

# University of Galway Programming and Tools for AI (CT5132/CT5148) 2022/23 Assignment 3

James McDermott

Suppose there is a Halloween party at the Students' Union, and everyone who attends wears a costume which fully disguises their real identity.

During the party, Dracula talks to a Pumpkin. The next day, Dracula decides it would be nice to meet the Pumpkin again. But Dracula doesn't know the Pumpkin's real name. How can Dracula find the Pumpkin?

Luckily, everyone at the disco is a member of one or more Student Societies. Before the disco, each Society had a meeting to choose their costumes in secret. So, everyone in a particular Society knows the costume of everyone else in the society, but they don't know the costume of anyone outside the society.

For example, suppose Dracula is a member of the Karate Society and the Debating Society. Dracula can ask everyone in those Societies whether they know the Pumpkin's real name. If not, they can ask anyone they know in other Societies, and so on, recursively.

Here are the memberships of each Society (this is a small example):

```
{
"Karate Society": ["Dracula", "Frankenstein", "Werewolf", "Pilot", "Luke", "Leia"],
"Debating Society": ["Dracula", "Frankenstein", "Leia", "Cockroach", "Iron Man", "Zombie"],
"Musical Society": ["Bunny", "Pumpkin", "Cinderella", "Goblin Queen", "Goblin King"],
"History Society": ["Bunny", "Pilot", "Luke"]
}
```

Dracula wonders: will I ever find the Pumpkin? Who exactly will pass my message on at each step? In this example, we can see the path: [Dracula, Luke, Bunny, Pumpkin]. There are other paths equally long, but none shorter.

Write a program to answer these questions. It should work in general, not just for the example given above.

It should show the list of people who will pass the message from Dracula to Pumpkin (in order), or else should output that the message will not reach Pumpkin.

The Societies will be given in an input file, e.g. one of the following, provided in Bb.

- societies1.txt
- societies2.txt

Your program should be named `dracula.py` and it will be called e.g. as follows:

```
$ python dracula.py societies1.txt
```

It should *read* the input specified on the command-line (i.e. don't copy-paste the input into your program) and provide the solution, and should work for either input.

Hint 1: you can use 'eval'. Hint 2: you can use NetworkX if you like, but you don't have to.

**Groups** Students may work solo or in groups of 2, as desired, but not work with any student who they have worked with in any other assignment in this module or any other. In groups of 2, both students should submit and submissions should be identical. In any communication concerning the assignment, students must cc all group members.

**Plagiarism** Students are reminded of the University's policy on plagiarism. Students may discuss the assignment with other students/groups but must not look at other students'/groups' work, including students from other years, or allow others to look at theirs. Any online sources used must be cited with URL and date of access in a comment. Materials from CT5132/CT5148 need not be cited. By making a submission in Bb, you declare that you have abided by these conditions. Students may be called to interview to discuss their submissions. Suspected infringements will be investigated, marks may be deducted, and cases may be referred to U Galway authorities.

**Deadline** midnight Friday 25th November

**Weighting** 10% of the module.

**Submission** Submit a single `dracula.py` file only, containing the student name(s) and ID(s) in a comment at the top.