Problem 1: Containers

This assignment should be done individually

- To do so, create a mooshak account with your ualg login (without @ ualg.pt) in mooshak. Example: the student with the number 12345 uses a12345. NB: do not use your name in the login.
- Submit your code to mooshak http://deei-mooshak.ualg.pt/~jvo/ Problem B up to:

  October 14, 2024 – 17h
- A submission will remain *pending* until validated by the Instructor during the lab class. Only *final* submissions will be considered for evaluation. Deadline for validation:

  November 04, 2024

Problem

Containers are used in the transport of all kinds of goods. Once in land, it is often required to move them from place to place. Some containers are heavier than others which have implications in the energy required to move them.

The goal is to move a set of containers to a desired configuration starting from an initial configuration with a minimum energy cost and following a few simple rules:

i) Only one container can be moved at a time: it may either be placed on the ground or placed on the top of another container.
ii) The only containers that can be moved are those that do not have any other container(s) on their top. See Fig. 1 for an illustration using only three containers.

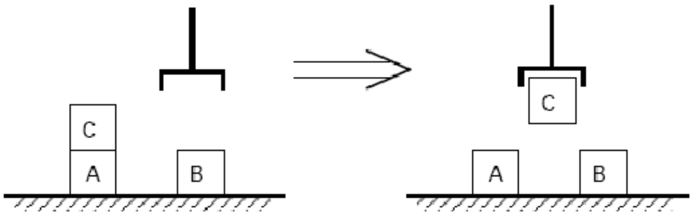

Fig. 1 – An example of possible movement: As the container C on the left has no other container on top, it can be moved to either the ground or to the top of container B (whose top is also clear).

Task
There are many approaches to address this problem. However, your task is to write a program that, based on the approach followed in Tutorial 1, returns the successive configurations required for achieving the final configuration starting from the initial one, along the minimum cost path, i.e., along the path requiring the lower possible energy cost.

For comparing configurations, it does not matter the ordering of the stacks on the ground. For example, a configuration with the containers A, B and C on the ground is viewed as equal to B, A, and C on the ground. The ground can hold as many containers as needed.

Please note that any other approach, however meritorious it may be, will be quoted with 0 (zero).

Input
The input has two lines. The first line represents the initial configuration while the second line denotes the goal configuration. In both lines, the containers are identified by a single letter, e.g., A or a; a stack is composed by a set of containers, starting from the ground up; different stacks are separated by spaces. In the first line a single digit positive integer representing the energy cost of moving the container immediately follows its identifier. Example 1 bellow corresponds to the example in Fig.1 where all costs are equal to 1.

All given instances will have a unique solution.

Output
The sequence of configurations from the initial to the final one (both *inclusivé*), with a blanc line separating each one of them. At the end of this sequence, and after a new empty line, a non negative integer with the minimum cost of the path found should be also presented.

A configuration represents each stack of containers in a separated row. Each stack is represented within [] and two consecutive containers within the same stack are separated by a comma followed by a space. The stacks are shown sorted by lexicographical order of their container in contact with the ground.

Sample Input 1

```
A1C1 B1
A C B
```

Sample output 1

```
[A, C]
[B]

[A]
[B]
[C]

1
```

Sample input 2

```
A1B2C3
A B C
```

Sample output 2

```
[A, B, C]

[A, B]
[C]

[A]
[B]
[C]

5
```