

Trò chơi biến đổi số

Xét trò chơi biến đổi số như sau: Ban đầu, người chơi được cho một dãy số nguyên dương a_1, a_2, \dots, a_n (các số đôi một khác nhau) và số nguyên dương k . Người chơi cần thực hiện không quá k phép biến đổi để từ số a_1 nhận được số a_n . Mỗi lần thực hiện biến đổi, từ số a_i có thể biến đổi thành số a_j nếu $6 \times a_i + a_j$ là một số nguyên tố.

Yêu cầu: Cho dãy a_1, a_2, \dots, a_n và hai số nguyên dương k, M . Hãy đếm số cách biến đổi không quá k bước để từ a_1 nhận được a_n chia dư cho M .

Input

- Dòng đầu chứa số nguyên dương T ($T \leq 20$) là số bộ dữ liệu trong file;
- T nhóm dòng sau, mỗi nhóm có dạng:
 - o Dòng đầu của nhóm gồm 3 số nguyên dương n, k, M ($n \leq 20$);
 - o Dòng thứ hai của nhóm gồm n số nguyên dương a_1, a_2, \dots, a_n ($a_i \leq 10^9$).

Output

- Gồm T dòng, mỗi dòng chứa một số nguyên tương ứng với kết quả của bộ dữ liệu vào.

Dữ liệu vào	Kết quả ra
1 3 2 100 1 5 7	2

Subtask 1: $k, M \leq 10^6$;

Subtask 2: $k, M \leq 10^9$;

Subtask 3: $k, M \leq 10^{12}$.

Tô màu

Cho bảng $m \times n$, Minh tô màu m dòng và n cột của bảng bằng một trong hai màu màu xanh hoặc màu vàng. Ô giao giữa dòng i và cột j sẽ có màu như sau:

- Nếu dòng i , cột j cùng được tô màu xanh (hoặc vàng) thì ô (i, j) sẽ có màu xanh (hoặc vàng)
- Nếu dòng i , cột j được tô bằng hai màu khác nhau thì ô (i, j) có màu xanh lá cây.

Sau một thời gian, một số ô bị mờ đi và không thể xác định được ô đó là màu gì. Minh cũng không thể nhớ được mình đã tô màu các dòng và các cột như thế nào? Bạn hãy lập trình giúp Minh đếm xem có bao nhiêu khả năng khác nhau có thể có của bảng.

Input

- Dòng 1: gồm 2 số m, n ($1 \leq m, n \leq 30$)
- m dòng sau, mỗi dòng n số mô tả màu các ô của bảng (1-màu xanh, 2-màu vàng, 3-màu xanh lá cây, 0-không xác định được màu)

Output

- Gồm một số là số khả năng khác nhau có thể có của bảng.

Dữ liệu vào	Kết quả ra
3 4 1 0 0 0 3 0 0 0 0 0 0 0	16
2 2 3 3 3 3	1
2 2 2 2 2 3	0

bsearch

Xét chương trình con theo thuật toán tìm kiếm nhị phân như sau:

```
int binary_search(vector<int> &a, int value) {
    int l = 0;
    int r = (int)a.size() - 1;
    while (l <= r) {
        int m = (l + r + 1) / 2;
        if (a[m] == value)
            return m;
        else if (a[m] > value)
            r = m - 1;
        else
            l = m + 1;
    }
    return -1;
}
```

Yêu cầu: Đếm số hoán vị của $1, 2, \dots, n$ để chương trình con trên vẫn tìm được phần tử v ($1 \leq v \leq n$) trong dãy hoán vị dù hoán vị đó không được sắp xếp.

Input

- Gồm hai số nguyên n, v .

Output

- Số hoán vị đếm được, ghi ra số là phần dư cho $(10^9 + 7)$.

Dữ liệu vào	Kết quả ra
3 1	4

Subtask 1: $n \leq 10$;

Subtask 2: $n \leq 100$;

Subtask 3: $n \leq 10000$;

Holiday

Jian-Jia lên kế hoạch cho kỳ nghỉ tiếp theo tại Đài Loan. Trong kỳ nghỉ này, Jian-Jia di chuyển từ thành phố này đến thành phố khác và thăm các điểm du lịch tại các thành phố.

Có n thành phố tại Đài Loan, tất cả nằm dọc theo một con đường cao tốc. Các thành phố được đánh số liên tiếp bắt đầu từ 0 đến $n - 1$. Thành phố i với $0 < i < n - 1$, liền kề với thành phố $i - 1$ và $i + 1$. Thành phố duy nhất liền kề với thành phố 0 là thành phố 1 và thành phố duy nhất liền kề với thành phố $n - 1$ là thành phố $n - 2$.

Mỗi thành phố có một số điểm du lịch. Jian-Jia có d ngày cho kỳ nghỉ và lên kế hoạch để thăm được nhiều điểm du lịch nhất. Jian-Jia đã chọn một thành phố làm thành phố xuất phát cho kỳ nghỉ của mình. Mỗi ngày trong kỳ nghỉ, Jian-Jia hoặc di chuyển đến thành phố liền kề hoặc đi thăm tất cả các điểm du lịch của thành phố mà nó đang dừng chân, nhưng không thực hiện cả hai điều này. Jian-Jia sẽ không bao giờ thăm các điểm du lịch trong cùng một thành phố hai lần ngay cả khi Jian-Jia ở trong thành phố nhiều lần. Hãy giúp Jian-Jia lên kế hoạch cho kỳ nghỉ của mình để thăm được nhiều điểm du lịch khác nhau nhất.

Ví dụ, giả sử Jian-Jia có 7 ngày cho kỳ nghỉ, có 5 thành phố (được liệt kê trong bảng dưới đây) và Jian-Jia xuất phát từ thành phố 2. Trong ngày thứ nhất, Jian-Jia thăm quan 20 điểm du lịch ở thành phố 2. Trong ngày thứ hai Jian-Jia di chuyển từ thành phố 2 đến thành phố 3 và trong ngày thứ ba thăm 30 điểm du lịch ở thành phố 3. Jian-Jia dành ba ngày tiếp theo để di chuyển từ thành phố 3 đến thành phố 0 và thăm 10 điểm du lịch ở thành phố 0 vào ngày thứ bảy. Tổng số điểm du lịch mà Jian-Jia thăm là $20 + 30 + 10 = 60$, đó là số lượng lớn nhất các điểm du lịch mà Jian-Jia có thể thăm trong 7 ngày nếu xuất phát từ thành phố 2.

thành phố	số điểm du lịch
0	10
1	2
2	20
3	30
4	1

ngày	hoạt động
1	thăm các điểm du lịch ở thành phố 2
2	di chuyển từ thành phố 2 đến thành phố 3
3	thăm các điểm du lịch ở thành phố 3
4	di chuyển từ thành phố 3 đến thành phố 2
5	di chuyển từ thành phố 2 đến thành phố 1
6	di chuyển từ thành phố 1 đến thành phố 0
7	thăm các điểm du lịch ở thành phố 0

Yêu cầu: Tính số lượng nhiều nhất các điểm du lịch mà Jian-Jia có thể thăm.

Input

- Dòng 1: $n, start, d$; $(0 \leq d \leq 2n + \lfloor \frac{n}{2} \rfloor)$
- Dòng 2: $attraction[0], \dots, attraction[n - 1]$. (các số không âm và không vượt quá 10^9)

Output

- Gồm một số là lượng nhiều nhất các điểm du lịch mà Jian-Jia có thể thăm.

Dữ liệu vào	Kết quả ra
5 2 7 10 2 20 30 1	60

Subtask 1: $n \leq 20$;

Subtask 2: $n \leq 10^5$; xuất phát từ thành phố 0;

Subtask 3: $n \leq 3000$;

Subtask 4: $n \leq 10^5$;