



Packing Biscuits (biscuits)

Teta Khong organizuje takmičenje sa x učesnika, i želi da svakom učesniku pokloni **Kesu keksića**. Postoje k različite vrste keksića, numerisane od 0 do $k - 1$. Svaki keksić tipa i ($0 \leq i \leq k - 1$) ima **vrijednost ukusnosti** od 2^i . Teta Khong ima u svojoj ostavi $a[i]$ (moguće i nula) keksića tipa i .

Svaka torba tetke Khong sadržaće nula ili više keksića svake vrste. Ukupan broj keksića tipa i u svim vrećicama ne smije biti veći od $a[i]$. Zbir vrednosti ukusa svih keksića u vrećici naziva se **vrijednost ukusnosti** vrećice.

Pomozi teti Khong da sazna koliko različitih vrijednosti y postoji, tako da je moguće spakovati x vrećica keksića, od kojih svaka ima ukupnu ukusnost jednaku y .

Detalji implementacije

Potrebno je implemetirati sljedeću funkciju:

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

- x : broj vrećica keksića za pakovanje.
- a : niz dužine k . Za $0 \leq i \leq k - 1$, $a[i]$ označava broj keksića tipa i u ostavi.
- Funkcija treba da prikaže broj različitih vrijednosti y , tako da teta Khong može da spakuje x vrećica keksića, od kojih svaka ima ukupnu ukusnost od y .
- Funkcija se poziva ukupno q puta (pogledaj odjeljke Ograničenja i Podzadaci za dozvoljene vrijednosti q). Svaki od ovih poziva treba tretirati kao zaseban scenario.

Primjeri

Primjer 1

Razmotrite sljedeći poziv:

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

Ovo znači da teta Khong želi da spakuje 3 torbe, a u ostavi postoje 3 vrste keksića:

- 5 keksića tipa 0, svaki sa vrijednosti ukusnosti od 1,
- 2 keksića tipa 1, svaki sa vrijednosti ukusnosti od 2,
- 1 keksić tipa 2, svaki sa vrijednosti ukusnosti od 4.

Moguće vrijednosti y su $[0, 1, 2, 3, 4]$. Na primjer, da bi spakovala 3 vrećice ukupne ukusnosti 3, teta Khong može da spakuje:

- jednu vrećicu koje sadrži tri keksića tipa 0, i
- dvije vrećice, svaka sadrži po jedan keksić tipa 0 i jedan keksić tipa 1.

Pošto postoji 5 mogućih vrijednosti y , funkcija treba da vrati 5.

<div> <div></div> <div></div> <div></div> </div> $y = 0$	<div> <div>1</div> <div>1</div> <div>1</div> </div> $y = 1$	<div> <div>1,1</div> <div>2</div> <div>2</div> </div> $y = 2$
<div> <div>1,1,1</div> <div>1,2</div> <div>1,2</div> </div> $y = 3$	<div> <div>1,1,2</div> <div>1,1,2</div> <div>4</div> </div> $y = 4$	

Primjer 2

Razmotrite sledeći poziv:

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

Ovo znači da teta Khong želi da spakuje 2 torbe, a u ostavi postoje 3 vrste keksića:

- 2 keksića tipa 0, svaki sa vrijednosti ukusnosti od 1,
- 1 keksić tipa 1, vrijednosti ukusnosti od 2,
- 2 keksića tipa 2, svaki sa vrijednosti ukusnosti od 4.

Moguće vrijednosti y su $[0, 1, 2, 4, 5, 6]$. Pošto postoji 6 mogućih vrijednosti y , funkcija treba da vrati 6.

Ograničenja

- $1 \leq k \leq 60$
- $1 \leq q \leq 1000$
- $1 \leq x \leq 10^{18}$
- $0 \leq a[i] \leq 10^{18}$ (za sve $0 \leq i \leq k - 1$)
- Za svaki poziv `count_tastiness`, zbir vrijednosti ukusa svih keksića u ostavi ne prelazi 10^{18} .

Podzadaci

1. (9 bodova) $q \leq 10$, and for each call to `count_tastiness`, the sum of tastiness values of all biscuits in the pantry does not exceed 100 000.
2. (12 bodova) $x = 1, q \leq 10$
3. (21 bod) $x \leq 10\,000, q \leq 10$
4. (35 bodova) Tačna povratna vrijednost svakog poziva na `count_tastiness` ne prelazi 200 000.
5. (23 boda) Nema dodatnih ograničenja.

Program za ocjenjivanje (grader)

Program za ocjenjivanje (grader) čita ulaz u sledećem formatu. Prvi red sadrži cio broj q . Nakon toga slijede q parova redova, a svaki par opisuje jedan scenario u sljedećem formatu:

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

Rezultat programa za ocjenjivanje (grader) je u sledećem formatu:

- red i ($1 \leq i \leq q$): vraća vrijednost `count_tastiness` za i -ti scenario u ulazu.