

# Тренировки по алгоритмам 3.0 от Яндекса — Дивизион В

🕒 4 апр 2024, 02:43:21  
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 участников Задачи Псылки

## 31. Поиск в глубину

	Все языки	Python 3.6
Ограничение времени	2 секунды	5 секунд
Ограничение памяти	256Mb	256Mb
Ввод	стандартный ввод или input.txt	
Вывод	стандартный вывод или output.txt	

Дан неориентированный граф, возможно, с петлями и кратными ребрами. Необходимо построить компоненту связности, содержащую первую вершину.

### Формат ввода

В первой строке записаны два целых числа  $N$  ( $1 \leq N \leq 10^3$ ) и  $M$  ( $0 \leq M \leq 5 \cdot 10^5$ ) — количество вершин и ребер в графе. В последующих  $M$  строках перечислены ребра — пары чисел, определяющие номера вершин, которые соединяют ребра.

### Формат вывода

В первую строку выходного файла выведите число  $K$  — количество вершин в компоненте связности. Во вторую строку выведите  $K$  целых чисел — вершины компоненты связности, перечисленные в порядке возрастания номеров.

### Пример

Ввод	Вывод
4 5 2 2 3 4 2 3 1 3 2 4	4 1 2 3 4

Язык Python 3.9 (PyPy 7.3.11) ▾

Набрать здесь Отправить файл

```
1 from collections import defaultdict
2
3 fin = open('input.txt')
4 fin.readline()
5
6 graph = defaultdict(set)
7 for line in fin:
8     if not line.rstrip(): break
9     x, y = [int(x) for x in line.split()]
10    graph[x].add(y)
11    graph[y].add(x)
12
13 stack, visited = [1], set([1])
14 while stack:
15     pos = stack.pop()
16
17     for next_pos in graph[pos]:
18         if next_pos not in visited:
19             visited.add(next_pos)
20             stack.append(next_pos)
21
22 print(len(visited))
23 print(*sorted(visited))
```

Отправить 📘 осталось 100 попыток

Предыдущая

Следующая

- ✓ 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

Время посылки	ID	Задача	Компилятор	Вердикт	Тип посылки	Время	Память	Тест	Баллы	
29 окт 2023, 18:58:01	95093095	31	Python 3.9 (PyPy 7.3.11)	OK	-	0.715s	101.09Mb	-	-	<a href="#">отчёт</a>
29 окт 2023, 18:43:04	95090674	31	Python 3.9 (PyPy 7.3.11)	WA	-	190ms	27.38Mb	1	-	<a href="#">отчёт</a>
29 окт 2023, 18:40:04	95090200	31	Python 3.9 (PyPy 7.3.11)	WA	-	194ms	28.10Mb	2	-	<a href="#">отчёт</a>
29 окт 2023, 18:39:11	95090069	31	Python 3.9 (PyPy 7.3.11)	WA	-	204ms	28.09Mb	4	-	<a href="#">отчёт</a>
29 окт 2023, 18:38:39	95089993	31	Python 3.9 (PyPy 7.3.11)	WA	-	192ms	28.10Mb	4	-	<a href="#">отчёт</a>
29 окт 2023, 18:37:00	95089740	31	Python 3.9 (PyPy 7.3.11)	RE	-	214ms	28.09Mb	1	-	<a href="#">отчёт</a>
29 окт 2023, 18:36:19	95089657	31	Python 3.9 (PyPy 7.3.11)	RE	-	229ms	28.09Mb	1	-	<a href="#">отчёт</a>