Graduation Ceremony X54305_en

The high school graduation party is coming. All students have to come in pairs. In order to simplify the always conflicting partner choice, we decided to automatize the process. Every student will write a secret list of acceptable partners into a program, that will assign a partner from the list to each student, if it is possible. We assume that there is an even number of students and we do not make any distinction among sexual orientation. Write a program in Python that, using a SAT solver, finds a solution to the problem.

Although this problem is a variation of the perfect matching problem, that can be solved in polynomial time, we have decided to use a SAT solver. In order to use the optilog library, the program has to include something like:

```
from optilog.solvers.sat import *
...
solver = Glucose41()
solver.add_clauses(...)
solver.solve()
solver.model()
```

Input

The input is a text (in the stdin) that contains a list of desired partners. Every list has as the first element the name of the student writing the wish, and as the rest of the elements, all the other students he/she would accept as possible partners. For instance:

```
Alex   Bernie Casey
Bernie Alex Casey
Casey  Alex Bernie Denny
Denny  Alex Bernie Casey
```

Output

The output is also a text (in the stdout) with a list of pairs representing a possible solution. In this example:

```
{ Alex, Bernie }
{ Casey, Denny }
```

If there is not a solution, the answer must be:

```
No solution
```

Sample input 1

```
Alex   Bernie Casey
Bernie Alex Casey
Casey  Alex Bernie Denny
Denny  Alex Bernie Casey
```

Sample output 1

```
{ Alex, Bernie }
{ Casey, Denny }
```

Sample input 2

```
Alex   Bernie Casey
Bernie Alex Casey
Casey  Alex Bernie Denny
Denny  Alex Bernie
```

Sample output 2

```
No solution
```

Samples have been selected to ensure that there exists (up to permutations) at most one solution.

Problem information

Author: Jordi Levy

Generation: 2026-01-25T16:53:25.276Z