ModRef 2022: Model and Solve Competition
Team Formation

1 Problem Statement

You have to form teams of students from different universities into teams to maximize the effectiveness of the teams.

You are given n people, each person has a seniority in the range 1..5 and a university encoded as integers $1..m$. Each pair of students has a mutual compatibility when working together given by an $n \times n$ symmetric array.

The goal is to produce teams of 2 to 3 members each so that

- There is different seniority levels in the team
- Not all members of the team are from the same university.
- Each person is assigned to exactly one team.

Given data defined by

```
{
  "n" : 8,
  "compatibility" : [[ 0,  5, -3, -7,  4,  2,  1, -4],
                     [ 5,  0, -2,  0,  5,  1, -4,  5],
                     [-3, -2,  0,  4, -6, -2,  3,  0],
                     [-7,  0,  4,  0,  3, -1, -3, -2],
                     [ 4,  5, -6,  3,  0, -5, -2,  2],
                     [ 2,  1, -2, -1, -5,  0,  6, -1],
                     [ 1, -4,  3, -3, -2,  6,  0,  1],
                     [-4,  5,  0, -2,  2, -1,  1,  0]],
  "m" : 3,
  "univ" : [1, 1, 2, 2, 3, 3, 3, 3],
  "seniority" : [2, 1, 1, 3, 2, 1, 1, 1]
}
```

A potential solution is teams

	{1,	6,	7}	{3,	5,	8}	{2,	4 }
seniority	2	1	1	1	2	1	1	4
uni	1	3	3	2	3	3	2	2

showing that the constraints are met. The total compatibility is

$$(2 + 1 + 6) + (-6 + 0 + 2) + (0) = 5.$$

The solution can be represented in JSON as

```
{
   people : [
      {"set" : [1,6,7]},
      {"set" : [3,5,8]},
      {"set" : [2,4]}
   ]
}
```

but note the lists don't have to be sorted, and you are allowed to have additional empty lists in the output, so the following would be seen as equivalent

```
{
   people : [
      {"set" : [4,2]},
      {"set" : []},
      {"set" : [6,1,7]},
      {"set" : [3,5,8]},
      {"set" : []}
   ]
}
```