

Тренировки по алгоритмам 4.0 от Яндекса — Занятие 3 (Кратчайшие пути в графах)

2 апр 2024, 06:38:56
старт: 8 ноя 2023, 22:00:00
...

Объявления жюри

Положение участников Задачи Посылки

С. Быстрый алгоритм Дейкстры

Ограничение времени	5 секунд
Ограничение памяти	512Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

- ✓

A. Дейкстра
- ✓

B. Дейкстра с восстановлением пути
- ✓

C. Быстрый алгоритм Дейкстры
- ✓

D. Автобусы в Васюках
- ✓

E. На санях

Вам дано описание дорожной сети страны. Ваша задача – найти длину кратчайшего пути между городами A и B.

Формат ввода

Сеть дорог задана во входном файле следующим образом: первая строка содержит числа N и K ($1 \leq N \leq 100000$, $0 \leq K \leq 300000$), где K – количество дорог. Каждая из следующих K строк содержит описание дороги с двусторонним движением – три целых числа a_i , b_i и l_i ($1 \leq a_i, b_i \leq N$, $1 \leq l_i \leq 10^6$). Это означает, что имеется дорога длины l_i , которая ведет из города a_i в город b_i . В последней строке находятся два числа A и B – номера городов, между которыми надо посчитать кратчайшее расстояние ($1 \leq A, B \leq N$)

Формат вывода

Выведите одно число – расстояние между нужными городами. Если по дорогам от города A до города B доехать невозможно, выведите −1.

Пример

Ввод	Вывод
6 4 1 2 7 2 4 8 4 5 1 4 3 100 3 1	115

Язык Python 3.12.1

Набрать здесь Отправить файл

```
1 import heapq
2 from collections import defaultdict
3
4 fin = open('input.txt')
5 N, K = [int(x) for x in fin.readline().split()]
6
7 graph = defaultdict(list)
8 for i in range(1, K+1):
9     a, b, d = [int(x) for x in fin.readline().split()]
10    graph[a].append((d, b))
11    graph[b].append((d, a))
12
13 start, end = [int(x) for x in fin.readline().split()]
14
15 visited = set()
16 distance = [float('inf')]*(N+1)
17 distance[start] = 0
18
19 stack = [(0, start)]
20 while stack:
21     curr_dist, curr_point = heapq.heappop(stack)
22
23     if curr_point not in visited:
24         for next_dist, next_point in graph[curr_point]:
25             if curr_dist+next_dist < distance[next_point]:
26                 distance[next_point] = curr_dist+next_dist
27                 heapq.heappush(stack, (curr_dist+next_dist, next_point))
28
29     visited.add(curr_point)
30
31 print(distance[end] if distance[end] != float('inf') else print(-1))
```

Отправить ⓘ осталось 97 попыток

Предыдущая

Следующая

Время посылки	ID	Задача	Компилятор	Вердикт	Тип посылки	Время	Память	Тест	Баллы
11 ноя 2023, 04:26:30	96776533	C	Python 3.12.1	OK	-	2.348s	103.35Mb	-	-
11 ноя 2023, 02:35:01	96773045	C	Python 3.9 (PyPy 7.3.11)	OK	-	1.425s	93.30Mb	-	-
11 ноя 2023, 02:31:41	96772861	C	Python 3.9 (PyPy 7.3.11)	RE	-	218ms	28.09Mb	2	-