

Антон Макеев

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

9 окт 2021, 12:44:04

старт: 23 сен 2021, 12:00:00

начало: 23 сен 2021, 12:00:00

D. Бусинки

Ограничение времени	2 секунды
Ограничение памяти	256Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Маленький мальчик делает бусы. У него есть много пронумерованных бусинок. Каждая бусинка имеет уникальный номер — целое число в диапазоне от 1 до N . Он выкладывает все бусинки на полу и соединяет бусинки между собой произвольным образом так, что замкнутых фигур не образуется. Каждая из бусинок при этом оказывается соединенной с какой-либо другой бусинкой. Требуется определить, какое максимальное количество последовательно соединенных бусинок присутствует в полученной фигуре.

Формат ввода

В первой строке — количество бусинок $1 \leq N \leq 2500$. В последующих $N-1$ строках по два целых числа — номера, соединенных бусинок.

Формат вывода

Вывести одно число — искомое количество бусинок.

Пример 1

Ввод 2
1 2Вывод

2

Пример 2

Ввод 5
2 1
2 3
2 4
2 5Вывод

3

Пример 3

Ввод Вывод

```
10
1 2
2 3
3 4
4 5
1 6
6 10
10 9
9 8
8 7
```

Язык

```
1 import Foundation
2
3 let n = Int(readLine()!)!
4 var connections = [String: [String]]()
5
6 for _ in 0.. $(n-1)$  {
7     let pair = readLine()!.components(separatedBy: " ")
8     connections[pair[0], default: [String]()].append(pair[1])
9     connections[pair[1], default: [String]()].append(pair[0])
10 }
11
12 func longestTree(from node: String, parent: String? = nil) -> [String] {
13     let currentConnections = connections[node]!
14     if currentConnections.count == 1 && parent != nil {
15         return [node]
16     }
17     var currentLongest = [String]()
18     for connection in currentConnections {
19         if connection == (parent ?? "") {
20             continue
21         }
22         let newChain = longestTree(from: connection, parent: node)
23         currentLongest = currentLongest.count < newChain.count ? newChain : currentLongest
24     }
25     return [node]+currentLongest
26 }
27
28 let firstD = longestTree(from: "1").last!
29 let ans = longestTree(from: firstD).count
30
```