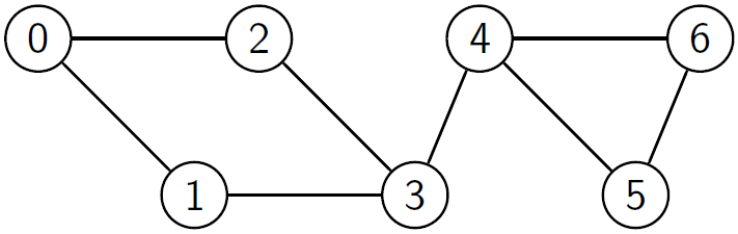
Problem C
Weak Links

Source file name: `wlinks.c`, `wlinks.cpp` *or* `wlinks.java`

The Thought Police (TP) is trying to dismantle a network of thought-criminals (TC). As it may be easily inferred from the name, TCs commit thought-crimes and they do so by sending messages through a communication network that TCs believe to be secure, but which has already been infiltrated by the TP. In order to preserve the security of the communication network, each TC only communicates with a reduced set of contacts, so a message may need to go through several intermediate TCs to reach its destination. The TCs in the communication network are connected in such a way that a message from any source may reach any destination. Even though the TP knows all the TCs of the network, they want to interrupt its activity in the most subtle way. The plan is to identify the weak links of the network. A *weak link* is any single link that if removed, would make it impossible for at least one TC to communicate with some other TC in the network. For instance, the next figure shows a network with 7 TCs.



The connection between TCs 3 and 4 is a weak link since, if removed, it would make it impossible for TCs $0, 1, 2, 3$ to communicate with TCs $4, 5, 6$. All the other connections of the network are not weak links. The government has hired you to support its crusade to defend the country from thought-criminals and to preserve the security of its citizens, by helping the TP to carry on its plan identifying the weak links of the TCs communication network.

Input

The input contains several test cases. The first line of each test case contains two blank-separated integers n and m, where n is the number of TCs ($2 \leq n \leq 1000$), and m is the number of direct communications links between them ($1 \leq m \leq 10000$). Then m lines follow, each one containing two blank-separated integers N_i and N_j ($0 \leq N_i < N_j < n$), indicating that there is a direct communication link between TCs N_i and N_j in the communication network. Note that all communication links are bidirectional. A line with two zeros "`0 0`" indicates the end of the input.

The input must be read from standard input.

Output

For each test case, a line with the list of the weak links has to be printed.

Each line starts with a number p indicating the number of weak links in the communication network, then p links of the form N_{i_k} N_{j_k} must follow ($0 \leq N_{i_k} < N_{j_k} < n$). Weak links must be printed in ascending order of their N_{i_k}. If two weak links have the same N_{i_k}, the link with the minimum N_{j_k} should be printed before. All numbers printed in each line should be separated by a single blank. There are no blanks after the last number of any line.

The output must be written to standard output.

Sample input	**Output for the sample input**
4 3 0 3 0 1 0 2 7 8 0 1 0 2 1 3 2 3 3 4 4 5 4 6 5 6 0 0	3 0 1 0 2 0 3 1 3 4