

1. Tổng ước số

Với mọi số tự nhiên m ($m \geq 2$) đều có ít nhất hai ước dương khác nhau đó là 1 và chính nó. Ta kí hiệu $s(m)$ là tổng của hai ước dương nhỏ nhất (khác nhau) của m .

Ví dụ: $s(3) = 1+3 = 4$; $s(8) = 1+2 = 3$.

Cho dãy số nguyên dương a_1, a_2, \dots, a_n và cặp chỉ số i, j ($1 \leq i \leq j \leq n$).

Yêu cầu: Tính tổng $s(a_i) + s(a_{i+1}) + \dots + s(a_j)$.

Dữ liệu vào từ file văn bản SUMDIV.INP như sau:

- Dòng 1 ghi hai số nguyên dương n và T tương ứng là số số hạng của dãy và số testcase.
- Dòng 2 ghi n số nguyên dương a_1, a_2, \dots, a_n .
- T dòng tiếp theo, mỗi dòng là một testcase tương ứng là một cặp chỉ số i, j ($1 \leq i \leq j \leq n$).

Kết quả ghi ra file văn bản SUMDIV.OUT gồm T dòng, mỗi dòng là kết quả của testcase tương ứng.

Ví dụ:

SUMDIV.INP	SUMDIV.OUT
5 3	10
2 3 8 9 100	4
1 3	10
4 4	
3 5	

Giới hạn:

- 30% số test ứng với $n \leq 100$ và $T \leq 10$, $2 \leq a_i \leq 10^7$;
- 30% số test khác ứng $n \leq 100000$, $T \leq 1000$ và $2 \leq a_i \leq 10^4$;
- 40% số test còn lại ứng với $n \leq 100000$, $T \leq 10^4$; $2 \leq a_i \leq 10^7$.

2. Số đặc biệt

An rất yêu thích số nguyên tố, đồng thời cũng rất yêu thích số 5. Do đó, cậu ta luôn coi các số nguyên tố có tổng các chữ số chia hết cho 5 là số đặc biệt. Lần này, thầy giáo đưa cho An 2 số nguyên dương L, R ($L \leq R$). An rất muốn biết trong đoạn $[L, R]$ có bao nhiêu số đặc biệt nên nhờ các bạn trả lời giúp.

Dữ liệu: Vào từ file văn bản SPRIME.INP

- Dòng đầu tiên chứa số nguyên dương $T \leq 100$ là số lượng test trong file.
- T dòng tiếp theo, mỗi dòng chứa hai số nguyên dương L, R ($L \leq R$) theo thứ tự, phân tách nhau bởi dấu cách.

Kết quả: Đưa ra file văn bản **SPRIME.OUT** T dòng, mỗi dòng ghi một số là số lượng số đặc biệt trong đoạn $[L, R]$, tương ứng theo thứ tự trong file input. Dòng thứ i trong file output là kết quả của cặp số $[L, R]$ ở dòng $i + 1$ trong file input.

Ví dụ:

SPRIME.INP	SPRIME.OUT
2	1
1 10	2
4 20	

Giải thích:

- Trong đoạn $[1, 10]$ có 1 số đặc biệt là 5.
- Trong đoạn $[4, 20]$ có 2 số đặc biệt là 5 và 19 ($1+9 = 10$).

Giới hạn:

- 20% số test có $T = 1$; $L \leq R \leq 20$
- 20% số test tiếp theo có $T = 1$; $L, R \leq 10^3$
- 30% số test tiếp theo có $2 \leq T \leq 10$; $L, R \leq 10^5$
- 30% số test cuối cùng có $10 \leq T \leq 100$; $0 < L, R \leq 3 \cdot 10^6$

3. Số chính phương

Để tạo niềm vui cho mọi người, chính quyền quyết định lắp đặt một thiết bị ở nơi công cộng. Thiết bị này giao tiếp với mọi người thông qua bàn phím và màn hình và có một số nguyên lưu bên trong bộ nhớ của nó. Ban đầu số nguyên này khởi đầu bằng 1.

Thiết bị hoạt động như sau:

- Một người gõ một số nguyên từ bàn phím
- Thiết bị sẽ nhân số trong bộ nhớ của nó với số nguyên vừa gõ và kết quả được lưu lại vào chính bộ nhớ này.
- Thiết bị sẽ hiển thị lời chào lên màn hình nếu như số trong bộ nhớ là số chính phương. Khi đó người gõ số sẽ được nhiều may mắn.

Viết chương trình, cho biết dãy số nguyên mà những người chơi lần lượt gõ, xác định xem người chơi nào sẽ là người may mắn.

Dữ liệu: Vào từ file SQRNUM.INP

- Dòng đầu tiên chứa số nguyên N ($1 \leq N \leq 500000$) là số lượng người tham gia giao tiếp với thiết bị.
- Tiếp theo là N dòng, mỗi dòng ghi một số nguyên được gõ bởi một người theo thứ tự giao tiếp với thiết bị. Các số nguyên này nằm giữa 1 và 10^6 .

Chú ý rằng kết quả số trong bộ nhớ có thể vượt quá kiểu số nguyên 64 bit

Kết quả: Ghi ra file văn bản SQRNUM.OUT

Gồm N dòng là kết quả mà mỗi người nhận được theo thứ tự. Ghi "YES" nếu kết quả là số chính phương và "NO" trong trường hợp ngược lại.

Ví dụ:

SQRNUM.INP	SQRNUM.OUT
7	NO
2	NO
3	YES
6	NO
15	NO
35	YES
21	YES
64	

Subtasks:

- Subtask 1: Số trong bộ nhớ không vượt kiểu 64 bits [50%]
- Subtask 2: Số trong bộ nhớ vượt kiểu 64 bits [50%]