

zad1. Mag planuje zabić wszystkich mieszkańców pewnego królestwa i w tym celu musi odprawić serię rytuałów. W królestwie znajduje się n wież połączonych teleportami w jedną stronę i w każdej musi rzucić zaklęcie, zaś każde zaklęcie musi być częścią dokładnie jednego rytuału. Ponieważ mag boi się, że ktoś może próbować pokrzyżować jego plany, podczas wykonywania jednego rytuału musi być w stanie wrócić się do wieży w której już rzucił zaklęcie należące do tego samego rytuału. Ponieważ każdy rytuał zużywa manę, to mag chce wykonać ich jak najmniej. Dodatkowo, aby zakończyć rytuał, mag musi ponownie rzucić zaklęcie w wieży w której zaczął. Po wykonaniu rytuału w danym zbiorze wieży nie może już do nich wracać. Napisz funkcję `nadgorliwy_mag(G)`, gdzie G to macierz sąsiedztwa reprezentująca teleportsy między wieżami, która zwróci listę przedstawiającą kolejność w jakiej mag powinien odwiedzić wieże.

zad2. Mag pomimo tego że jest wyśmienitym czarownikiem to czasem lubi się napić. I tak się stało dzień przed oblężeniem. Odprawienie rytuałów pozostało tobie. Znalazłeś encyklopedię magową. Litery rozumiesz ale rozmieszczenie ich w alfabecie jest dla ciebie obce. Jak wiadomo słowa w encyklopedii są posortowane leksykograficznie. Napisz funkcję `pijany_mag(T)`, która wywiedzie kolejność leksykograficzną liter w tym języku, jeżeli taka kolejność jest niemożliwa zwróć pusty string. Zakładamy że litery są od a do z bez polskich znaków.

Przykład:

`T = ["z", "x"]`

`odp = "zx"`

`T = ["z", "x", "z"]`

`odp = ""`

zad3. Drugi mag potrzebuje pomocy, bo jest słabszym magiem niż pierwszy, ale wie że jeżeli uda mu się zakreślić obszar domknięty na grafie skierowanym w postaci macierzowej to uda mu się obronić niewidzialne miasta w środku. Jednak mag ten wstawił wodę na herbatę w domu więc musi to zrobić najkrótszą trasą. Pomóż magowi znaleźć najkrótszą trasę która zakreśla obszar domknięty. Napisz funkcję `gorszy_mag(G)`, która znajdzie taką trasę w tablicy lub `None` jeżeli taka trasa nie istnieje.