Lazar is the CEO of a company that constructs electrical touristic taxis which travel around the cities of Musaland. The road network is comprised of bidirectional highways, connecting two cities.

A Taxi begins its journey from a certain city and visits all others in a few days. At the end of every day, the taxi parks to recharge, and in the morning, it sets off again from the same city it stayed overnight. What is more, the taxi cannot revisit cities during the day. Your goal is to find an optimal itinerary.

Every city is characterized by *prestige* - positive integer . The price for passing through a highway between two cities is calculated as the squared difference of the prestige values of the two cities, or: for two cities i and j, connected by a direct road. Also, every recharging contributes to the expenses.

Interestingly, the *prestige* is subjective, and you have the opportunity to assign the *prestige* values in advance. From a given set of *prestiges*, you have to distribute them among the cities. After that you cannot modify them.

Being given the road network and a set of *prestiges*, your task is to distribute the *prestiges* and find a route with the lowest possible total price. The initial city is of your choice.

**Input**

The first line of the file **taxi.in** contains two integers n and m – the number of cities and the number of highways. The next line contains n integers - , the *prestige* values.

Each of the next m lines consists of 2 numbers - , describing the two ends of a highway.

**Output**

On the first line of the file **taxi.out** print n numbers - where denotes the *prestige* of city i, for every i from 1 to n.

On the second line print k – the number of days the taxi will travel. The next k lines must comprise of the routes you have chosen – print the count of cities and the cities . Every city has to be part of at least one route and every day’s initial city, except the first day’s city, has to be equal to the previous day’s last city.

**Scoring**

If 1) the taxi doesn’t visit each city at least once, 2) the taxi revisits a city in a day 3) every day’s first city is not the same as yesterday’s last city, 4) the total number of cities across all routes is more than or 5) *yourScore* exceeds , you will receive “Error” message and 0 points for the test. Otherwise:

For each test, let *minScore* be the smallest score among all participants' scores and *yourScore* be your score. You will be awarded multiplied by the amount of points for the test.

**Constraints**

**Time limit: 5 sec.**

**Memory limit: 256 MB**

The tests are distributed as follows:

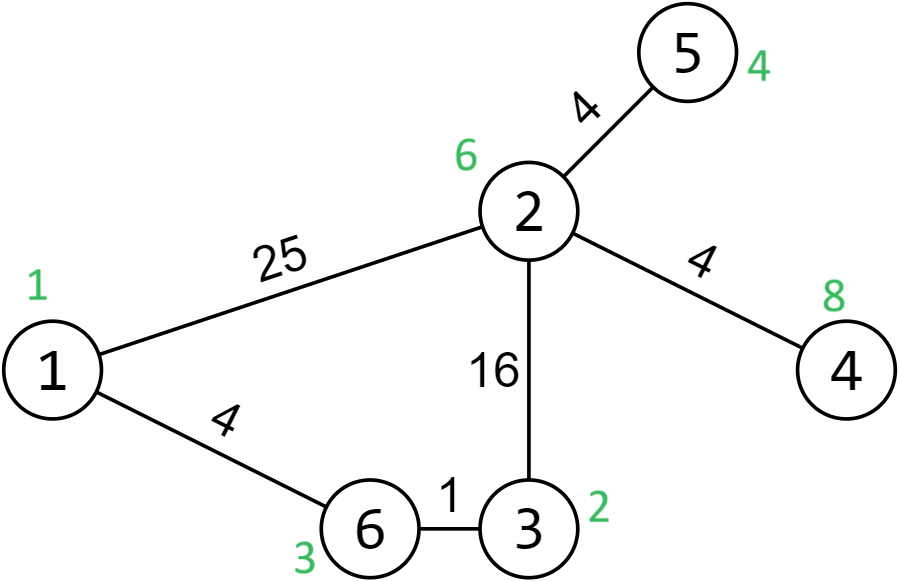
|  |  |
| --- | --- |
| **Percentage of tests** | n |
| 20% |  |
| 20% |  |
| 20% |  |
| 20% |  |
| 20% |  |

**Sample test**

|  |  |
| --- | --- |
| **Input (taxi.in)** | **Output (taxi.out)** |
| 6 6  3 1 4 6 2 8  1 2  3 6  2 4  2 3  5 2  6 1 | 1 6 2 8 4 3  4  3 1 2 5  4 5 2 3 6  2 6 3  3 3 2 4 |

**Example explanation**

The *prestige* values are coloured in green while the numbers on the segments represent the prices for passing through the highways. The taxi travels for 4 days. The first city is 1 and all cities are visited. The prices are:

  
Day 1: 25 + 4 = 29 (1-2-5) Day 2: 4 + 16 + 1 = 21 (5-2-3- 6) Day 3: 1 (6-3) Day 4: 16 + 4 = 20 (3-2-4) Overall: 71 The total price is 71\*4 = 284.