A. Equal Subsequences

time limit per test: 1 second
memory limit per test: 256 megabytes

We call a bitstring* perfect if it has the same number of 101 and 010 subsequences†. Construct a perfect bitstring of length n where the number of 1 characters it contains is exactly k.

It can be proven that the construction is always possible. If there are multiple solutions, output any of them.

*A bitstring is a string consisting only of the characters 0 and 1.

†A sequence a is a subsequence of a string b if a can be obtained from b by the deletion of several (possibly zero or all) characters.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 500$). The description of the test cases follows.

The first line of each test case contains two integers n and k ($1 \le n \le 100, 0 \le k \le n$) — the size of the bitstring and the number of 1 characters in the bitstring.

Output

For each test case, output the constructed bitstring. If there are multiple solutions, output any of them.

Example

input

```
5
4 2
5 3
5 5
6 2
1 1
```

output

```
1010
10110
11111
100010
1
```

Note

In the first test case, the number of 101 and 010 subsequences is the same, both being 1, and the sequence contains exactly two 1 characters.

In the second test case, the number of 101 and 010 subsequences is the same, both being 2, and the sequence contains exactly three 1 characters.

In the third test case, the number of 101 and 010 subsequences is the same, both being 0, and the sequence contains exactly five 1 characters.

Supported by