Problem D. Diameter Two

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

You are building a computer network for a new company. The network consists of n nodes numbered from 1 to n. The nodes can be connected via bidirectional wires. Each wire connects exactly two nodes. Each pair of nodes can be connected with at most one wire. If a wire connects two nodes, we'll say that these two nodes are *directly connected*.

The first k nodes (with indices $1, 2, \ldots, k$) will be *untrusted* and must be connected to the network *securely*. Each of these nodes must be directly connected to *exactly one* other node.

The remaining $n - k$ nodes (with indices $k+1, k+2, \ldots, n$) will be *trusted* and must be connected to the network *reliably*. Each of these nodes must be directly connected to *at least two* other nodes.

The *diameter* of the network must not exceed 2: for any two nodes i and j, they must either be directly connected, or there must exist a node k such that nodes i and k are directly connected, and nodes k and j are directly connected.

To minimize the costs, the number of used wires must be as small as possible.

Build a network satisfying all the conditions above, or report if this is impossible.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 50$). Description of the test cases follows.

The only line of each test case contains two integers n and k, denoting the total number of nodes and the number of untrusted nodes, respectively ($3 \le n \le 50$; $0 \le k \le n$).

Output

For each test case, if it is impossible to build a network satisfying the given conditions, print a single integer -1.

Otherwise, in the first line, print the number of used wires m. In each of the following m lines, print two integers u_i and v_i, denoting the indices of the nodes connected with the i-th wire ($1 \le u_i, v_i \le n$; $u_i \ne v_i$).

Example

standard input	*standard output*
3 3 0 5 2 6 6	3 1 2 1 3 2 3 5 1 3 2 3 3 4 3 5 4 5 -1