

# Тренировки по алгоритмам 3.0 от Яндекса — Дивизион В

🕒 4 апр 2024, 02:44:45  
старт: 28 окт 2023, 21:14:17  
финиш: 26 ноя 2023, 03:14:16  
длительность: 28д. 5ч.  
...

Объявления жюри

📘 Ваше участие в соревновании завершено. Вы можете дорешивать задачи и отправлять решения вне соревнования

Положение участниковЗадачиПосылки

## 40. Метро

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Метрополитен состоит из нескольких линий метро. Все станции метро в городе пронумерованы натуральными числами от 1 до N. На каждой линии расположено несколько станций. Если одна и та же станция расположена сразу на нескольких линиях, то она является станцией пересадки и на этой станции можно пересесть с любой линии, которая через нее проходит, на любую другую (опять же проходящую через нее).  
Напишите программу, которая по данному вам описанию метрополитена определит, с каким минимальным числом пересадок можно добраться со станции A на станцию B. Если данный метрополитен не соединяет все линии в одну систему, то может так получиться, что со станции A на станцию B добраться невозможно, в этом случае ваша программа должна это определить.

### Формат ввода

Сначала вводится число N — количество станций метро в городе ( $2 \leq N \leq 100$ ). Далее следует число M — количество линий метро ( $1 \leq M \leq 20$ ). Далее идет описание M линий. Описание каждой линии состоит из числа  $P_i$  — количество станций на этой линии ( $2 \leq P_i \leq 50$ ) и  $P_i$  чисел, задающих номера станций, через которые проходит линия (ни через какую станцию линия не проходит дважды).  
Затем вводятся два различных числа: A — номер начальной станции, и B — номер станции, на которую нам нужно попасть. При этом если через станцию A проходит несколько линий, то мы можем спуститься на любую из них. Так же если через станцию B проходит несколько линий, то нам не важно, по какой линии мы приедем.

### Формат вывода

Выведите минимальное количество пересадок, которое нам понадобится. Если добраться со станции A на станцию B невозможно, программа должна вывести одно число −1 (минус один).

### Пример

Ввод	Вывод
5 2 4 1 2 3 4 2 5 3 3 1	0

Язык Python 3.12.1

Набрать здесьОтправить файл

```
1 from collections import defaultdict, deque
2
3 fin = open('input.txt')
4 N = int(fin.readline())
5 M = int(fin.readline())
6
7 stations, color_of_station = [[] for i in range(N)], defaultdict(set)
8 for i in range(M):
9     num, *points = [int(x)-1 for x in fin.readline().split()]
10    num = num+1
11
12    for point in points:
13        color_of_station[point].add(i)
14
15        for other_point in points:
16            if point != other_point: stations[point].append(other_point)
17
18 start, end = [int(x)-1 for x in fin.readline().split()]
19
20 def getMinPath(stations, start, end, color_of_station):
21     if start == end: return 0
22
23     stack, distance = deque(), defaultdict(lambda: float('inf'))
24     distance[start] = 0
25
26     for color in color_of_station[start]:
27         stack.append((start, color, 0))
28
29     while stack:
30         curr_station, curr_color, changes = stack.pop()
31
32         for next_station in stations[curr_station]:
33             if next_station == end:
34                 if curr_color in color_of_station[next_station]:
35                     distance[next_station] = min(changes, distance[next_station])
36                 else:
37                     distance[next_station] = min(changes+1, distance[next_station])
38
```

Отправить📘 осталось 100 попыток

Предыдущая

Время посылки	ID	Задача	Компилятор	Вердикт	Тип посылки	Время	Память	Тест	Баллы	
30 окт 2023, 05:06:14	95160233	40	Python 3.12.1	OK	-	73ms	4.40Mb	-	-	<a href="#">отчёт</a>
30 окт 2023, 05:05:49	95160226	40	Python 3.12.1	PE	-	54ms	4.12Mb	1	-	<a href="#">отчёт</a>
30 окт 2023, 04:26:28	95159519	40	Python 3.12.1	WA	-	54ms	4.15Mb	11	-	<a href="#">отчёт</a>
30 окт 2023, 04:23:03	95159443	40	Python 3.12.1	WA	-	58ms	4.15Mb	11	-	<a href="#">отчёт</a>
30 окт 2023, 04:22:07	95159417	40	Python 3.12.1	WA	-	53ms	4.14Mb	2	-	<a href="#">отчёт</a>
30 окт 2023, 04:21:16	95159404	40	Python 3.12.1	WA	-	53ms	4.20Mb	2	-	<a href="#">отчёт</a>
30 окт 2023, 04:16:46	95159314	40	Python 3.12.1	WA	-	55ms	4.15Mb	11	-	<a href="#">отчёт</a>
30 окт 2023, 04:12:01	95159191	40	Python 3.12.1	WA	-	53ms	4.14Mb	6	-	<a href="#">отчёт</a>
30 окт 2023, 04:02:21	95158930	40	Python 3.12.1	TL	-	1.09s	4.27Mb	2	-	<a href="#">отчёт</a>
30 окт 2023, 04:00:46	95158887	40	Python 3.12.1	WA	-	53ms	4.16Mb	2	-	<a href="#">отчёт</a>

- ✓

1. Гистограмма
- ✓

2. Красивая строка
- ✓

3. Коллекционер Диего
- ✓

4. Контрольная работа
- ✓

5. Хорошая строка
- ✓

6. Операционные системы lite
- ✓

7. SNTP
- ✓

8. Минимальный прямоугольник
- ✓

9. Сумма в прямоугольнике
- ✓

10. Скучная лекция
- ✓

11. Стек с защитой от ошибок
- ✓

12. Правильная скобочная последовательность
- ✓

13. Постфиксная запись
- ✓

14. Сортировка вагонов lite
- ✓

15. Великое Лайнландское переселение
- ✓

16. Очередь с защитой от ошибок
- ✓

17. Игра в пьяницу
- ✓

18. Дек с защитой от ошибок
- ✓

19. Хипуй
- ✓

20. Пирамидальная сортировка
- ✗

21. Три единицы подряд
22. Кузнечик
23. Калькулятор
24. Покупка билетов
25. Гвоздики
26. Самый дешевый путь
27. Вывести маршрут максимальной стоимости
28. Ход конём
29. Кафе
30. НОП с восстановлением ответа
- ✓

31. Поиск в глубину
- ✓

32. Компоненты связности
- ✓

33. Списывание
- ✓

34. Топологическая сортировка
- ✓

35. Поиск цикла
- ✓

36. Длина кратчайшего пути
- ✓

37. Путь в графе
- ✓

38. Блохи
- ✓

39. Путь спелеолога
- ✓

40. Метро