CS 4820, Spring 2017 **Homework 1, Problem 2**
Name: Yuxiang Peng
NetID: yp344
Collaborators:jl3455, zl542

(2) *(10 points)*
You've decided to put your algorithms skills to good use by helping an international team of jewel thieves. (Possibly you've been interpreting "greedy algorithms" too literally.) They've hatched an elaborate plan to bring their stolen goods to the black market. It involves n heists, each committed by a different thief to obtain a parcel of stolen jewels, and n couriers, each of whom must receive a parcel from one of the thieves. For each thief i and each courier j, there is a designated day $D(i, j)$ on which they are to meet; these n^2 meetings all take place on different days. At any such meeting, if the thief is in possession of a parcel, it may be transferred to the courier. If such a transfer takes place, the courier remains in possession of the parcel after that. (Parcels are never transferred from couriers back to thieves.)

The thieves and couriers are tremendously fearful that the police will obtain evidence of a connection between the heists. Accordingly, they've specified some rules that must be obeyed to escape detection:

1. A courier can never be in possession of more than one parcel.
2. When thief i and courier j meet on day $D(i, j)$, the two criminals, combined, must be in possession of *at most one parcel*.

They have asked you to help by designing a *transfer schedule* that specifies, for each day $D(i, j)$, whether a transfer should take place on that day. It's not obvious that one can always design a transfer schedule to satisfy the above constraints, but in fact it is possible to do so. Design an efficient algorithm to compute a transfer schedule that satisfies all of the above constraints, given an input that specifies the date $D(i, j)$ for each i, j.

This question is analogous to the stable matching problem taught in the lecture. The n thieves could be seen as n men. The n couriers could be seen as n women. Thieves preference list for the couriers could be defined in a way that the first courier he meets has the highest preference, the second courier he meets has the second highest preference and so on. The preference list for the couriers could be defined in an opposite way: the first thief he meets has the lowest preference, the second thief he meets has the second lowest preference and so on. The unstable situation here is that both the thief and the courier have a jewel when they meet. The transfer schedule is the one to one matching between the thief and the courier similar to the one to one matching between the men and women.

The algorithm is as follow:
1. Select a thief t that has a jewel on hand
2. t meets a courier c who he most prefers and he has not met
3. c gets the jewel from t if c does not has a jewel or c prefers t over the owner of c's current jewel.

The pseudocode is as follow:

```
Initially all thieves have jewels and no couriers have jewels
While there is a thief t who is has a jewel and hasn't tried every courier for transfer
.       choose such a thief t
.       let courier t be the most preferred courier in t's preference list to which t hasn't tried to
transfer the jewel
.       If c does not have a jewel then
.            c gets t's jewel
.       Else c has a jewel from t'
.           If c prefers t' to t then
.                t keeps the jewel
.           Else c prefers t to t'
.                c takes t's jewel
.                t' gets back its jewel
.           Endif
.      Endif
Endwhile
```