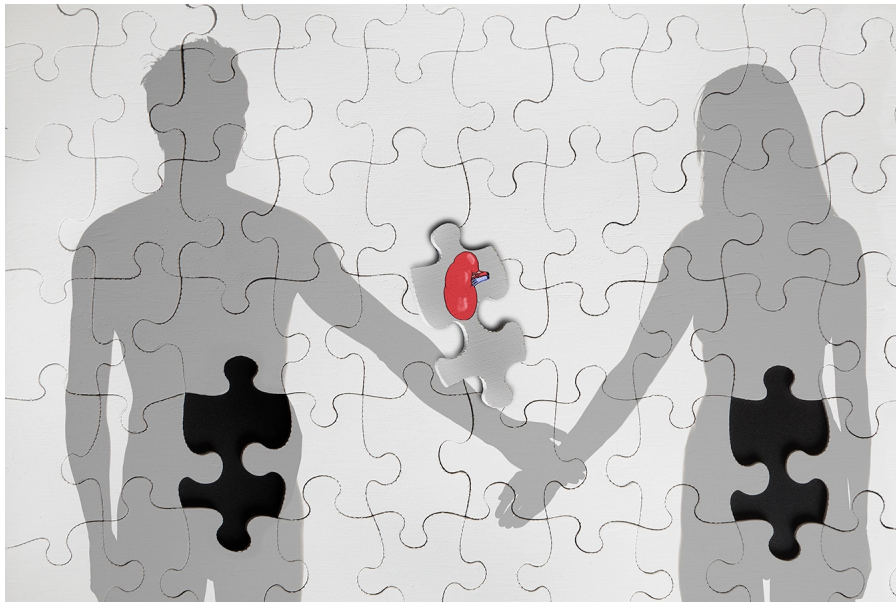


Programming Assignment 1: Living Kidney Donation

This programming assignment is to be completed in either Java or Python (your choice) and submitted on Gradescope.

Collaboration is restricted to just 2 people (other than yourself), and it must be "Whiteboard Only". In summary, this means that you may discuss the assignment with 2 others, but you cannot preserve any artifacts (digital or physical) from these discussions. In particular, this means you may not take and keep photos, notes, video, audio recordings, whiteboard contents, etc. from any discussions.

All collaborators and other resources consulted should be listed as a comment at the top of your code submission. Course staff and official course materials do not need to be listed.



Motivation

Kidneys are organs in the human body whose function is critical for maintaining life. Put simply, the kidneys primarily serve to regulate the salinity, water content, and pH of one's blood, and also remove toxins of various kinds from the body. If someone has no working kidneys then they must either receive frequent kidney dialysis (which cycles blood through a machine to perform the function of a kidney), or else receive a new kidney from an organ donor.

Humans are generally born with two kidneys, and one functioning kidney is sufficient for survival, so it is possible (though not risk-free) for a living person to donate a kidney to a patient in need. To protect from disease, the human immune system has evolved to recognize and attack cells from other individuals. As such, before donation, we must check whether a donor and recipient are compatible. Donor-patient compatibility is determined by weighing many factors, but one important factor is HLA match compatibility.

HLA Matching Human Leukocyte Antigens (HLAs) are proteins that attach to the membranes of cells to help the immune system identify which cells are "self" cells and which are "invader" cells. Each person has a collection of different HLAs, and the more HLAs that are shared between the donor and the recipient the safer the transplant will be. The similarity is measured by a "HLA match score", and we will consider a patient and donor to be compatible in the case that their match score is at least 60.

Because of all the factors at play for donor-recipient compatibility, it may be the case that a recipient does not know any compatible donors. To help that patient find a donor, donation networks have been set up. The idea is that if Recipient A needs a kidney, but their friend Donor A is not compatible, then perhaps Donor A could give a kidney to compatible Recipient B in exchange for Donor B giving a kidney to Recipient A. So in short, if Recipient A has no known donors, they can try to identify a "kidney trade" with another recipient using one of their donors. These trades could also be larger "kidney donation cycles", where Donor A gives to Recipient B, Donor B gives to recipient C, and Donor C gives to recipient A. For the sake of this assignment, our cycles can be of any length, but they must be cycles (the idea being that Donor A is only willing to donate a kidney if the result is Recipient A receiving one).

Problem Statement

For this assignment you will implement an algorithm to identify whether a given recipient can receive a kidney through a donation cycle.

Kidney Donation Cycle Definition a sequence of recipients $(r_0, r_1, r_2, \dots, r_{n-1})$ is a *kidney donation cycle of length n* provided:

- all of r_1, \dots, r_{n-1} are kidney recipients,
- for each choice of $0 \leq i < n - 1$ there exists a donor associated with r_i is compatible with r_{i+1} ,
- and a donor associated with r_{n-1} is compatible with r_0 .

Input Your input will be string (which you can think of as the contents of a file) which contains the following information:

- One row giving the number of recipients (call this n). Each recipient will be identified by a number between 0 and $n - 1$ (inclusive)
- One row giving the number of donors (call this m). Each donor will be identified by a number between 0 and $m - 1$ (inclusive)
- One row containing m comma-separated integers between 0 and $n - 1$ indicating the recipient that each donor is associated with.
- A table giving the HLA match points for every Donor-Recipient pair, represented as a sequence of m rows each with n comma-separated integers.
- One row giving a "query recipient"

Overall, the input will have $m + 4$ lines.

Output Your code will print just a single Boolean value (True or False in python, true or false in java) indicating whether there exists a kidney donation cycle which includes the query recipient.

Running Time The worst-case asymptotic running time of your program should belong to $O(n \cdot m)$, where n is the number of recipients and m is the number of donors (if it helps you to verify your running time, observe that it must be that $m \geq n$).

Example

This graph would be the result of the following text. We have an edge from recipient 0 to 1 because donor 2 has a match score of 71 with recipient 1, and donor 2 is the friend of recipient 0. There is an edge from recipient 1 to recipient 2 because donor 6 has a match score of 100 with recipient 2, and donor 6 is friends with recipient 1. There is an edge from recipient 2 to recipient 0 because donors 7 and 8 have a match score of 90 with recipient 0.

```
3
9
0,0,0,1,1,1,1,2,2
10,15,20
20,20,9
0,71,20
2,2,2
3,3,3
4,4,4
5,5,100
90,0,0
90,0,0
0
```

