Choosing teams

In a certain Computer Science course, students are assigned to groups according to preferences that they specify. Each student is sent an electronic survey and asked to give answers to three questions:

1. What is your IU email username?
2. Please choose one of the options below and follow the instructions.
   (a) You would like to work alone. In this case, just enter your userid in the box and nothing else.
   (b) You would like to work in a group of 2 or 3 people and already have teammates in mind. In this case, enter all of your userids (including your own!) in the box below, in a format like userid1-userid2 for a team of 2, or userid1-userid2-userid3 for a team of 3.
   (c) You would like to work in a group of 2 or 3 people but do not have any particular teammates in mind. In this case, please enter your user ID followed by one "zzz" per missing teammate (e.g. djcran-zzz where djcran is your user ID to request a team of 2, or djcran-zzz-zzz for a team of 3).
   (d) You would like to work in a group of 3 people and have some teammates in mind but not all. Enter all of your ids, with zzz's to mark missing teammates (e.g. if I only have one teammate (vkvats) in mind so far, I'd enter djcran-vkvats-zzz).
3. If there are any people you DO NOT want to work with, please enter their userids here (separated by commas, e.g. userid1,userid2,userid3).

Unfortunately — and as we already discovered while assigning teams for Assignment 1 — the student preferences may not be compatible with each other: student A may request working with student B, but B may request not to work with A, for example. Students are going to complain, so the course staff decides to minimize their own work. They estimate that:

- It will take 5 minutes to grade each assignment, so total grading time is 5 times the number of teams.
- Each student who requested a specific group size and was assigned to a different group size will send a complaint email to an instructor, and it will take the instructor 2 minutes to read this email.
- If a student is not assigned to someone they requested, there is a 5% probability that the two students will still share code, and if this happens it will take 60 minutes for the instructor to walk through the Academic Integrity Policy with them. If a student requested to work with multiple people, then this will happen independently for each person they were not assigned to. If both students requested each other, there will be two meetings.
- Each student who is assigned to someone they requested *not* to work with (in question 3 above) complains to the Dean, who meets with the instructor for 10 minutes. If a student is assigned to a group with multiple people they did not want to work with, a separate meeting will be needed for each.

The total time spent by the course staff is equal to the sum of these four components. You can assume that each student fills out the survey exactly once, and fills it out according to the instructions. Your goal is to write a program to find an assignment of students to teams that minimizes the total amount of work the staff needs to do, subject to the constraint that no team may have more than 3 students. Your program should take as input a text file that contains each student's response to these questions on a single line, separated by spaces. For example, a sample file might look like:

```
djcran djcran-vkvats-nthakurd sahmaini
sahmaini sahmaini _
sulagaop sulagaop-xxx-xxx _
fanjun fanjun-xxx nthakurd
nthakurd nthakurd djcran,fanjun
vkvats vkvats-sahmaini _
```

where the underscore character (_) indicates an empty value.

We have provided skeleton code to get you started, which can be run like:

```
python3 ./assign.py [input-file]
```

Your job is to complete the solver() function. The function should return the final groups (each named according to the students in the group, separated by hyphens), and the total cost (time spent by instructors in minutes). For example, one assignment for the above file could be:

```
["djcran-vkvats-nthakurd", "sahmaini", "sulagaop-fanjun"]
```

which has a cost of 34, computed by the sum of: 1. There are three groups' assignments to grade ($3 \times 5 = 15$ minutes) 2. Three people (sulagaop, nthakurd, and vkvats) didn't get the requested number of teammates ($3 \times 2 = 6$ minutes) 3. One person (nthakurd) had to work with someone they requested not to work with (djcran) (10 minutes) 4. One person (vkvats) didn't get to work with a person they requested (sahmaini) ($0.05 \times 60 = 3$ minutes)

Hint: It may not always be possible to find the actual best solution in a reasonable amount of time, and "reasonable amount of time" may differ from one problem to the next. Our grading program will eventually exit your program if it takes too long. Your program is thus allowed to generate multiple solutions, which may be useful if your approach can quickly produce an estimate of the solution, and then as it performs more computation, finds better and betters solutions. You'll call yield() each time you have found an answer — see skeleton code for details.