

Bài 1: Mật khẩu – PASS.*

Để bảo vệ tài khoản mạng của mình, Tí dùng một mật khẩu là một dãy nhị phân độ dài N . Nhưng việc nhớ một dãy nhị phân là tương đối khó khăn mà lưu lại thì Tí sợ người khác nhìn thấy và bị lộ. Vì vậy, Tí quyết định thay vì lưu dãy nhị phân Tí lưu lại một dãy số nguyên dương ngẫu nhiên và mỗi lần cần nhập mật khẩu Tí cho chạy một chương trình để biến dãy số của mình thành mật khẩu với quy định: nếu số thứ i trong dãy N số nguyên đó là số nguyên tố thì bit thứ i trong mật mã của Tí là 1, ngược lại thì bit 0. Một ngày nọ chương trình của Tí bị lỗi nên Tí không thể tái tạo lại mật khẩu của mình thông qua dãy số đã lưu. Bạn hãy giúp Tí viết lại chương trình trên.

Input: Cho trong tệp PASS.INP có cấu trúc: Dòng đầu tiên ghi số N ($N \leq 10^4$); N dòng tiếp theo, dòng thứ i chứa một số nguyên A_i ($1 \leq A_i \leq 10^9$)

Output: Ghi ra tệp PASS.OUT gồm N dòng, dòng thứ i ghi bit thứ i của mật mã.

Ví dụ:

PASS.INP	PASS.OUT
6	0
1	1
3	0
14	1
31	0
36	1
7331	

Bài 2: Đếm số nguyên tố trong đoạn – PCOUNTAB.*

Với hai số nguyên dương A, B cho trước ($A \leq B$). Đếm số lượng các số nguyên tố thuộc đoạn $[A, B]$.

Input: Tệp PCOUNTAB.INP gồm: Dòng đầu tiên ghi số k ($1 \leq k \leq 10^3$) là số các đoạn $[A, B]$; k dòng tiếp theo, mỗi dòng ghi 2 số A, B . ($1 \leq A \leq B \leq 10^7$)

Output: Tệp PCOUNTAB.OUT gồm k dòng, dòng thứ i ghi một số là số các số nguyên tố trong đoạn $[A, B]$ thứ i đã cho.

Ví dụ:

PCOUNTAB.INP	PCOUNTAB.OUT
3	8
2 20	17
20 100	25
1 100	

Bài 3: Số nguyên tố ghép – MPRIME.*

Xét dãy A các số nguyên tố
2, 3, 5, 7, 11, 13, 17, 19,...

và dãy B gồm các số thu được từ dãy A bằng cách ghép hai số liên tiếp trong A :

23, 57, 1113, 1719, ...

Trong dãy B có những phần tử là số nguyên tố. Chẳng hạn 23, 3137, 8389, 157163...

Các số nguyên tố trong dãy B gọi là *số nguyên tố ghép*.

Yêu cầu: Cho trước số nguyên dương $K \leq 500$, hãy tìm số nguyên tố ghép thứ K.

Input: Tập MPRIME.INP gồm 1 dòng duy nhất ghi số nguyên dương K

Output: Tập MPRIME.OUT gồm 1 số duy nhất là số nguyên tố ghép thứ K.

Ví dụ:

MPRIME.INP	MPRIME.OUT
2	3137

Bài 4. Tính F - FCAL.*

Gọi $Pri(x)$ là tập các ước nguyên tố của x. Ví dụ: $Pri(9) = \{3\}$, $Pri(10) = \{2, 5\}$

Gọi $g(x,p)$ là số nguyên dương lớn nhất p^k sao cho x chia hết cho p^k (k là một số nguyên)

Gọi $f(x,y)$ là tích của các $g(y,p)$ với mọi p thuộc $Pri(x)$

Ví dụ: $f(30,70) = g(70,2) \cdot g(70,3) \cdot g(70,5) = 2^1 \cdot 3^0 \cdot 5^1 = 10$

Yêu cầu: Cho trước hai số nguyên dương x và n. Hãy tính tích $f(x,1) \cdot f(x,2) \cdot \dots \cdot f(x,n)$

Input: vào từ tệp FCAL.INP gồm một dòng ghi lần lượt hai số x và n ($2 \leq x \leq 10^9$, $1 \leq n \leq 10^{18}$)

Output: ghi ra tệp FCAL.OUT gồm 1 dòng ghi một số là tích tính được. Vì tích này có thể rất lớn nên chỉ cần ghi ra phần dư của tích khi chia cho 10^9+7

Ví dụ:

FCAL . INP	FCAL . OUT
10 2	2

Bài 5. Nguyên tố kép - DPRIME.*

Một số được gọi là *nguyên tố kép* nếu nó có ít nhất hai ước nguyên tố và hiệu của ước nguyên tố lớn nhất của nó với ước nguyên tố nhỏ nhất của nó cũng là một số nguyên tố. Ví dụ số 15 có hai ước nguyên tố là 3 và 5, hiệu của ước nguyên tố lớn nhất của 15 với ước nguyên tố nhỏ nhất của 15 là $5-3=2$ cũng là một số nguyên tố nên số 15 được gọi là số *nguyên tố kép*; số 105 không phải là số *nguyên tố kép* vì 105 có ba ước nguyên tố là 3, 5 và 7, hiệu của ước nguyên tố lớn nhất của 105 với ước nguyên tố nhỏ nhất của 105 là $7-3=4$ không phải là số nguyên tố.

Yêu cầu: cho trước một đoạn $[A,B]$, đếm số lượng số *nguyên tố kép* thuộc đoạn này.

Input: vào từ tệp văn bản DPRIME.INP có cấu trúc như sau:

- Dòng đầu ghi số nguyên dương T là số test.
- T dòng tiếp theo, dòng thứ i ghi hai số nguyên dương A_i và B_i ($A_i \leq B_i$) cho biết đoạn thứ i cần tính.

Output: ghi ra tệp văn bản DPRIME.OUT gồm T dòng, dòng thứ i ghi số lượng số *nguyên tố kép* thuộc đoạn thứ i cho trong tệp dữ liệu vào.

Ví dụ:

DPRIME.INP	DPRIME.OUT
3	0
2 6	1
5 10	4
10 20	

Ràng buộc:

- 30% số tests tương ứng với 30% số điểm có $1 \leq T \leq 10, 0 \leq Bi-Ai \leq 100, Bi \leq 200$
- 30% số tests tương ứng với 30% số điểm có $1 \leq T \leq 10, 0 \leq Bi-Ai \leq 100, Bi \leq 1000$
- 20% số tests tương ứng với 20% số điểm có $T \leq 100, 0 \leq Bi-Ai \leq 10000$
- 20% số tests còn lại tương ứng với 50% số điểm có $T \leq 1000000, Bi \leq 200000$

Bài 6. Tích các ước - MULOFDIVS.*

Cho số nguyên dương N được biểu diễn dưới dạng tích của các số nguyên tố như sau:

$$N = P_1 * P_2 * \dots * P_M$$

Yêu cầu: Tính tích của các ước của N .

Input: vào từ tệp văn bản MULOFDIVS.INP có cấu trúc như sau:

- Dòng đầu ghi số nguyên dương M là số lượng các số nguyên tố trong biểu diễn của N
- Dòng tiếp theo ghi M số cách nhau bởi dấu cách, số thứ i chính là P_i

Output: ghi ra tệp văn bản MULOFDIVS.OUT gồm 1 dòng, ghi một số là tích của các ước của N . Vì tích này có thể rất lớn nên chỉ cần ghi ra phần dư của tích khi chia cho 10^9+7

Ví dụ:

MULOFDIVS.INP	MULOFDIVS.OUT
2	100
2 5	

Ràng buộc:

- 30% số tests tương ứng với 30% số điểm có $1 \leq M \leq 5, 2 \leq P_i \leq 10$
- 20% số tests tương ứng với 20% số điểm có $5 \leq M \leq 10, 2 \leq P_i \leq 15$
- 50% số tests còn lại tương ứng với 50% số điểm có $11 \leq M \leq 2 \times 10^5, 2 \leq P_i \leq 2 \times 10^5$