# Coding Assignment 3

ECS 122A Algorithm Design and Analysis

## Building a Water Supply System

### Problem Description

In ByteLand, there are $n$ villages and $m$ roads. Each road connects a pair of villages. There are wells in some villages, which provide water supply to their neighborhood.

Recently, ByteLand was hit by a severe earthquake that destroys all the roads and wells. As the first step of reconstruction, the government plans to rebuild some roads and wells such that it is possible to travel from each village to at least one village with a (rebuilt) well. This is to ensure that every village has access to water.

The cost of rebuilding each road and well may vary. Can you write a program that, given the cost of rebuilding each road and well, computes a minimum-cost reconstruction plan that meets the government's requirement?

### Input and Output

Each input file contains several test instances. The first line contains a single integer $C$, representing the number of test instances. The description of the $C$ instances follows.

Each test instance is formatted as follows. The first line contains two integers $n$ and $m$, representing the number of villages and the number of roads in Byteland, respectively. The villages are numbered $1, 2, \ldots, n$. The second line contains $n$ space-separated integers $w_1, w_2, \ldots, w_n$. If the $i^{th}$ integer is positive, then there was a well at village $i$ before the earthquake, and it costs $w_i$ dollars to rebuild this well; otherwise, $w_i$ must be $-1$, indicating that there is no well in the $i^{th}$ village.

Each of the subsequent $m$ lines contains three space-separated integers $u$, $v$, and $c$ ($1 \leq u, v \leq n$), indicating that there is a road between village $u$ and village $v$, and the cost of rebuilding this road is $c$ dollars.

For each test instance, your program should output a single integer: the minimum cost for a reconstruction plan that meets the government's requirements.

In this assignment, you are provided with sample submissions that contain code for input and output. You only need to complete a function that, given the roads and the costs of rebuilding the roads and wells, returns the minimum cost of a desired reconstruction plan. Your function should ideally be able to compute the answer in $O(m \log n)$ time.

### Constraints

Each input file contains at most 2000 test instances. In a single input file, the total number of roads across all test instances is at most $4 \cdot 10^5$. The time limit (for all test cases in a single file) is 1 second for C/C++ and 3 seconds for Python.

Each test instance satisfies the following additional constraints:

- $1 \leq n \leq 10^5$.
- $m \leq 4 \cdot 10^5$.
- The cost of rebuilding a road or a well is a positive integer not greater than $10^9$.
- For every village $u$, there exists at least one village $v$ such that (i) $u$ is reachable from $v$ by roads and (ii) $v$ contains a water well.

**Remark 1:** Under the above constraints, it is always possible to rebuild ByteLand. For example, one can always rebuild all roads and wells so that all villages can access at least one well. However, this reconstruction plan may not be optimal.

**Remark 2:** Note that the answer may be as large as $n \cdot 10^9$, which may not fit into a 32-bit integer type. We advise using a 64-bit integer type (for C/C++).

## Test Cases

Your program will be evaluated on 10 test files. Passing all the instances within a file will earn you 0.8 points. Note that if your program outputs an incorrect answer for even one of the instances within a file, you will receive 0 points for that file.

**Sample Input 1:**

```
3
8 10
20 10 -1 -1 -1 -1 15 90
2 1 25
1 6 25
7 8 30
8 3 15
5 3 40
1 4 90
1 7 30
6 2 25
6 7 5
3 1 5
6 5
1 -1 -1 -1 -1 -1
1 2 1000000000
3 1 1000000000
1 4 1000000000
4 5 1000000000
6 5 1000000000
1 0
1
```

**Sample Output 1:**

```
200
5000000001
1
```

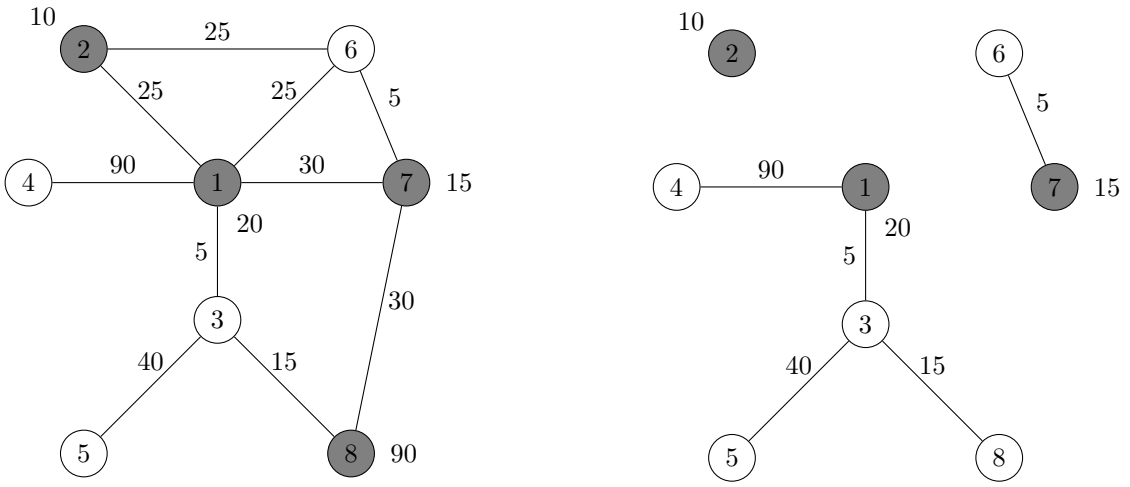
## Sample Explanation

Sample Input 1 contains $C = 3$ test instances. The first test instance is given in Fig. 1. Fig. 1a shows the villages, roads, and wells before the earthquake. Each vertex represents a vertex, each edge represents a road, and the villages with wells are in gray. The label on each edge represents the cost of

reconstructing the corresponding road; the label on each gray vertex represents the cost of rebuilding the well in the village.

Fig. 1b gives the optimal reconstruction plan, represented by the roads and wells that should be reconstructed. The total cost of this plan is 200. Note that if ByteLand is reconstructed according to this plan, every village will be able to access at least one well: village 2 will have its own well; villages 6 and 7 can access the well in village 7; villages 1, 3, 4, 5, and 8 can travel to the well in village 1.

For the second and third test instances, the optimal reconstruction plan is to rebuild all roads and wells.




(a) The villages, roads, and wells before the earthquake.

(b) The roads and wells that should be reconstructed.

Figure 1: The first test instance in the sample input.

## Submission Guideline

Write your program in either C, C++ or Python **in a single file**. Submit the file on Gradescope. The time limit on Gradescope is 1 second for C/C++ and 3 seconds for Python. You can make at most 10 submission attempts. You may refer to `sample.cpp` or `sample.py` for sample code that takes input and writes output.