

Тренировки по алгоритмам 3.0 от Яндекса — Дивизион В

4 апр 2024, 02:40:40
старт: 28 окт 2023, 21:14:17
финиш: 26 ноя 2023, 03:14:16
длительность: 28д. 5ч.
...

Объявления жюри

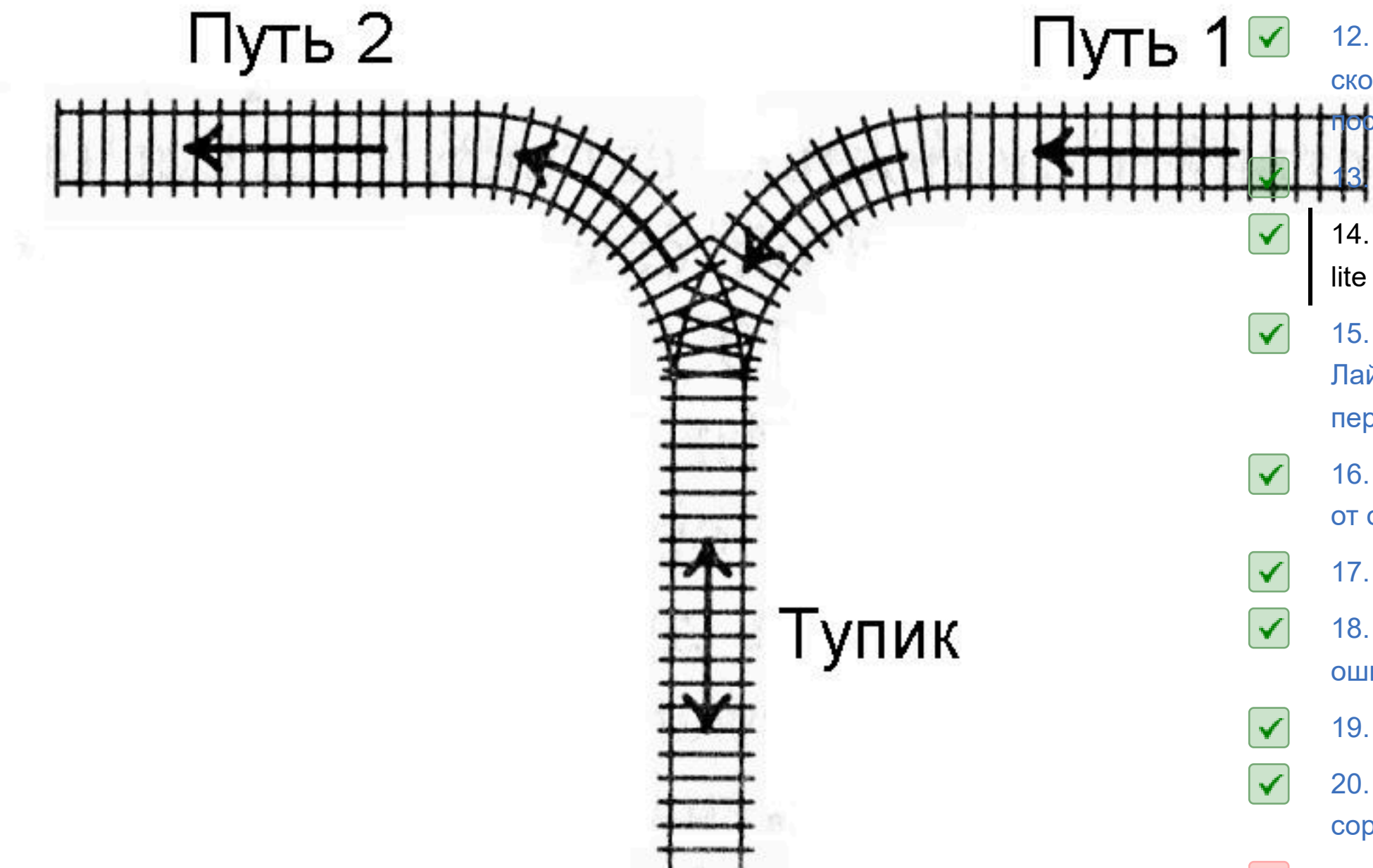
Ваше участие в соревновании завершено. Вы можете дорешивать задачи и отправлять решения вне соревнования

Положение участников Задачи Посылки

14. Сортировка вагонов lite

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

К тупику со стороны пути 1 (см. рисунок) подъехал поезд. Разрешается отцепить от поезда один или сразу несколько первых вагонов и завезти их в тупик (при желании, можно даже завезти в тупик сразу весь поезд). После этого часть из этих вагонов вывезти в сторону пути 2. После этого можно завезти в тупик еще несколько вагонов и снова часть оказавшихся вагонов вывезти в сторону пути 2. И так далее (так, что каждый вагон может лишь один раз заехать с пути 1 в тупик, а затем один раз выехать из тупика на путь 2). Заезжать в тупик с пути 2 или выезжать из тупика на путь 1 запрещается. Нельзя с пути 1 попасть на путь 2, не заезжая в тупик.



Известно, в каком порядке изначально идут вагоны поезда. Требуется с помощью указанных операций сделать так, чтобы вагоны поезда шли по порядку (сначала первый, потом второй и т.д., считая от головы поезда, едущего по пути 1 в сторону тупика). Напишите программу, определяющую, можно ли это сделать.

Формат ввода

Вводится число N — количество вагонов в поезде ($1 \leq N \leq 100$). Далее идут номера вагонов в порядке от головы поезда, едущего по пути 1 в сторону тупика. Вагоны пронумерованы натуральными числами от 1 до N, каждое из которых встречается ровно один раз.

Формат вывода

Если сделать так, чтобы вагоны шли в порядке от 1 до N, считая от головы поезда, когда поезд поедет по пути 2 из тупика, можно, выведите сообщение YES, если это сделать нельзя, выведите NO.

Пример 1

Ввод	Вывод
3 3 2 1	YES

Пример 2

Ввод	Вывод
4 4 1 3 2	YES

Пример 3

Ввод	Вывод
3 2 3 1	NO

- 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

Язык Python 3.9 (PyPy 7.3.11)

Набрать здесь Отправить файл

```
1 fin = open('input.txt')
2 N = int(fin.readline())
3
4 path = [int(x) for x in fin.readline().split()]
5 path.reverse()
6
7 def answer(path):
8     stack = []
9     seen = set()
10
11     curr_number = 1
12     while curr_number < N:
13         if curr_number not in seen:
14             if not path: return 'NO'
15
16             num = path.pop()
17             if num == curr_number:
18                 curr_number += 1
19             else:
20                 stack.append(num)
21                 seen.add(num)
22         else:
23             if not stack or curr_number != stack[-1]:
24                 return 'NO'
25             else:
26                 stack.pop()
27                 curr_number += 1
28
29     return 'YES'
30
31 print(answer(path))
```

Отправить 1 осталось 100 попыток

Предыдущая

Следующая

Время посылки	ID	Задача	Компилятор	Вердикт	Тип посылки	Время	Память	Тест	Баллы
29 окт 2023, 03:56:04	95029924	14	Python 3.9 (PyPy 7.3.11)	OK	-	204ms	28.10Mb	-	- отчёт