

# Pakowanie herbatników (biscuits)

Ciocia Khong organizuje zawody dla  $x$  zawodników i chce każdemu zawodnikowi wręczyć **torbę herbatników**. Mamy  $k$  różnych rodzajów herbatników, ponumerowanych od 0 do  $k - 1$ . Każdy herbatnik rodzaju  $i$  ( $0 \leq i \leq k - 1$ ) ma **smakowitość** równą  $2^i$ . Ciocia Khong ma  $a[i]$  (być może zero) herbatników rodzaju  $i$  w swojej spiżarni.

Każda z toreb cioci Khong będzie zawierać zero lub więcej herbatników każdego rodzaju. Liczba wszystkich herbatników rodzaju  $i$  we wszystkich torbach nie może przekroczyć  $a[i]$ . **Całkowitą smakowitością** torby nazwiemy sumę smakowitości wszystkich herbatników w torbie.

Pomóż cioci Khong dowiedzieć się, ile istnieje różnych wartości  $y$ , dla których jest możliwe spakowanie  $x$  toreb herbatników tak, aby całkowita smakowitość każdej z tych toreb wynosiła  $y$ .

## Szczegóły implementacyjne

Twoim zadaniem jest napisanie następującej funkcji:

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

- $x$ : liczba toreb herbatników, które należy spakować.
- $a$ : tablica rozmiaru  $k$ . Dla  $0 \leq i \leq k - 1$ ,  $a[i]$  oznacza liczbę herbatników rodzaju  $i$  w spiżarni.
- Funkcja ta powinna zwrócić liczbę różnych wartości  $y$ , dla których ciocia jest w stanie spakować  $x$  toreb z herbatnikami, gdzie każda z nich ma całkowitą smakowitość równą  $y$ .
- Funkcja ta zostanie wywołana  $q$  razy (sprawdź sekcje Ograniczenia oraz Podzadania, aby poznać dozwolone wartości  $q$ ). Każde z tych wywołań powinno być traktowane jako osobny scenariusz.

## Przykłady

### Przykład 1

Rozważmy następujące wywołanie:

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

Oznacza ono, że ciocia chce spakować 3 torby oraz mamy 3 rodzaje herbatników w spiżarni:

- 5 herbatników rodzaju 0, każdy o smakowitości 1,
- 2 herbatniki rodzaju 1, każdy o smakowitości 2,
- 1 herbatnik rodzaju 2 o smakowitości 4.

Możliwe wartości  $y$  to  $[0, 1, 2, 3, 4]$ . Na przykład, aby spakować 3 torby z herbatnikami o całkowitej smakowitości 3, ciocia może spakować:

- jedną torbę zawierającą trzy herbatniki rodzaju 0 oraz
- dwie torby, z których każda zawiera jeden herbatnik rodzaju 0 i jeden herbatnik rodzaju 1.

Jako że mamy 5 możliwych wartości  $y$ , funkcja powinna zwrócić 5.

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

## Przykład 2

Rozważmy następujące wywołanie:

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

Oznacza ono, że ciocia chce spakować 2 torby oraz mamy 3 rodzaje herbatników w spiżarni:

- 2 herbatniki rodzaju 0, każdy o smakowitości 1,
- 1 herbatnik rodzaju 1 o smakowitości 2,
- 2 herbatniki rodzaju 2, każdy o smakowitości 4.

Możliwe wartości  $y$  to  $[0, 1, 2, 4, 5, 6]$ . Jako że mamy 6 możliwych wartości  $y$ , funkcja powinna zwrócić 6.

## Ograniczenia

- $1 \leq k \leq 60$
- $1 \leq q \leq 1000$
- $1 \leq x \leq 10^{18}$

- $0 \leq a[i] \leq 10^{18}$  (dla  $0 \leq i \leq k - 1$ )
- Dla każdego wywołania `count_tastiness`, suma smakowości wszystkich herbatników w spiżarni nie przekracza  $10^{18}$ .

## Podzadania

1. (9 punktów)  $q \leq 10$  i dla każdego wywołania `count_tastiness`, suma smakowości wszystkich herbatników w spiżarni nie przekracza 100 000.
2. (12 punktów)  $x = 1, q \leq 10$
3. (21 punktów)  $x \leq 10\,000, q \leq 10$
4. (35 punktów) Poprawna wartość, która powinna być zwrócona dla każdego wywołania `count_tastiness` nie przekracza 200 000.
5. (23 punkty) Brak dodatkowych ograniczeń.

## Przykładowy program oceniający

Przykładowy program oceniający wczytuje wejście w następującym formacie. Pierwszy wiersz zawiera liczbę całkowitą  $q$ . Po nim następuje  $q$  par wierszy, gdzie każda para opisuje pojedynczy scenariusz w następującym formacie:

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

Przykładowy program oceniający wypisuje wyjście w następującym formacie:

- wiersze  $i$  ( $1 \leq i \leq q$ ): wartość zwrócona przez `count_tastiness` dla  $i$ -tego scenariusza na wejściu.