

## Խնդիր BinSearch

Input file        stdin  
Output file      stdout

```
bool binary_search(int n, int p[], int target){
    int left = 1, right = n;
    while(left < right){
        int mid = (left + right) / 2;
        if(p[mid] == target)
            return true;
        else if(p[mid] < target)
            left = mid + 1;
        else
            right = mid - 1;
    }
    if(p[left] == target) return true;
    else return false;
}
```

Բոլորին հայտնի է, որ եթե  $p$ -ն սորտավորված է, ապա այս կոդը վերադարձնում է  $\text{true}$  այն և միայն այն դեպքում եթե  $\text{target}$ -ը հանդիպում է  $p$ -ում: Սակայն եթե  $p$ -ն սորտավորած չէ դա կարող է ոչ միշտ տեղի ունենալ:

Տրված է մի  $n$  դրական ամբողջ թիվ և մի  $b_1, \dots, b_n \in \{\text{true}, \text{false}\}$  հաջորդականություն: Երաշխավորվում է, որ  $n = 2^k - 1$  ինչ, որ դրական ամբողջ  $k$ -ի համար: Դուք պետք է կառուցեք  $p \{1, \dots, n\}$  տեղափոխությունը, որպեսզի այն բավարարի որոշակի հատկությունների: Թող  $S(p)$ -ն լինի այն ինդեքսների քանակը  $i \in \{1, \dots, n\}$  որոնց համար  $\text{binary\_search}(n, p, i)$  չի վերադարձնում  $b_i$ : Պետք է այնպես ընտրեք  $p$ -ն, որպեսզի  $S(p)$  լինի փոքր (ինչպես որ նկարագրված է “Սահմանափակումներ” բաժնում):

(Սահմանում:  $\{1, \dots, n\}$ -ի տեղափոխությունը  $n$  ամբողջ թվերի տեղափոխություն է, որտեղից  $n$  թվերը հանդիպում են  $\delta$  ժամանակ անգամ:)

## Մուտքային տվյալներ

Մուտքային տվյալների բաղկացած են մի քանի թեստերից: Առաջին տողը պարունակում է  $T$ , թեստերի քանակը: Հաջորդում են թեստերը:

Ամեն թեստի առաջին տողը պարունակում է  $n$  թիվը: Երկրորդ տողը պարունակում է  $n$  երկարության տող, ‘0’ և ‘1’ սիմվոլներից բաղկացած: Այդ սիմվոլները իրարից բացատանիշով անջատված չեն: Եթե  $i^{\text{րդ}}$  սիմվոլը ‘1’ է, ապա  $b_i = \text{true}$ , և եթե ‘0’ է, ապա  $b_i = \text{false}$ .

## Ելքային տվյալներ

Ելքային տվյալները բաղկացած են  $T$  թեստերի պատասխաններից: Ամեն թեստի պատասխանը  $p$  տեղափոխություն է, որը բավարարում է պայմաններին:

## Սահմանափակումներ

- Թող  $\sum n$ -ը լինի բոլոր  $n$ -երի գումարը մուտքային տվյալներում:
- $1 \leq \sum n \leq 100\,000$ .
- $1 \leq T \leq 7\,000$ .
- $n = 2^k - 1$  ինչ որ  $k \in \mathbb{N}$ ,  $k > 0$ .
- Եթե  $S(p) \leq 1$  բոլոր թեստերի համար մի ենթախնդրում, դուք կստանաք այդ ենթախնդրի միավորների 100%-ը:
- Հակառակ դեպքում, եթե  $0 \leq S(p) \leq \lceil \log_2 n \rceil$  (i.e.  $1 \leq 2^{S(p)} \leq n + 1$ ) բոլոր թեստերի համար մի ենթախնդրում, դուք կստանաք այդ ենթախնդրի միավորների 50%-ը:

#	Միավոր	Սահմանափակումներ
1	3	$b_i = \text{true}$ .
2	4	$b_i = \text{false}$ .
3	16	$1 \leq n \leq 7$ .
4	25	$1 \leq n \leq 15$ .
5	22	$n = 2^{16} - 1$ և յուրաքանչյուր $b_i$ ընտրվում է $\{\text{true}, \text{false}\}$ -ից պատահականորեն:
6	30	Լրացուցիչ սահմանափակումներ չկան:

## Օրինակներ

Input file	Output file
4 3 111 7 1111111 3 000 7 000000000	1 2 3 1 2 3 4 5 6 7 3 2 1 7 6 5 4 3 2 1
2 3 010 7 0010110	3 2 1 7 3 1 5 2 4 6

## Բացատրություն

**Օրինակ 1.** Առաջին ենթախնդրի առաջին երկու թեստերում ունենք  $S(p) = 0$ :

Երրորդ թեստում, ունենք  $S(p) = 1$ , որովհետև `binary_search(n, p, 2)` վերադարձնում է `true`, բայց  $b_2 = \text{false}$ .

Չորրորդ թեստում, ունենք  $S(p) = 1$ , որովհետև `binary_search(n, p, 4)` վերադարձնում է `true`, բայց  $b_4 = \text{false}$ .

**Օրինակ 2.** Երկու թեստերի համար էլ  $S(p) = 0$ :