

Тренировки по алгоритмам 3.0 от Яндекса — Дивизион В

🕒 4 апр 2024, 02:43:41
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 Посылки

33. Списывание

	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	

Во время контрольной работы профессор Флойд заметил, что некоторые студенты обмениваются записками. Сначала он хотел поставить им всем двойки, но в тот день профессор был добрым, а потому решил разделить студентов на две группы: списывающих и дающих списывать, и поставить двойки только первым. У профессора записаны все пары студентов, обменявшихся записками. Требуется определить, сможет ли он разделить студентов на две группы так, чтобы любой обмен записками осуществлялся от студента одной группы студенту другой группы.

Формат ввода

В первой строке находятся два числа N и M — количество студентов и количество пар студентов, обменивающихся записками ($1 \leq N \leq 10^2$, $0 \leq M \leq N(N-1)/2$). Далее в M строках расположены описания пар студентов: два числа, соответствующие номерам студентов, обменивающихся записками (нумерация студентов идёт с 1). Каждая пара студентов перечислена не более одного раза.

Формат вывода

Необходимо вывести ответ на задачу профессора Флойда. Если возможно разделить студентов на две группы - выведите YES; иначе выведите NO.

Пример 1

Ввод	Вывод
3 2 1 2 2 3	YES

Пример 2

Ввод	Вывод
3 3 1 2 2 3 1 3	NO

Язык Python 3.12.1

Набрать здесь Отправить файл

```
1 from collections import defaultdict
2
3 fin = open('input.txt')
4 N, M = [int(x) for x in fin.readline().split()]
5
6 graph = defaultdict(set)
7 for line in fin:
8     x, y = [int(x) for x in line.split()]
9     graph[x].add(y)
10    graph[y].add(x)
11
12 def answer(graph):
13     visited, colors = set(), dict()
14
15     for student in graph:
16         if student not in visited:
17             visited.add(student)
18             colors[student] = 0
19
20             stack = [(student)]
21             while stack:
22                 curr_student = stack.pop()
23
24                 for next_student in graph[curr_student]:
25                     if next_student not in colors:
26                         colors[next_student] = (colors[curr_student]+1)%2
27                     else:
28                         if colors[next_student] != (colors[curr_student]+1)%2: return False
29
30                     if next_student not in visited:
31                         visited.add(next_student)
32                         stack.append((next_student))
33
34     return True
35
36 print('YES') if answer(graph) else print('NO')
```

Отправить 📘 осталось 100 попыток

Предыдущая

Следующая

Время посылки	ID	Задача	Компилятор	Вердикт	Тип посылки	Время	Память	Тест	Баллы	
29 окт 2023, 20:40:45	95109499	33	Python 3.12.1	OK	-	61ms	4.26Mb	-	-	отчёт
29 окт 2023, 20:39:16	95109298	33	Python 3.12.1	OK	-	60ms	4.26Mb	-	-	отчёт
29 окт 2023, 20:16:07	95105604	33	Python 3.12.1	WA	-	59ms	4.14Mb	9	-	отчёт
29 окт 2023, 20:15:03	95105454	33	Python 3.12.1	WA	-	55ms	4.14Mb	2	-	отчёт
29 окт 2023, 19:58:41	95102897	33	Python 3.12.1	TL	-	2.075s	4.15Mb	13	-	отчёт
29 окт 2023, 19:54:59	95102347	33	Python 3.12.1	TL	-	2.078s	4.19Mb	13	-	отчёт
29 окт 2023, 19:54:13	95102229	33	Python 3.9 (PyPy 7.3.11)	TL	-	2.065s	28.32Mb	13	-	отчёт

- ✓

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 дешёвый 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