

Bài 1: Mật khẩu – PASS.PAS

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.PAS

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.PAS

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. Số siêu nguyên tố - SPRIME.PAS

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 ($0 < a, b < 10^7$)

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 5. Đếm các thừa số nguyên tố - COUNTPRD.PAS

Cho số nguyên dương N ($2 \leq N < 10^{10}$), hỏi có bao nhiêu số nguyên tố khác nhau trong phân tích N ra tích các thừa số nguyên tố.

Ví dụ: $10 = 2 \cdot 5$. Có 2 số nguyên tố khác nhau trong phân tích 10 ra tích các thừa số nguyên tố là: 2 và 5.

Input: Vào từ file văn bản COUNTPRD.INP gồm một dòng chứa số nguyên N .

Output: Đưa ra file văn bản COUNTPRD.OUT số các số nguyên tố. Ví dụ:

COUNTPRD.INP	COUNTPRD.OUT
10	2

Bài 6. Đếm ước nguyên tố - CPRDIV.PAS

Cho hai số nguyên dương M và N ($1 \leq M \leq N \leq 60000$) và số S được xác định bằng công thức sau: $S = n!/(m!(n-m)!)$

Yêu cầu: Đếm số lượng ước nguyên tố của S

Input: Tập CPRDIV.INP gồm một dòng ghi hai số N và M cách nhau một dấu cách.

Output: Tập CPRDIV.OUT gồm một dòng ghi 1 số duy nhất là số lượng ước nguyên tố của S. Ví dụ:

CPRDIV.INP	CPRDIV.OUT
7 3	2