



Critical Network Lines

Input file: net.in

Output file: net.out

Source code: net.pas/.c/.cpp

100 points

Time limit: 3 sec

Memory limit: 64 MB

Consider a communication network that consists of a set of nodes and a set of two-way direct communication lines between pairs of nodes. It is known that the investigated network is connected, that is, there is a communication path between every pair of nodes. Some nodes provide service type *A* to all nodes (including itself), while some nodes provide service type *B* to all nodes (including itself). The same node may provide both types of services. Every node must have access to both types of services.

If a direct line falls out, it might happen that a service becomes unavailable to some node; a direct line with this property is called a critical network line.

Task

You are to write a program that determines the number of critical network lines (Subtask A) and the pairs of nodes they connect (Subtask B).

Input

The first line of the text file net.in contains four integers, *N*, *M*, *K*, and *L*. *N* ($1 \leq N \leq 100\,000$) is the number of nodes of the network, *M* ($1 \leq M \leq 1\,000\,000$) is the number of direct communication lines, *K* ($1 \leq K \leq N$) is the number of the nodes that provide service *A*, and *L* ($1 \leq L \leq N$) is the number of the nodes that provide service *B*. The nodes are identified by the numbers from 1 to *N*. The second line contains *K* integers, the identifiers of the nodes that provide service *A*. The third line contains *L* integers, the identifiers of the nodes that provide service *B*. Each of the following *M* lines contains a pair of integers, *p q* ($1 \leq p, q \leq N, p \neq q$); the pair defines a direct communication line. There is at most one direct communication line between any two nodes.

Output

The first line of the text file net.out contains a single integer, *S*, the number of the critical lines of the network (Subtask A). Each of the following *S* lines contains a pair of integers, *p q* ($1 \leq p, q \leq N$), defining a critical network line (Subtask B). You can output the critical network lines in any order, and for each line you can output the endpoints in any order.

Example

| net.in | net.out |
|----------|---------|
| 9 10 3 4 | 3 |
| 2 4 5 | 3 2 |
| 4 9 8 3 | 5 6 |
| 1 2 | 7 9 |
| 4 1 | |
| 2 3 | |
| 4 2 | |
| 1 5 | |
| 5 6 | |
| 6 7 | |
| 6 8 | |
| 7 9 | |
| 8 7 | |