

## Bài 1: Đếm dãy chia hết

Cho một dãy số nguyên dương, đếm số lượng dãy con liên tiếp có tổng chia hết cho  $d$ . Hai dãy con được gọi là khác nhau nếu ít nhất một trong hai điểm đầu hoặc điểm cuối hai dãy con đó trong dãy đã cho là khác nhau. Ví dụ với  $d = 4$ , dãy  $(2, 1, 2, 1, 4, 1)$  có 4 dãy con thỏa mãn là  $(1, 2, 1)$ ,  $(1, 2, 1, 4)$ ,  $(4)$ ,  $(2, 1, 4, 1)$ .  $d = 2$ , dãy  $1, 1, 1, 1$  có 4 dãy con thỏa mãn.

**Dữ liệu:** Vào từ file văn bản DIVSEQ.INP

- ★ Dòng đầu tiên là số  $T$  – số lượng test ( $T \leq 100$ )
- ★  $T$  nhóm dòng tiếp theo, mỗi dòng tương ứng một yêu cầu
  - Dòng đầu là 2 số nguyên dương  $d$  và  $N$  ( $d \leq 10^6, N \leq 5 \cdot 10^4$ )
  - Dòng thứ 2 chứa  $N$  số nguyên  $\leq 10^9$  biểu diễn dãy số.

**Kết quả:** Ghi ra file văn bản DIVSEQ.OUTT dòng là kết quả các test tương ứng theo thứ tự.

Ví dụ

DIVSEQ.INP	DIVSEQ.OUT
1	4
4 6	
2 1 2 1 4 1	

Chú ý: 50% số test có  $n \leq 1000$

## Giải thuật:

### Đếm dãy chia hết

Version 1.0: Với mỗi cặp  $(i, j): 1 \leq i \leq j \leq n$  thử dãy con  $(a_i, a_{i+1}, \dots, a_j)$  nếu tổng chia hết cho  $d$  thì tăng biến đếm:

**ret=0;**

**for(i=1; i<=n; i++)**

**for(j=i; j<=n; j++) if ((s[j]-s[i-1])%d==0) ret++;**

Độ phức tạp của thuật toán là  $O(n^2)$  và được 50% số điểm của bài.

Version 2.0: Nhận xét:  $(s[i] - s[j]) \% d = 0 \Leftrightarrow s[i]$  và  $s[j]$  đồng dư khi chia cho  $d$ . Vì  $d \leq 10^5$  nên từ dãy  $s[0], s[1], \dots, s[n]$  ta có thể lập mảng  $c[0], c[1], \dots, c[d-1]$  với  $c[i] =$  số lượng phần tử của mảng  $s$  có số dư khi chia cho  $d$  bằng  $i$  ( $i = 0, 1, 2, \dots, d-1$ ).

Hai phần tử của mảng  $s$  đồng dư với  $d$  cho một dãy. Vậy kết quả cần tìm là:

$$ds = \frac{c[0](c[0]-1)}{2} + \dots + \frac{c[d-1](c[d-1]-1)}{2}$$

Độ phức tạp của thuật toán giảm xuống còn  $O(n)$

## Bài 2: Số nghiệm

Đếm số lượng bộ số nguyên không âm  $(x, y, z)$  thỏa mãn:

$$\begin{cases} x^d + y^d + z^d \equiv m \pmod{N} \\ 0 \leq x, y, z \leq U \end{cases}$$

**Dữ liệu:** Vào từ file văn bản CNTSOLS.INP

- ★ Dòng đầu tiên ghi số nguyên dương  $T$  là số lượng bộ dữ liệu. Tiếp theo là  $T$  dòng, mỗi dòng ghi 4 số nguyên  $U, d, m, N$

**Kết quả:** Ghi ra file văn bản CNTSOLS.OUT

- ★ Với mỗi bộ dữ liệu in ra số lượng tìm được. Con số này có thể rất lớn nên bạn chỉ cần in phần dư của nó khi chia cho  $10^9 + 7$ . Giả thiết rằng  $0^0 = 1$

**Subtasks:**

- $1 \leq T \leq 10$
- $1 \leq U \leq 10^9$
- $0 \leq d \leq 10^9$
- $1 \leq N \leq 40$
- $0 \leq m < N$

Ví dụ:

CNTSOLS.INP	CNTSOLS.OUT	Giải thích
2	4	Case 1: (0,2,2),(2,2,0),(2,0,2),(1,1,1)
2 2 3 5	1	Case 2: (1,1,1)
1 2013 3 31		

**Hướng dẫn giải thuật:**

### Số nghiệm

Trước tiên chúng ta đếm số bộ  $(X, Y, Z)$  thỏa mãn:

$$\begin{aligned} X^d + Y^d + Z^d &\equiv m \pmod{N} \\ 0 \leq X, Y, Z &< N \end{aligned}$$

Do  $N$  nhỏ nên điều này có thể thực hiện một cách đơn giản.

Việc cuối cùng là đếm xem có bao nhiêu bộ  $(x, y, z)$  thỏa mãn:

$$\begin{aligned} x &\equiv X \pmod{N}, y \equiv Y \pmod{N}, z \equiv Z \pmod{N} \\ 0 \leq x, y, z &\leq U \end{aligned}$$

Một điều lưu ý là để tính lũy thừa  $a^d$  sử dụng kỹ thuật "chia để trị" tính nhanh:

```
int luythua(int a,int d) {  
    if (d==0) return 1;  
    int b=luythua(a,d/2);  
    b=(b*b) % N;  
    if (d%2) b=(b*a) % N;  
    return b;  
}
```

## Bài 3: Hội chợ

Hà Nội tổ chức hội chợ nông sản trên phố đi bộ Hồ Gươm. Dọc tuyến phố, ban tổ chức đã xây dựng  $m$  gian hàng liền nhau đánh số lần lượt  $1, 2, \dots, m$ . Tuy nhiên chỉ có  $n$  gian hàng trong số chúng được thuê. Gian hàng thứ  $i$  được thuê có số hiệu  $x_i$ . Không có hai gian hàng được thuê có cùng số hiệu.

Để tiết kiệm chi phí, ban tổ chức chỉ che mưa cho những gian hàng được thuê bằng những tấm bạt. Một tấm bạt phủ được từ gian hàng số hiệu  $u$  đến gian hàng số hiệu  $v$  ( $u \leq v$ ) được coi là có kích thước  $v - u + 1$ . Giá của một tấm bạt kích thước  $w$  là  $C_w$ . Chú ý rằng những tấm bạt kích thước lớn hơn không nhất thiết phải đắt hơn những tấm bạt kích thước nhỏ hơn.

Hãy giúp ban tổ chức tính số tiền ít nhất để có thể mua bạt che tất cả các gian hàng được thuê. Chú ý rằng trong phương án tối ưu các tấm bạt có thể phủ chồng lên nhau ở một số gian hàng.

**Dữ liệu:** Vào từ file văn bản MARKET.INP

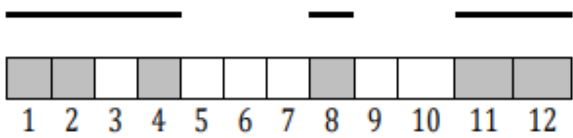
- ★ Dòng đầu ghi hai số nguyên dương  $n, m$  ( $1 \leq n \leq 5000, 1 \leq m \leq 10^5$ )
- ★ Dòng thứ hai chứa  $n$  số nguyên dương  $x_1, x_2, \dots, x_n$  ( $1 \leq x_i \leq m, x_i \neq x_j \forall i \neq j$ )
- ★ Dòng thứ ba chứa  $m$  số nguyên  $C_1, C_2, \dots, C_m$  ( $1 \leq C_i \leq 10^6$ ) là giá của những tấm bạt kích thước  $1, 2, \dots, m$

Các số liên tiếp trên cùng một dòng của file dữ liệu vào cách nhau ít nhất một dấu cách.

**Kết quả:** Ghi ra file văn bản MARKET.OUT

- ★ Một số nguyên duy nhất là chi phí nhỏ nhất tìm được.

Ví dụ:

MARKET.INP	MARKET.OUT T	Hình minh họa ví dụ
6 12 1 2 11 8 4 12 2 3 4 4 8 9 15 16 17 18 19 19	9	 <p>Có 12 gian hàng, các gian màu xám là các gian được thuê. Ở trên sử dụng ba tấm bạt kích thước 4, 1 và 2</p>

**Hướng dẫn giải thuật:**

Không mất tổng quát ta có thể coi  $x_1 < x_2 < \dots < x_n$ . Bài toán được giải theo phương pháp qui hoạch động.

Gọi  $f[i]$  là chi phí nhỏ nhất để phủ hết các gian hàng  $1, 2, \dots, i$  ta có

$$f[i] = \min\{f[k] + cp(k+1, i) : k = 0, 1, 2, \dots, i-1\} \quad (*)$$

$$f[0] = 0$$

Trong đó  $cp(u, v)$  là chi phí nhỏ nhất để phủ một tấm bạt che từ gian hàng  $x_u$  đến gian hàng  $x_v$ . Có thể tính  $cp(u, v)$  theo công thức:

$$cp(u, v) = G[v - u + 1] = \min\{c_{v-u+1}, \dots, c_m\}$$

Mảng  $G$  có thể được chuẩn bị trước trong thời gian  $O(m)$  và do vậy độ phức tạp chung của thuật toán là  $O(m + n^2)$

#### Bài 4: Tham quan Nam Định

Vova đang hào hứng được tham dự cuộc thi Olympic năm nay được tổ chức tại thành phố Nam Định. Là một người đam mê khám phá các phong cảnh đẹp, Vova đã tìm hiểu các khu du lịch ở thành phố này như khu du lịch Đền Trần, Chùa Phố Minh, Tượng đài Quốc công Tiết chế Hưng Đạo Đại Vương Trần Quốc Tuấn, ... cũng như chi phí khi vào thăm. Giả sử có  $n$  điểm du lịch đánh số  $1..n$ , điểm du lịch  $i$  có chi phí vào thăm là  $a_i$ . Hiện tại, Vova được mẹ cho số tiền  $S$ .

Bạn hãy giúp Vova **tính số cách khác nhau** có thể thăm quan sao cho tổng chi phí là  $S$ . Các cách đi là hoán vị của nhau chỉ tính là một cách. Trong mỗi cách đi, mỗi điểm du lịch được thăm không quá 1 lần.

**Dữ liệu:** Vào từ file văn bản TRAVEL.INP

- Dòng đầu chứa hai số nguyên dương  $n, S$  ( $n \leq 40, S \leq 2 \cdot 10^9$ )
- Dòng tiếp theo chứa  $n$  số nguyên  $a_1, a_2, \dots, a_n$  ( $|a_i| \leq 10^9$ )

**Kết quả:** Ghi ra file văn bản TRAVEL.OUT

Ghi một số nguyên duy nhất là số cách đi của Vova. Trong trường hợp không có cách đi nào, dữ liệu in ra số 0.

Ví dụ:

Travel.inp	Travel.out	Giải thích
4 4 1 2 3 4	2	Có hai cách đi: $\{4\}$ và $\{1, 3\}$

**Các giới hạn:**

- 30% số test đầu tiên có  $n \leq 20$  và  $S \leq 10^5$
- 30% số test tiếp theo có  $n \leq 20$  và  $S \leq 2 \cdot 10^9$
- 40% số test còn lại có  $n \leq 40$  và  $S \leq 2 \cdot 10^9$ .

#### Hướng dẫn giải thuật



Bài toán quy về đếm số tập con của tập  $\{a_1, \dots, a_n\}$  có tổng bằng  $S$ .

**Subtask1:** Quy hoạch động

Gọi  $F[i]$  là số cách đi tham quan với số tiền  $i$ .

Xét điểm tham quan  $j$  ( $j = 1..n$ ):  $F[i] = F[i] + F[i - a[j]]$

Khởi tạo:  $F[i] = 0, i = 1..n; F[0] = 1$ .

**Subtask2:** Vết cạn

Sinh tất cả các tập con có thể của tập  $\{a_1, \dots, a_n\}$ , với mỗi tập con tính tổng của tập con ấy, nếu tổng  $= S$  thì tăng *đếm* (khởi tạo *đếm* $=0$ ).

**Subtask3:** Duyệt chia đôi tập hợp

Tập 1:  $\{a_1, \dots, a_{\frac{n}{2}}\}$ , tập 2:  $\{a_{\frac{n}{2}+1}, \dots, a_n\}$ ,

Sinh tất cả các tập con có thể của tập 1, với mỗi tập con tính tổng của tập con ấy, lưu vào mảng  $S_1$ .

Sinh tất cả các tập con có thể của tập 2, với mỗi tập con tính tổng của tập con ấy, lưu vào mảng  $S_2$ .

Sắp xếp mảng  $S_1$  tăng dần. Với mỗi phần tử của mảng  $S_2$ , tìm kiếm nhị phân trên mảng  $S_1$  số phần tử có giá trị bằng  $S - S_1[i]$ .

----- **Hết** -----