

# CSES 1750

*Planets Queries I*

**Prof. Edson Alves**

**Faculdade UnB Gama**

**You are playing a game consisting of  $n$  planets. Each planet has a teleporter to another planet (or the planet itself).**

**Your task is to process  $q$  queries of the form: when you begin on planet  $x$  and travel through  $k$  teleporters, which planet will you reach?**

Você está jogando um jogo composto de  $n$  planetas. Cada planeta tem um teletransporte para outro planeta (ou para si mesmo).

Sua tarefa é processar  $q$  consultas da seguinte forma: partindo do planeta  $x$  e viajando por  $k$  teletransportes, você chegará em qual planeta?

## Input

*The first input line has two integers  $n$  and  $q$ : the number of planets and queries. The planets are numbered  $1, 2, \dots, n$ .*

*The second line has  $n$  integers  $t_1, t_2, \dots, t_n$ : for each planet, the destination of the teleporter. It is possible that  $t_i = i$ .*

*Finally, there are  $q$  lines describing the queries. Each line has two integers  $x$  and  $k$ : you start on planet  $x$  and travel through  $k$  teleporters.*

## Output

*Print the answer to each query.*

## Entrada

A primeira linha da entrada tem dois inteiros  $n$  e  $q$ : o número de planetas e consultas. Os planetas são numerados  $1, 2, \dots, n$ .

A segunda linha contém  $n$  inteiros  $t_1, t_2, \dots, t_n$ : para cada planeta, o destino do teletransporte. É possível que  $t_i = i$ .

Finalmente, há  $q$  linhas descrevendo as consultas. Cada linha tem dois inteiros  $x$  e  $k$ : você inicia no planeta  $x$  e viaja através de  $k$  teletransportes.

## Saída

Imprima a resposta para cada consulta.

## **Exemplo de entrada e saída**

## Exemplo de entrada e saída

4 3

## Exemplo de entrada e saída

4 3  
↑  
*# de planetas*



## Exemplo de entrada e saída

4 3

1

2

3

4

## Exemplo de entrada e saída

4 3  
↑  
*# de consultas*

1

2

3

4

## Exemplo de entrada e saída

4 3

2 1 1 4

1

2

3

4

## Exemplo de entrada e saída

4 3  
2 1 1 4  
↑  
 $t_1$

1

2

3

4

## Exemplo de entrada e saída

4 3  
2 1 1 4  
↑  
 $t_1$



## Exemplo de entrada e saída

4 3  
2 1 1 4  
↑  
 $t_2$



## Exemplo de entrada e saída

4 3  
2 1 1 4  
↑  
 $t_2$



## Exemplo de entrada e saída

4 3  
2 1 1 4  
          ↑  
         $t_3$





## Exemplo de entrada e saída

4 3  
2 1 1 4  
          ↑  
         $t_3$



## Exemplo de entrada e saída

4 3  
2 1 1 4  
          ↑  
           $t_4$



## Exemplo de entrada e saída

4 3  
2 1 1 4  
          ↑  
           $t_4$



## Exemplo de entrada e saída

4 3

2 1 1 4

1 2



## Exemplo de entrada e saída

4 3  
2 1 1 4  
1 2  
↑  
 $x$



## Exemplo de entrada e saída

4 3  
2 1 1 4  
1 2  
↑  
 $x$



## Exemplo de entrada e saída

4 3  
2 1 1 4  
1 2  
↑  
 $k$



## Exemplo de entrada e saída

4 3

2 1 1 4

1 2





## Exemplo de entrada e saída

4 3

2 1 1 4

1 2



## Exemplo de entrada e saída

4 3  
2 1 1 4  
1 2 → 1



## Exemplo de entrada e saída

4 3

2 1 1 4

1 2

3 4



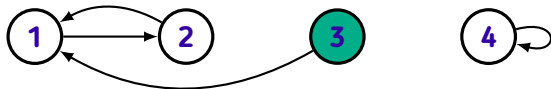
## Exemplo de entrada e saída

4 3

2 1 1 4

1 2

3 4



## Exemplo de entrada e saída

4 3

2 1 1 4

1 2

3 4



## Exemplo de entrada e saída

4 3

2 1 1 4

1 2

3 4



## Exemplo de entrada e saída

4 3

2 1 1 4

1 2

3 4



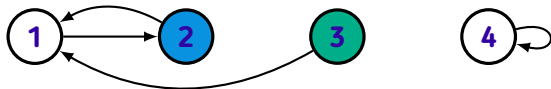
## Exemplo de entrada e saída

4 3

2 1 1 4

1 2

3 4





## Exemplo de entrada e saída

4 3  
2 1 1 4  
1 2  
3 4 → 2



## Exemplo de entrada e saída

4 3

2 1 1 4

1 2

3 4

4 1



## Exemplo de entrada e saída

4 3

2 1 1 4

1 2

3 4

4 1



## Exemplo de entrada e saída

4 3

2 1 1 4

1 2

3 4

4 1



## Exemplo de entrada e saída

4 3  
2 1 1 4  
1 2  
3 4  
4 1 → 4



## **Solução**

## Solução

- ★ Computar cada consulta em  $O(k)$  leva a um veredito TLE

## Solução

- ★ Computar cada consulta em  $O(k)$  leva a um veredito TLE
- ★ Contudo, é possível responder as consultas em  $O(\log k)$



## Solução

- ★ Computar cada consulta em  $O(k)$  leva a um veredito TLE
- ★ Contudo, é possível responder as consultas em  $O(\log k)$
- ★ Para isso, é preciso pré-computar  $\text{succ}(u, k)$  para as potências  $2^i$  tais que  $2^i \leq k$ , com  $i \geq 0$ , em  $O(n \log k)$

## Solução

- ★ Computar cada consulta em  $O(k)$  leva a um veredito TLE
- ★ Contudo, é possível responder as consultas em  $O(\log k)$
- ★ Para isso, é preciso pré-computar  $\text{succ}(u, k)$  para as potências  $2^i$  tais que  $2^i \leq k$ , com  $i \geq 0$ , em  $O(n \log k)$
- ★ Esta solução tem complexidade  $O((n + k) \log k)$

```
vector<int> solve(int N, const vector<int>& ts, const vector<ii>& qs)
{
    precomp(N, iMax, ts);

    vector<int> ans;

    for (auto [x, k] : qs)
        ans.emplace_back(succ(x, k));

    return ans;
}
```

```
void precomp(int N, int M, const vector<int>& s)
{
    for (int u = 1; u <= N; ++u)
        S[u][0] = s[u];

    for (int i = 1; i <= M; ++i)
        for (int u = 1; u <= N; ++u)
            S[u][i] = S[S[u][i - 1]][i - 1];
}

int succ(int u, int k)
{
    for (int i = 0; (1 << i) <= k; ++i)
        if (k & (1 << i))
            u = S[u][i];

    return u;
}
```