



Packing Biscuits (biscuits)

Aunty Khong đang tổ chức một cuộc thi với x người tham gia và muốn tặng mỗi người tham gia một **túi bánh quy**. Có k loại bánh quy khác nhau, được đánh số từ 0 đến $k - 1$. Loại bánh quy i ($0 \leq i \leq k - 1$) có **giá trị độ ngon** là 2^i . Aunty Khong có $a[i]$ (có thể bằng không) bánh quy loại i trong tủ đựng thức ăn của mình.

Mỗi túi của Aunty Khong sẽ chứa không hoặc nhiều bánh quy mỗi loại. Tổng số bánh quy loại i trong tất cả các túi không được vượt quá $a[i]$. Tổng giá trị độ ngon của tất cả bánh quy trong túi được gọi là **tổng độ ngon** của túi.

Hãy giúp Aunty Khong tính số lượng giá trị khác nhau của y để có thể đóng gói x túi bánh quy, mỗi túi có tổng độ ngon bằng y .

Chi tiết cài đặt

Bạn cần cài đặt hàm sau:

```
int64 count_tastiness(int64 x, int64[] a)
```

- x : số túi bánh quy cần đóng gói.
- a : mảng có độ dài k . Với mỗi $0 \leq i \leq k - 1$, $a[i]$ là số lượng bánh quy loại i trong tủ đựng thức ăn.
- Hàm sẽ trả về số lượng giá trị khác nhau của y , sao cho Aunty có thể đóng gói x túi bánh quy, mỗi túi có tổng độ ngon bằng y .
- Hàm được gọi tất cả q lần (xem phần Ràng buộc và Subtask để biết các giá trị cho phép của q). Mỗi lần gọi như vậy được coi là một kịch bản riêng biệt.

Ví dụ

Ví dụ 1

Xét lời gọi hàm sau:

```
count_tastiness(3, [5, 2, 1])
```

Điều này có nghĩa là Aunty muốn đóng gói 3 túi và có 3 loại bánh quy trong tủ đựng thức ăn:

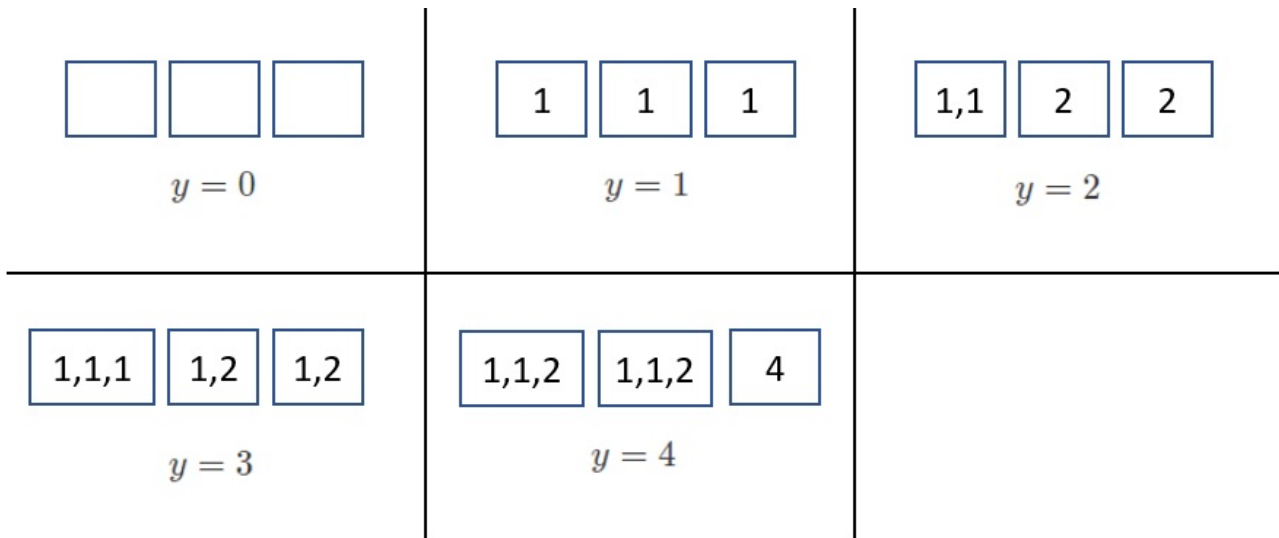
- Có 5 bánh quy loại 0, mỗi cái có giá trị độ ngon bằng 1,

- Có 2 bánh quy loại 1, mỗi cái có giá trị độ ngon bằng 2,
- Có 1 bánh quy loại 2, có giá trị độ ngon bằng 4.

Các giá trị có thể có của y là $[0, 1, 2, 3, 4]$. Ví dụ, để đóng gói 3 túi có tổng độ ngon là 3, Aunty có thể đóng gói:

- một túi chứa ba bánh quy loại 0 và
- hai túi, mỗi túi chứa một bánh quy loại 0 và một bánh quy loại 1.

Do có thể có 5 giá trị khác nhau của y nên hàm cần trả về 5.



Ví dụ 2

Xét lời gọi hàm sau:

```
count_tastiness(2, [2, 1, 2])
```

Điều này có nghĩa là Aunty muốn đóng gói 2 túi và có 3 loại bánh quy trong tủ đựng thức ăn:

- Có 2 bánh quy loại 0, mỗi cái có giá trị độ ngon bằng 1,
- Có 1 bánh quy loại 1, có giá trị độ ngon bằng 2,
- Có 2 bánh quy loại 2, mỗi cái có giá trị độ ngon bằng 4.

Các giá trị có thể có của y là $[0, 1, 2, 4, 5, 6]$. Do có thể có 6 giá trị của y , hàm cần trả về 6.

Ràng buộc

- $1 \leq k \leq 60$
- $1 \leq q \leq 1000$
- $1 \leq x \leq 10^{18}$
- $0 \leq a[i] \leq 10^{18}$ (với mọi $0 \leq i \leq k - 1$)
- Với mỗi lần gọi hàm `count_tastiness`, tổng giá trị độ ngon của tất cả bánh quy trong tủ đựng thức ăn không vượt quá 10^{18} .

Subtask

1. (9 điểm) $q \leq 10$ và đối với mỗi lần gọi hàm `count_tastiness`, tổng giá trị độ ngon của tất cả bánh quy trong tủ đựng thức ăn không vượt quá 100 000.
2. (12 điểm) $x = 1, q \leq 10$
3. (21 điểm) $x \leq 10\,000, q \leq 10$
4. (35 điểm) Giá trị chính xác trả về của mỗi lần gọi hàm `count_tastiness` không vượt quá 200 000.
5. (23 điểm) Không có ràng buộc gì thêm.

Trình chấm mẫu

Trình chấm mẫu đọc dữ liệu vào theo định dạng sau. Dòng đầu chứa số nguyên q . Tiếp theo là q cặp dòng, mỗi cặp mô tả một kịch bản duy nhất theo định dạng sau:

- dòng 1: $k \ x$
- dòng 2: $a[0] \ a[1] \ \dots \ a[k - 1]$

Kết quả ra của trình chấm mẫu có định dạng sau:

- dòng i ($1 \leq i \leq q$): giá trị trả về của hàm `count_tastiness` cho kịch bản thứ i trong dữ liệu vào.