

Тренировки по алгоритмам 4.0 от Яндекса — Занятие 4 (Перебор и методы его оптимизации)

2 апр 2024, 06:42:44
старт: 18 ноя 2023, 15:00:00
...

Объявления жюри

Положение участников **Задачи** Посылки

D. Простая задача коммивояжера

Ограничение времени	2 секунды
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Неориентированный взвешенный граф задан матрицей смежности. Найдите кратчайший цикл, который начинается и заканчивается в вершине номер 1 и проходит через все вершины по одному разу.

Формат ввода

В первой строке вводится число N ($N \leq 10$) — количество вершин графа. Следующие N строк содержат по N целых неотрицательных чисел и задают матрицу смежности. Число 0 означает, что ребро отсутствует. Любое другое число задаёт вес ребра.

Формат вывода

Выведите минимальную суммарную длину цикла или число -1, если цикл построить невозможно.

Пример 1

Ввод	Вывод
1 0	0

Пример 2

Ввод	Вывод
2 0 1 1 0	2

Пример 3

Ввод	Вывод
2 0 85 85 0	170

Язык Python 3.9 (PyPy 7.3.11)

Набрать здесь Отправить файл

```
1 fin = open('input.txt')
2 N = int(fin.readline())
3
4 graph = [None]*N
5 for i in range(N):
6     graph[i] = [int(x) for x in fin.readline().split()]
7
8 def findMinPath(graph, start_point, end_point):
9     def findPath(curr_point, curr_dist, visited_count, all_points, remaining_path):
10         nonlocal best_path
11
12         if visited_count == all_points:
13             if graph[curr_point][end_point] != 0:
14                 best_path = min(best_path, curr_dist+graph[curr_point][end_point])
15                 return curr_dist+graph[curr_point][end_point]
16             else:
17                 return float('inf')
18         else:
19             min_path = float('inf')
20
21             for i in range(len(graph[0])):
22                 if not visited[i] and graph[curr_point][i] != 0:
23                     visited[i] = True
24                     if curr_dist+remaining_path < best_path:
25                         min_path = min(min_path, findPath(i, curr_dist+graph[curr_point][i], visited_count+1, all_points, remaining_path-1))
26                     visited[i] = False
27
28             return min_path
29
30 N = len(graph[0])
31 if N == 1: return 0
32 if N == 2:
33     if graph[0][1] != 0 and graph[1][0] != 0:
34         return graph[0][1]+graph[1][0]
35     else:
36         return -1
37
38
```

Отправить *i* осталось 98 попыток

Предыдущая

Следующая

Время посылки	ID	Задача	Компилятор	Вердикт	Тип посылки	Время	Память	Тест	Баллы
22 ноя 2023, 04:05:20	98417395	D	Python 3.9 (PyPy 7.3.11)	OK	-	231ms	28.10Mb	-	-
22 ноя 2023, 03:29:37	98416196	D	Python 3.9 (PyPy 7.3.11)	OK	-	404ms	28.09Mb	-	-