạ gương của nó và tiếp tục như vậy cho tới khi nhận được số đối xứng.

Ví dụ từ 48 ta có  $48 + 84 = 132$ ,  $132 + 231 = 363$ . Như vậy 363 là số đối xứng tương ứng với 48.

**Yêu cầu:** cho số nguyên dương  $n$ , tìm số đối xứng tương ứng của  $n$ .

**Dữ liệu:** vào từ tập tin văn bản **PALNUM.INP** chứa số nguyên dương  $n$  ( $11 \leq n \leq 99$ )

**Kết quả:** ghi ra tập tin văn bản **PALNUM.OUT** số đối xứng tương ứng.

**Ví dụ:**

| PALNUM.INP |
|------------|
| 48         |

| PALNUM.OUT |
|------------|
| 363        |

## TRÔNG XE

Một bãi đỗ xe nhận trông xe trong vòng một tháng. Mỗi xe sẽ được gắn một số hiệu là một số nguyên dương  $T(10102010 \leq T \leq 10109999)$ . Hai xe khác nhau sẽ được gắn hai số hiệu khác nhau. Một xe có thể ra vào bãi đỗ xe nhiều lần, mỗi lần vào bãi đỗ xe, người trông xe sẽ ghi vào sổ sách số hiệu của chiếc xe đó.

Cuối tháng dựa vào sổ ghi chép, người trông xe làm thống kê về số lần vào bãi đỗ xe của từng chiếc xe để tiến hành thu phí. Nếu một chiếc xe vào bãi đỗ xe  $p$  lần, cuối tháng chủ xe phải trả một lượng phí  $C$  được tính như sau:

$$C = \begin{cases} 100, & p \leq 5 \\ 100 + (p - 5), & p > 5 \end{cases}$$

**Yêu cầu:** Tính tổng số phí người trông xe thu được vào cuối tháng.

**Dữ liệu:** vào từ tập tin văn bản **PARK.INP** có dạng:

- Dòng đầu chứa một số nguyên dương  $k(0 < k \leq 10^6)$
- $k$  dòng tiếp theo, mỗi dòng chứa số hiệu của một chiếc xe

**Kết quả:** ghi ra tập tin văn bản **PARK.OUT** một số nguyên là tổng số phí thu được.

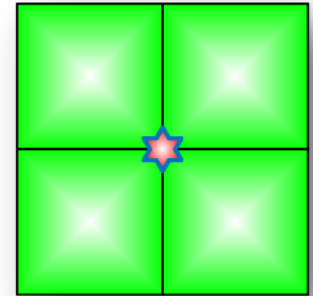
**Ví dụ:**

| PARK.INP | PARK.OUT |
|----------|----------|
| 7        | 201      |
| 10102010 |          |
| 10108888 |          |
| 10102010 |          |
| 10102010 |          |
| 10102010 |          |
| 10102010 |          |
| 10102010 |          |

## ĐỖ XE

Ở thời đại khi ô tô bay phát triển, nạn kẹt xe giảm hẳn, nhưng lại xuất hiện những vấn đề mới. Bất chấp những quy định nghiêm ngặt nhất, nhiều người vẫn đỗ xe lung tung khắp nơi, kể cả trên thảm cỏ xanh được chăm sóc hết sức cẩn thận ở quảng trường trung tâm thành phố. Người ta phải dựng các chốt chống đỗ xe.

Thảm cỏ có dạng hình chữ nhật kích thước  $n \times m$  ô. Mỗi xe khi đỗ cần đúng một ô trống không có vật cản (kể cả ở trên biên). Việc xây dựng các chốt chống đỗ xe cũng khá tốn kém và nếu dựng quá nhiều cũng sẽ mất mỹ quan. Vì vậy người ta cố gắng dựng càng ít càng tốt. Ví dụ, với  $n = m = 2$  thì chỉ cần dựng một chốt ở tâm là đủ.



**Yêu cầu:** Cho  $n, m (1 \leq n, m \leq 10^6)$ . Hãy xác định số chốt ít nhất cần dựng.

**Dữ liệu:** vào từ tập tin văn bản **PARKING.INP** chứa 2 số nguyên  $n$  và  $m$ .

**Kết quả:** ghi ra tập tin văn bản **PARKING.OUT** số lượng chốt ít nhất cần dựng.

**Ví dụ:**

| PARKING.INP |
|-------------|
| 2 2         |

| PARKING.OUT |
|-------------|
| 1           |

## ĐÒ TÌM MẬT KHẨU

Phòng thí nghiệm của trung tâm công nghệ cao được trang bị khá nhiều thiết bị máy móc hiện đại phục vụ cho công việc nghiên cứu. Các cửa ra vào đều được trang bị hệ thống nhận dạng thẻ từ hiện đại. Tuy nhiên, các nhà nghiên cứu thường là những người đăng trí nên họ hay để quên hoặc làm rớt thẻ từ.

Mặc dù các nhà nghiên cứu hay đăng trí nhưng họ rất thông minh và giỏi lập trình nên giám đốc liền nghĩ ra một cách thức nhận dạng mới mà không cần dùng đến thẻ từ. Trước mỗi cửa ra vào, giám đốc cho thiết kế một bảng điện tử cứ sau mỗi giây sẽ hiển thị 1 dãy hàng ngang gồm rất nhiều số nguyên dương. Mật khẩu để mở cửa chính là chuỗi số được tạo thành bằng cách ghép cặp số ở 2 vị trí khác nhau theo đúng thứ tự xuất hiện của chúng trong dãy thỏa điều kiện tổng các chữ số của cặp số này là khác nhau và lệch nhau nhỏ nhất. Nếu trong dãy tồn tại nhiều cặp phân tử như thế thì chọn cặp phân tử trái nhất.

Ví dụ dãy số hiển thị trên bảng như sau:

|                         |    |     |    |     |    |      |    |
|-------------------------|----|-----|----|-----|----|------|----|
| Dãy số                  | 75 | 305 | 19 | 221 | 43 | 1592 | 36 |
| Giá trị tổng các chữ số | 12 | 8   | 10 | 5   | 7  | 20   | 9  |

Cặp số 305 và 43 là cặp số đầu tiên tính từ trái sang có tổng các chữ số là 8 và 7 tương ứng, độ lệch tổng các chữ số của cặp này là 1, trong trường hợp này mật khẩu để mở cửa là 30543. Một nghiên cứu viên tập sự đang gặp khó khăn trong việc xác định mật khẩu theo phương pháp của giám đốc đưa ra. Bạn hãy giúp nghiên cứu viên này tìm ra mật khẩu để mở cửa trong thời gian nhanh nhất có thể.

**Dữ liệu:** vào từ tập tin văn bản **PASSWORD.INP**:

- Dòng đầu ghi số nguyên dương  $n (n \leq 10^6)$
- Dòng tiếp theo ghi  $n$  số nguyên  $a_1, a_2, \dots, a_n (1 \leq a_i \leq 10^9)$  là các số hiển thị trên bảng điện tử
- Dữ liệu được cho đảm bảo tồn tại ít nhất 2 phân tử có tổng các chữ số là khác nhau

**Kết quả:** xuất ra tập tin văn bản **PASSWORD.OUT** là mật khẩu tìm được.

**Ví dụ:**

| PASSWORD.INP             |
|--------------------------|
| 7                        |
| 75 305 19 221 43 1592 36 |

| PASSWORD.OUT |
|--------------|
| 30543        |

## ĐA GIÁC

Trên mặt phẳng có  $n$  đa giác không tự cắt, các đa giác đánh số từ 1 đến  $n$ . Ta nói rằng đa giác thứ  $i$  nằm trong đa giác thứ  $j$  nếu mọi điểm thuộc đa giác thứ  $i$  đều nằm trong đa giác thứ  $j$ . Với hai đa giác bất kì  $i$  và  $j$  ( $i \neq j$ ) thỏa hoặc đa giác  $i$  nằm trong đa giác  $j$  hoặc đa giác  $j$  nằm trong đa giác  $i$ .

Đa giác thứ  $k$  gọi là đa giác trung vị nếu có đúng  $m$  đa giác nằm trong đa giác thứ  $k$  với  $m = \left\lfloor \frac{n}{2} \right\rfloor$

, trong đó  $\left\lfloor \frac{n}{2} \right\rfloor$  là số nguyên lớn nhất không lớn hơn  $\frac{n}{2}$ .

**Yêu cầu:** Hãy tìm đa giác trung vị của  $n$  đa giác đã cho.

**Dữ liệu :** Vào từ file văn bản **POLYGON.INP**

- Dòng đầu tiên là số  $n$  ( $2 \leq n \leq 10^4$ ) – số đa giác.
- $n$  dòng tiếp theo, dòng thứ  $i$  mô tả đa giác thứ  $i$  : Đầu tiên là số đỉnh của đa giác  $a_i$  ( $3 \leq a_i \leq 100$ ), tiếp theo là  $2 \times a_i$  số nguyên có giá trị tuyệt đối không vượt quá  $2 \times 10^9$  :  $x_{i1}, y_{i1}, x_{i2}, y_{i2}, \dots$  với  $(x_{i1}, y_{i1}), (x_{i2}, y_{i2}), \dots$  lần lượt là tọa độ các đỉnh của đa giác được liệt kê theo một chiều nào đó.

**Kết quả:** Ghi ra file văn bản **POLYGON.OUT** số nguyên  $k$  – chỉ số của đa giác trung vị.

**Ví dụ :**

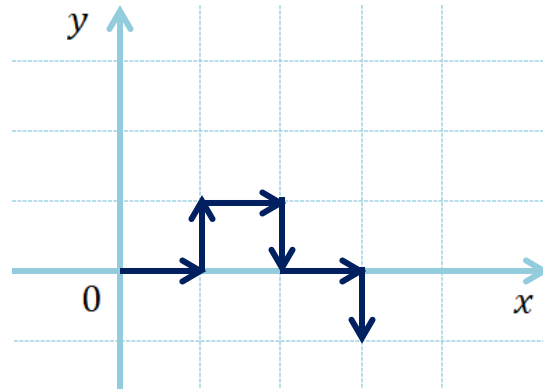
| POLYGON.INP  | POLYGON.OUT |
|--|-------------|
| 3<br>3 -2 1 8 9 12 1<br>3 7 5 6 3 7 4<br>4 4 3 7 7 9 3 1 2 | 3           |

## ROBOT DI CHUYỂN

Cho lưới nguyên Oxy. Điểm nguyên  $(x_1, y_1)$  và điểm nguyên  $(x_2, y_2)$  được gọi là kề nhau nếu thỏa điều kiện  $|x_1 - x_2| + |y_1 - y_2| = 1$ .

Một robot ban đầu đứng tại gốc tọa độ. Ở mỗi bước, robot sẽ di chuyển sang một điểm nguyên kề với vị trí hiện tại.

Từ bước di chuyển thứ 2 trở đi, robot có thể đi tiếp theo hướng cũ, rẽ sang trái, rẽ sang phải hay trở lại vị trí trước đó.



Trong ví dụ ở hình bên, từ ô  $(0, 0)$ , robot đi đến  $(1, 0)$ , rẽ trái sang ô  $(1, 1)$ , rẽ phải sang ô  $(2, 1)$ , rẽ phải sang ô  $(2, 0)$ , rẽ trái sang ô  $(3, 0)$  cuối cùng rẽ phải sang ô  $(3, -1)$ .

**Yêu cầu:** cho tọa độ các điểm nguyên mà robot đã đi qua. Hãy đếm xem robot đã rẽ phải bao nhiêu lần.

**Dữ liệu:** đọc từ tập tin văn bản **ROBOT.INP** có cấu trúc sau:

- Dòng đầu tiên chứa 1 số nguyên dương  $n$  ( $2 \leq n \leq 10000$ ) là tổng số điểm nguyên mà robot đã đi qua (kể cả vị trí xuất phát là gốc tọa độ)
- Dòng thứ  $i$  trong  $n$  dòng tiếp theo chứa 2 số nguyên  $x_i, y_i$  là tọa độ điểm nguyên mà robot đã đi qua.

**Kết quả:** xuất ra tập tin văn bản **ROBOT.OUT** gồm một số nguyên là số lần robot đã rẽ phải

**Ví dụ**

| ROBOT.INP |
|-----------|
| 7         |
| 0 0       |
| 1 0       |
| 1 1       |
| 2 1       |
| 2 0       |
| 3 0       |
| 3 -1      |

| ROBOT.OUT |
|-----------|
| 3         |



## QUAY BẢNG

Cho 4 số nguyên  $a, b, c, d$  được viết vào bảng vuông kích thước  $2 \times 2$  theo thứ tự sau

|     |     |
|-----|-----|
| $a$ | $b$ |
| $c$ | $d$ |

Ta gọi giá trị của bảng trên bằng

$$\frac{a}{c} - \frac{b}{d}$$

**Yêu cầu:** Xác định số lượt quay bảng 90 độ theo chiều kim đồng hồ để được bảng có giá trị lớn nhất. Nếu có nhiều cách quay bảng được cùng giá trị lớn nhất, chỉ ra số lượt quay ít nhất.

|     |    |               |     |    |
|-----|----|---------------|-----|----|
| 41  | 99 | $\Rightarrow$ | 100 | 41 |
| 100 | 13 |               | 13  | 99 |

**Dữ liệu:** vào từ tập tin văn bản **ROTATE.INP** gồm 1 dòng chứa 4 số nguyên dương

$a, b, c, d \leq 100$ .

**Kết quả:** ghi ra tập tin văn bản **ROTATE.OUT** một số nguyên duy nhất là số lượt quay tối thiểu tìm được.

**Ví dụ:**

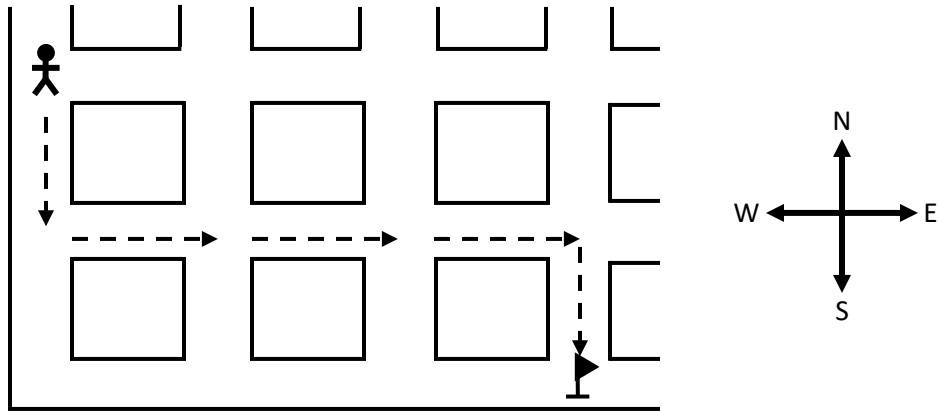
| ROTATE.INP   |
|--------------|
| 41 99 100 13 |

| ROTATE.OUT |
|------------|
| 1          |

## THAM QUAN THÀNH PHỐ

Bờm lần đầu tiên được tham quan khu trung tâm lớn nhất của thành phố Megacity. Khu trung tâm xây dựng nhiều con đường nối từ Bắc xuống Nam và từ Đông sang Tây. Các con đường được đánh thứ tự từ 1 trở đi, bắt đầu từ góc dưới bên trái (hình minh họa)

Bờm đang đứng tại giao lộ của 2 con đường  $x_1$  và  $y_1$  và cậu ta muốn đến khu trung tâm mua sắm nằm tại giao lộ của 2 con đường  $x_2$  và  $y_2$ . Chẳng hạn



Bờm đang đứng tại giao lộ của 2 con đường  $x_1 = 1$  và  $y_1 = 3$ , khu trung tâm mua sắm nằm tại giao lộ  $x_2 = 4$  và  $y_2 = 1$ . Bờm có thể đi 1 đoạn về hướng Nam (S), 3 đoạn về hướng Đông (E) và 1 đoạn về hướng Bắc (một đoạn được tính bắt đầu từ một giao lộ đến một giao lộ liền kề trên cùng hướng đi) hoặc một lộ trình khác là 2 đoạn về hướng Nam (S), 1 đoạn về hướng Đông (E), 1 đoạn về hướng Bắc (N), 2 đoạn về hướng Đông (E), 1 đoạn về hướng Nam (S).

**Yêu cầu:** hãy chỉ ra một lộ trình ngắn nhất hướng dẫn Bờm đến địa điểm cần tham quan.

**Dữ liệu:** vào từ tập tin văn bản **ROUTE.INP** gồm 4 số nguyên dương  $x_1, y_1, x_2, y_2$  có giá trị không vượt quá  $10^5$ . Điểm xuất phát và điểm đến không trùng nhau.

**Kết quả:** ghi ra tập tin văn bản **ROUTE.OUT** dãy gồm các kí tự in hoa {N, S, W, E} mô tả một lộ trình di chuyển ngắn nhất tìm được.

**Ví dụ:**

| ROUTE.INP |
|-----------|
| 1 3 4 1   |

| ROUTE.OUT |
|-----------|
| SEES      |

## DÃY SỐ VÔ HẠN

Người ta viết liên tục các số tự nhiên từ 1 trở đi không chứa khoảng cách tạo thành một chuỗi số vô hạn như sau 123456789101112131415...

**Yêu cầu:** cho số nguyên dương  $n \leq 10^{18}$ . Cho biết chiều dài của chuỗi số khi viết liên tiếp các số tự nhiên từ 1 đến  $n$  theo cách trên.

**Dữ liệu:** vào từ tập tin văn bản **SEQLEN.INP** số nguyên dương  $n$

**Kết quả:** ghi ra tập tin văn bản **SEQLEN.OUT** số nguyên dương là chiều dài của chuỗi số.

**Ví dụ:**

| SEQLEN.INP |
|------------|
| 15         |

| SEQLEN.OUT |
|------------|
| 21         |

## DÃY SỐ ĐẢO NGƯỢC

Xét dãy số nguyên  $a_1, a_2, \dots, a_n, \dots$ , trong đó  $a_1 = 1$ ,  $a_n$  được xác định như sau: đảo ngược thứ tự viết các chữ số của  $a_{n-1}$  (trong hệ cơ số 10) và cộng thêm 2 vào số nhận được.

Phần đầu của dãy số này có giá trị như sau:

| Chỉ số  | 1 | 2 | 3 | 4 | 5 | 6  | 7  | 8  | ... |
|---------|---|---|---|---|---|----|----|----|-----|
| Dãy $a$ | 1 | 3 | 5 | 7 | 9 | 11 | 13 | 33 | ... |

**Yêu cầu:** Cho số nguyên dương  $n$ . Hãy xác định  $a_n$  ( $1 \leq n \leq 10^{12}$ ).

**Dữ liệu:** Vào từ file văn bản **SEQUENCE.INP** gồm nhiều tests, mỗi test cho trên một dòng chứa một số nguyên  $n$ .

**Kết quả:** Đưa ra file văn bản **SEQUENCE.OUT**, kết quả mỗi test đưa ra trên một dòng.

**Ví dụ:**

| SEQUENCE.INP |
|--------------|
| 1            |
| 12           |

| SEQUENCE.OUT |
|--------------|
| 1            |
| 77           |

## ỐC SÊN

Con ốc sên đang ở gốc của một cái cây cao  $v$  mét tính từ gốc. Ốc sên muốn bò lên ngọn cây để ăn những lá non trên đó. Ban ngày ốc sên bò được  $a$  mét lên trên, nhưng ban đêm, khi ngủ nó bị trôi xuống dưới  $b$  mét.

**Yêu cầu:** Cho các số nguyên  $v$ ,  $a$  và  $b$  ( $1 \leq b < a \leq v \leq 10^9$ ). Hãy xác định số ngày cần thiết để ốc sên lên tới ngọn cây.

**Dữ liệu:** Vào từ file văn bản **SNAIL.INP** gồm một dòng chứa 3 số nguyên  $a$ ,  $b$  và  $v$ .

**Kết quả:** Đưa ra file văn bản **SNAIL.OUT** một số nguyên – kết quả tìm được.

**Ví dụ:**

| SNAIL.INP |
|-----------|
| 2 1 5     |

| SNAIL.OUT |
|-----------|
| 4         |



## TỔNG

Steve mới được quà sinh nhật từ bố mẹ. Đó là một chiếc máy tính bấm tay Casio mới tinh. Sau khi được hướng dẫn cách thực hiện liên hoàn các phép tính Steve chạy về phòng mình ngồi hàng giờ để tính tổng các số nguyên liên tiếp

$$a + (a+1) + (a+2) + \dots + b$$

Steve hãnh diện cho bố mẹ xem tổng  $S$  nhận được và ngẩn người khi được hỏi tổng  $S$  được tính từ đâu tới đâu!

**Yêu cầu:** Cho số nguyên  $S$  ( $1 \leq S \leq 10^{12}$ ). Hãy xác định các cặp số nguyên dương  $a, b$  ( $a \leq b$ ) tương ứng với  $S$  đã cho.

**Dữ liệu:** Vào từ file văn bản **SUM.INP** gồm một dòng chứa số nguyên  $S$ .

**Kết quả:** Đưa ra file văn bản **SUM.OUT**:

- Dòng đầu tiên chứa số nguyên  $k$  – số lượng cặp số tìm được,
- Mỗi dòng trong  $k$  dòng sau chứa một cặp số nguyên  $a, b$ .

Các cặp số đưa ra theo thứ tự tăng dần của  $a$ .

**Ví dụ:**

| SUM.INP |
|---------|
| 25      |

| SUM.OUT |
|---------|
| 3       |
| 3 7     |
| 12 13   |
| 25 25   |

## MUA VÉ XE

Để khuyến khích mọi người sử dụng các phương tiện giao thông trong thành phố, ngoài việc bán lẻ với giá  $p_1$  cho một vé thì ở hầu hết các nước đều có chế độ bán sỉ với giá  $p_2$  cho mỗi tập gồm  $k$  vé.

Bạn đến thành phố tham quan và dự kiến sẽ đi  $n$  chuyến trên các phương tiện giao thông công cộng. Vấn đề đặt ra là nên mua vé như thế nào để tiết kiệm tiền nhất. Dĩ nhiên, bạn không hề có ý nghĩ là sẽ đi lậu vé một lần nào.

**Yêu cầu:** Cho các số nguyên dương  $n, k, p_1, p_2$  các số đều có giá trị không vượt quá  $10^9$ . Nếu  $k = 1$  thì  $p_1 = p_2$ . Hãy tính chi phí tối thiểu cần thiết để mua vé.

**Dữ liệu:** vào từ tập tin văn bản **TICKETS.INP** 4 số nguyên  $n, k, p_1, p_2$ .

**Kết quả:** ghi ra tập tin văn bản **TICKETS.OUT** chi phí tối thiểu tìm được.

**Ví dụ:**

| TICKETS.INP  |
|--------------|
| 12 10 17 120 |

| TICKETS.OUT |
|-------------|
| 154         |

## XÂY THÁP

Có  $N$  khối đá hình hộp chữ nhật. Kích thước mỗi khối được biểu diễn bằng 3 số nguyên dương  $d_1, d_2, d_3$ . Người ta muốn xây một cái tháp bằng cách chồng các khối đá này lên nhau. Để đảm bảo an toàn, các khối đá được đặt theo nguyên tắc:

- Chiều cao của mỗi khối là kích thước nhỏ nhất trong ba kích thước  $d_1, d_2, d_3$ ;
- Các mép của các khối được đặt song song với nhau sao cho không có phần nào của khối nằm trên bị chìa ra ngoài so với khối nằm dưới.

**Yêu cầu:** Hãy tìm phương án xây dựng để đạt được tháp cao nhất.

**Dữ liệu:** vào từ tập tin văn bản **TOWER.INP**:

- Dòng đầu tiên là số  $N$
- $N$  dòng tiếp, mỗi dòng ghi 3 số nguyên dương là kích thước của một khối đá. Các khối đá được đánh số từ 1 theo trình tự xuất hiện trong file.

Các số trên một dòng trong file được ghi cách nhau ít nhất một dấu cách. Giới hạn số khối đá không quá 5000 và các kích thước của các khối đá không quá 255.

**Kết quả:** ghi ra tập tin văn bản **TOWER.OUT** một số nguyên dương là độ cao của tháp tìm được.

*Các tập tin dữ liệu mẫu:*

| TOWER.INP  | TOWER.OUT | Giải thích   |
|--|-----------|--|
| <b>9</b><br><b>7 5 5</b><br><b>4 4 8</b><br><b>1 1 5</b><br><b>4 2 2</b><br><b>5 1 5</b><br><b>4 2 7</b><br><b>2 9 2</b><br><b>1 3 3</b><br><b>5 5 5</b> | <b>13</b> | <b>Lần lượt chọn các khối đá 1, 9, 5 và 4 để xây tháp sẽ có độ cao là 13</b> |



## QUAY XÂU KÝ TỰ

Cho  $S$  là một chuỗi gồm không quá 80 ký tự, mỗi ký tự là một chữ cái trong bảng chữ cái tiếng Anh A-Z, a-z. Ta gọi phép dịch chuyển  $S$  đi 1 ký tự là việc chuyển ký tự đầu tiên của  $S$  xuống vị trí cuối cùng. Ký hiệu  $T(S)$  là chuỗi thu được sau phép dịch chuyển  $S$  đi một ký tự.

Cho trước chuỗi  $S$  và số nguyên dương  $N < 10^{100}$ , cần tìm chuỗi thu được sau khi thực hiện  $N$  phép dịch chuyển đối với  $S$ , tức là tìm  $T(T(\dots T(S)\dots))$  (có  $N$  chữ  $T$  trong biểu thức).

**Dữ liệu:** Vào từ file văn bản **TRANSTR.INP**:

- Dòng đầu tiên chứa số  $N$ ;
- Dòng thứ hai chứa chuỗi ký tự  $S$ .

**Kết quả:** Đưa ra file văn bản **TRANSTR.OUT** chuỗi ký tự thu được.

**Ví dụ:**

| TRANSTR.INP   | TRANSTR.OUT |
|---------------|-------------|
| 3<br>Computer | puterCom    |

## DÃ NGOẠI

Nhóm  $n$  người tổ chức một cuộc du lịch dã ngoại trên sông bằng thuyền đôi. Người thứ  $i$  trong nhóm có trọng lượng  $m_i$ . Mỗi thuyền du lịch đôi có thể chở 1 hoặc 2 người với tổng trọng lượng không quá  $d$ . Số thuyền mà hãng du lịch có là không hạn chế.

**Yêu cầu:** Hãy xác định số lượng thuyền ít nhất cần thuê.

**Dữ liệu:** đọc từ tập tin văn bản **TRIP.INP** gồm không quá 10 tests, mỗi test trên 2 dòng

- Dòng đầu tiên chứa 2 số nguyên  $n, d$  ( $1 \leq n, d \leq 10^5$ )
- Dòng thứ hai chứa  $n$  số nguyên  $m_1, m_2, \dots, m_n$  ( $1 \leq m_i \leq d$ )

**Kết quả:** xuất ra tập tin văn bản **TRIP.OUT** mỗi test trên một dòng dưới dạng số nguyên

**Ví dụ**

| TRIP.INP            | TRIP.OUT |
|---------------------|----------|
| 4 135               | 2        |
| 50 74 60 82         | 4        |
| 6 135               |          |
| 50 120 74 60 100 82 |          |

## SỐ SINH ĐÔI

Trong lý thuyết số hai số nguyên tố  $p$  và  $q$  được gọi là cặp số nguyên tố sinh đôi nếu  $q - p = 2$ .

Ví dụ, các cặp số  $(3, 5)$ ,  $(11, 13)$ ,  $(17, 19)$  là các cặp sinh đôi. Trong trường hợp tổng quát, với số nguyên dương  $k$  cho trước, cặp số nguyên tố  $p$  và  $q$  được gọi là sinh đôi (tổng quát) nếu  $q - p = k$ . Ví dụ, với  $k = 4$  cặp số nguyên tố  $(3, 7)$  được gọi là sinh đôi tổng quát.

Tồn tại giả thuyết là các cặp số nguyên tố sinh đôi nhiều vô hạn. Tuy nhiên, điều đó chưa được chứng minh. Dễ dàng thấy rằng với  $k$  cho trước, xác định số cặp sinh đôi tổng quát trong tập số tự nhiên là bài toán phức tạp không kém việc xác định số lượng cặp sinh đôi theo định nghĩa thông thường.

Ở đây chúng ta xét một bài toán đơn giản hơn.

**Yêu cầu:** Cho  $n$  và  $k$  ( $1 \leq n, k \leq 10^6$ ). Hãy xác định số cặp sinh đôi tổng quát trong phạm vi từ 1 đến  $n$ .

**Dữ liệu:** Vào từ file văn bản **TWINS.INP** gồm một dòng chứa 2 số nguyên  $n$  và  $k$ .

**Kết quả:** Đưa ra file văn bản **TWINS.OUT** một số nguyên – số lượng cặp tìm được.

**Ví dụ:**

| TWINS.INP |
|-----------|
| 17 2      |

| TWINS.OUT |
|-----------|
| 3         |

## TÔ MÀU

Jimmy chia bảng kích thước  $m \times n$  thành lưới ô vuông kích thước ô là  $1 \times 1$  và tô mỗi ô bằng một trong hai màu đen hoặc trắng sao cho không có 2 ô kề cạnh nào có cùng màu. Các hàng của lưới được đánh số từ dưới lên trên từ 1 đến  $m$ , các cột được đánh số từ 1 đến  $n$  từ trái qua phải.

Rôn muốn biết số lượng ô màu đen so với số lượng ô màu trắng như thế nào và nhận thấy rằng chỉ cần biết màu ở ô thứ  $j$  trên dòng  $i$  là có thể có ngay câu trả lời.

**Yêu cầu:** Cho  $m, n, i, j$  và  $c$ , trong đó  $c$  là màu ô thứ  $j$  trên dòng  $i$ ,  $c = 0$  là màu đen, còn  $c = 1$  là màu trắng ( $1 \leq m, n \leq 10^9$ ). Hãy xác định tương quan giữa số ô màu trắng và số ô màu đen, đưa ra câu trả lời:

- “**black**” nếu số ô màu đen nhiều hơn,
- “**white**” nếu số ô màu trắng nhiều hơn,
- “**equal**” nếu số ô của 2 màu là như nhau.

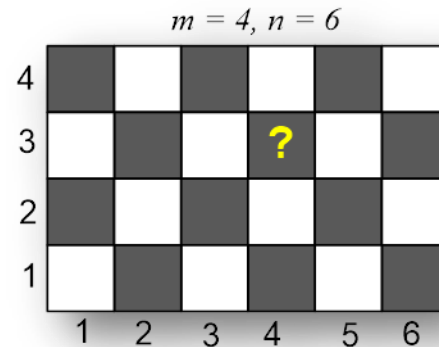
**Dữ liệu:** vào từ tập tin văn bản **W\_B.INP** gồm một dòng chứa 5 số nguyên  $m, n, i, j$  và  $c$ .

**Kết quả:** ghi ra tập tin văn bản **W\_B.OUT** câu trả lời tương ứng.

**Ví dụ:**

| W_B.INP   |
|-----------|
| 3 5 1 1 0 |

| W_B.OUT |
|---------|
| black   |



## DÂY DẪN

Cho  $n$  đoạn dây điện ( $1 \leq n \leq 10^5$ ). Đoạn thứ  $i$  có độ dài  $l_i$  ( $0 < l_i < 10^9$ ). Cần phải cắt các đoạn đã cho thành các đoạn sao cho có được  $k$  đoạn dây bằng nhau. Có thể không cần cắt hết các đoạn dây đã cho. Mỗi đoạn dây bị cắt có thể có phần còn thừa khác 0.

**Yêu cầu:** xác định độ dài lớn nhất của đoạn dây có thể nhận được. Nếu không có cách cắt thì đưa ra số 0.

**Dữ liệu:** vào từ tập tin văn bản **WIRES.INP** gồm nhiều tests, mỗi test trên một nhóm dòng

- Dòng đầu tiên chứa 2 số nguyên  $n$  và  $k$
- Dòng thứ  $i$  trong  $n$  dòng tiếp theo chứa số nguyên  $l_i$

**Kết quả:** ghi ra tập tin văn bản **WIRES.OUT**, kết quả mỗi test đưa ra trên một dòng dưới dạng số nguyên.

**Ví dụ:**

| WIRES.INP |
|-----------|
| 4 11      |
| 802       |
| 743       |
| 547       |
| 539       |

| WIRES.OUT |
|-----------|
| 200       |

## TỪ DÀI NHẤT

Cho xâu ký tự  $S$  chỉ gồm các chữ cái in hoa và dấu cách. Một dãy liên tiếp các chữ cái in hoa trong xâu  $S$  được gọi là một từ.

**Yêu cầu:** Hãy cho biết có bao nhiêu từ trong xâu ký tự và chiều dài của từ dài nhất.

**Dữ liệu:** vào từ tập tin văn bản **WORDS.INP** gồm một dòng chứa xâu ký tự  $S$  gồm không quá  $10^6$  ký tự.

**Kết quả:** ghi ra tập tin văn bản **WORDS.OUT** hai số nguyên là số từ trong xâu và độ dài của từ dài nhất.

**Ví dụ:**

| WORDS.INP        |
|------------------|
| TOP OF THE WORLD |

| WORDS.OUT |
|-----------|
| 4 5       |