31ST INTERNATIONAL
OLYMPIAD IN INFORMATICS
4-11 AUGUST 2019
BAKU · AZERBAIJAN

cycle
Practice Tasks
English (ISC)

# Lost in the cycle

While visiting the Palace of Shirvanshahs, you get lost and find yourself in a fancy cyclic trap. The trap consists of $n$ identical rooms. Let us number all rooms from $0$ to $n - 1$. Each room has exactly one open door that leads to the next room, i.e. the door in the room $i$ leads to the room $i + 1$, for $i = 0, 1, \ldots, n - 2$, where the door in the room $n - 1$ leads to the room $0$. After you use the door, it closes behind you, so you cannot go back. In the beginning, you are in some room $p$, but you do not know that, as the rooms are identical. You need to reach the room $0$ and wait there for help in order to escape the trap. Fortunately, you have received a device indicating the distance to the room $0$. It will tell you if you can reach the room $0$ by walking through no more than $\frac{n}{2}$ doors. Before the device runs out of battery, you can use it at most 35 times.

## Implementation details

You should implement the following procedure. It will be called by the grader once for each test case.

```
void escape(int n)
```

- $n$: number of rooms.
- You should be in the room $0$ after this procedure will be executed.

The above procedure can make calls to the following procedure:

```
bool jump(int x)
```

- $x$: number of doors you want to go through, before using the device. The value of $x$ must be between $0$ and $n - 1$, inclusive.
- This procedure returns `true` if after walking through $x$ doors you can reach the room $0$ by walking through no more than $\frac{n}{2}$ doors, and `false` otherwise. Please note that your position changes before you use the device.
- You can use this procedure at most 35 times.

## Constraints

- $2 \le n \le 10^9$,
- $1 \le p < n$,

- The grader **is not** adaptive. That means that the grader chooses $p$ before the method `escape` is executed, and does not change it during this execution.

## Subtasks

1. (11 points) $n \leq 30$,
2. (33 points) $n = 2^k - 1$ for some $k$,
3. (56 points) No additional constraints.

## Example

The grader makes the following procedure call:

```
escape(10)
```

There are $n = 10$ rooms. Suppose that you start in the room $p = 5$. Let us consider the following calls to the procedure `jump`:

- `jump(3)`: You are now in room 8. You have to use door 2 times, to reach room 0, which is less than $\frac{n}{2} = 5$, therefore this procedure returns `true`.
- `jump(2)`: You are now in room 0 and you can stay here, but you do not have to. The procedure will return `true`, as you do not need to use any doors to get to the room 0.
- `jump(3)`: You are now in room 3. You have to use door 7 times to reach 0, therefore the procedure will return `false`.
- `jump(1)`: You are now in room 4. You have to use door 6 times to reach 0, therefore the procedure will return `false` again.
- `jump(6)`: You are now in room 0. The procedure will return `true`, as explained above.

After these calls, you will finish in room 0 and this test will be accepted.
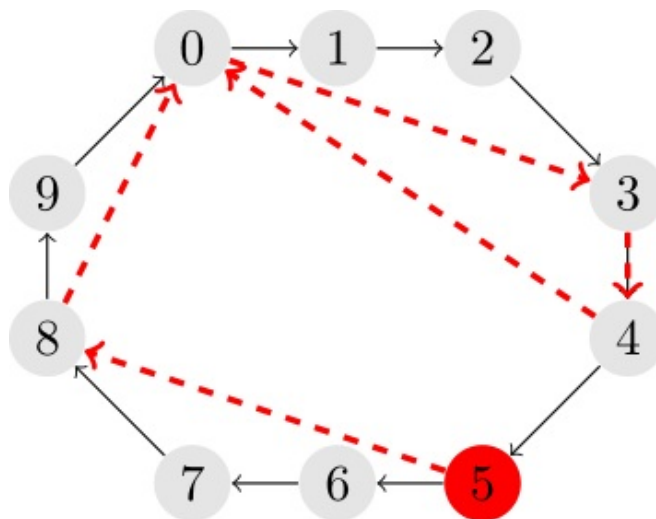
# Sample grader

The sample grader reads the input in the following format:

- line 1: $n$ $p$

If your solution finished successfully and you are in the room $0$ at the end, the sample grader prints a single line containing the word OK to the standard output, and then a single line to the standard error stream, containing the number of calls to the procedure jump. Otherwise, the sample grader prints to the standard output the reason why your solution fails.