

Яндекс. Тренировки по алгоритмам 2.0, занятие 8 (B)

C. Родословная: LCA

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

В генеалогическом древе определите для двух элементов их наименьшего общего предка. Наименьшим общим предком элементов A и B является такой элемент C, что C является предком A, C является предком B, при этом глубина C является наибольшей из возможных. При этом элемент считается своим собственным предком.

Формат ввода

Формат входных данных аналогичен предыдущей задаче.

Формат вывода

Для каждого запроса выведите наименьшего общего предка данных элементов.

Пример

Ввод

9
Alexei Peter_I
Anna Peter_I
Elizabeth Peter_I
Peter_II Alexei
Peter_III Anna
Paul_I Peter_III
Alexander_I Paul_I
Nicholaus_I Paul_I
Alexander_I Nicholaus_I
Peter_II Paul_I
Alexander_I Anna

Вывод

Paul_I
Peter_I
Anna

Язык

Python 3.12.1

Набрать здесь

Отправить файл

```

1 import sys
2 sys.setrecursionlimit(100000)
3
4
5 def all_ancestors(child, tree):
6     ancestors = set()
7     while child in tree.keys():
8         ancestors.add(child)
9         child = tree[child]
10    ancestors.add(child)
11    return ancestors
12
13
14 def lca(child, potential_ancestors):
15     while child not in potential_ancestors and child in tree:
16         child = tree[child]
17
18     if child in potential_ancestors:
19         return child
20     else:
21         return ''
22
23
24
25 # считываем данные
26 with open('input.txt', 'r') as f:
27     N = int(f.readline().strip()) # число элементов в генеалогическом древе
28     tree = dict() # генеалогическое древо ребенок:[родитель]
29     answer = [] # массив ответа
30     # добавляем записи в генеалогическое древо, кроме родоначальника
31     for _ in range(N - 1):
32         child, parent = f.readline().split()
33         tree[child] = parent
34     # проходим по данным для поиска элементам
35     for line in f.readlines():
36         human1, human2 = line.strip().split() # человек 1 и человек 2
37         ancestors = all_ancestors(human1, tree) # предки человека 1
38         answer.append(lca(human2, ancestors)) # добавляем в ответ общего предка

```

Отправить

Предыдущая

Следующая