CSCI 406: AlgoBOWL

Reminder: You are NOT allowed to consult the internet to solve this problem.

1 Problem Description

You are given

1. A universal set U of n elements $\{1, 2, 3, \ldots n\}$.
2. A list $S_1, S_2, \ldots S_m$ of subsets of U such that the union of the m subsets gives U.
3. The weight w_i of each subset S_i.

A *cover* is defined as a collection of subsets whose union is equal to U. The goal is to find a cover whose total weight is minimized.

Input Format: Line 1 of the input contains n, the number of elements in U. Line 2 contains m, the number of subsets of U. Each of the next m pairs of lines specifies a subset S_i and its weight w_i. In the example that follows, I've included comments for your understanding. Comments should **not** be included in actual inputs:

```
5                   // U = {1, 2, 3, 4, 5}
3                   // Three subsets: S1, S2, S3.
1 2 3               // S1 = {1, 2, 3}
15                  // w1
4 5                 // S2 = {4, 5}
6                   // w2
1 2 3 4 5           // S3 = {1, 2, 3, 4, 5}
26                  // w3
```

We have: $S_1 = \{1, 2, 3\}$ with weight $w_1 = 15$, $S_2 = \{4, 5\}$ with weight $w_2 = 6$, $S_3 = \{1, 2, 3, 4, 5\}$ with weight $w_3 = 26$. (Notice that $S_3 = U$, but this is okay because a set can be a subset of itself.)

Two possible covers are $C_1 = \{S_1, S_2\}$ and $C_2 = \{S_3\}$. Their weights are $15 + 6 = 21$ and 26, respectively, making C_1 the better solution.

Input Restrictions: Assume $n \leq 1000$ and $m \leq 500$. Each weight must be an integer in [1,1000].

Output Format: Line 1 of your output will contain the best weight that your algorithm was able to find. The next line will contain the IDs of the sets included in your cover. To illustrate, the output for cover C_1 in the example is

```
21
1 2
```

Note: This problem is NP-hard, which means that it is unrealistic to expect that your algorithm will compute an optimal solution in a reasonable time frame. Please keep this in mind as you work on this project.