



Przedmiotowy Konkurs Informatyczny LOGIA powołany przez Mazowieckiego Kuratora Oświaty

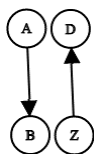
Zadanie Droga – LOGIA 18 (2017/18), etap 3

Treść zadania

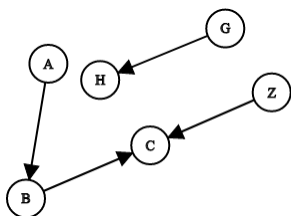
W Turtlandii drogi są jednokierunkowe, przy czym z danej miejscowości wychodzi co najwyżej jedna droga. Koszt przejścia z każdej miejscowości do innej połączonej bezpośrednio z nią wynosi 1. Ania wyrusza z miejscowości A, a Zbyszek z Z.

Napisz jednoparametrową funkcję **droga**, której wynikiem będzie minimalny koszt potrzebny do tego, by Ania i Zbyszek spotkali się w jednej miejscowości lub -1, gdy nie mogą się spotkać. Parametrem funkcji jest lista dwuelementowych list opisujących połączenia między miastami, gdzie pierwszy element pary to identyfikator miejscowości, w której zaczyna się droga, a drugi – w której bezpośrednio się kończy. Identyfikatory miejscowości są wielkimi literami alfabetu łacińskiego. Maksymalna długość listy wynosi 26.

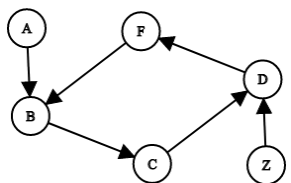
Przykłady:



wynikiem `droga(['A', 'B'], ['Z', 'D'])` jest -1



wynikiem `droga(['A', 'B'], ['G', 'H'], ['B', 'C'], ['Z', 'C'])` jest 3



wynikiem `droga(['Z', 'D'], ['A', 'B'], ['F', 'B'], ['B', 'C'], ['C', 'D'], ['D', 'F'])` jest 4



Przedmiotowy Konkurs Informatyczny LOGIA powołany przez Mazowieckiego Kuratora Oświaty

Omówienie rozwiązania

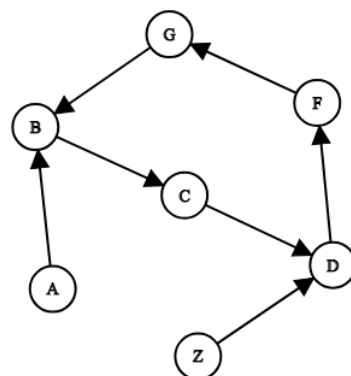
Zgodnie z treścią zadania możemy mieć kilka przypadków:

- Ania i Zbyszek nie spotkają się (jak w przykładzie pierwszym),
- spotkanie będzie w ściśle określonej miejscowości (w przykładzie drugim w C),
- spotkanie może nastąpić w wielu miejscowościach, w przykładzie są to miejscowości B, C, D lub F. Wtedy trzeba wybrać najlepszą (w przykładzie trzecim może to być B lub D – nie zawsze jednak drogi będą jednakowe).

Najpierw znajdujemy ścieżkę dla Ani – najdłuższą drogę jaką może przebyć zaczynając w miejscowości A, potem ścieżkę dla Zbyszka – najdłuższą drogę jaką może przejść z miejscowości Z aż do końca lub do ścieżki Ani. Znajdując cykle, musimy zadbać o to, by wykryć cykl, czyli nie wchodzić do miejscowości już wcześniej odwiedzonej. Wobec tego w każdym kroku sprawdzamy, czy miejscowość nie znajduje się na liście już odwiedzonych.

Jeżeli ścieżki są rozłączne, mamy przypadek pierwszy, wynikiem jest -1. W przeciwnym przypadku Ania i Zbyszek spotkają się. Musimy wtedy sprawdzić, czy gdy zaczniemy badanie ścieżek od Zbyszka nie uzyskamy lepszego rezultatu. Wobec tego znajdujemy najpierw ścieżkę dla Zbyszka, potem dla Ani – czyli jej ścieżkę do końca lub do miejsca, w którym był Zbyszek. Na koniec porównujemy oba wyniki i bierzemy lepszy. Warto zwrócić uwagę, że długość drogi, której szukamy to minimum z dwóch liczb:

- suma długości ścieżki Ani od początku do przecięcia ze ścieżką Zbyszka oraz długość ścieżki Zbyszka,
- suma długości ścieżki Zbyszka od początku do przecięcia ze ścieżką Ani oraz długość ścieżki Ani.



Przypadek, w którym zaczynając liczenie od A i od Z otrzymamy co innego.

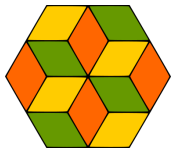


Przedmiotowy Konkurs Informatyczny LOGIA powołany przez Mazowieckiego Kuratora Oświaty

Rozwiązanie w języku Python

Rozwiązanie korzysta z funkcji pomocniczej `ściezka`, w której trzeci parametr ma wartość domyślną – lista pusta.

```
1. def sciezka(start, lista, pom2 = []):
2.     odwiedzone = []
3.     dalej = [e[0] for e in lista]
4.     wsk = start
5.
6.     while True:
7.         if wsk in odwiedzone:
8.             return odwiedzone
9.
10.        if wsk in pom2:
11.            odwiedzone.append(wsk)
12.            return odwiedzone
13.
14.        else:
15.            if wsk in dalej:
16.                nr = dalej.index(wsk)
17.                odwiedzone.append(wsk)
18.                wsk = lista[nr][1]
19.            else:
20.                odwiedzone.append(wsk)
21.                return odwiedzone
22.
23.
24. def droga(lista, starta="A", startb="Z"):
25.     x = sciezka(starta, lista)
26.     y = sciezka(startb, lista, x)
27.     if y[-1] in x:
28.         a = sciezka(startb, lista)
29.         b = sciezka(starta, lista, a)
30.         odl = x.index(y[-1])
31.         odl1 = a.index(b[-1])
32.         return min(odl+len(y)-1, odl1+len(b)-1)
33.     else:
34.         return -1
```



Przedmiotowy Konkurs Informatyczny LOGIA powołany przez Mazowieckiego Kuratora Oświaty

Testy

Testy obejmują przypadki opisane powyżej. Należy rozpatrzyć możliwości:

- gdy Ania i Zbyszek się nie spotkają,
- gdy się spotkają, ale miejsce spotkania nie zależy od kolejności rozpatrywanych ścieżek,
- gdy się spotkają, a miejsce spotkania jest wyznaczone przez ścieżkę Ani lub Zbyszka.

Rozmiar danych nie jest duży – ograniczony przez litery alfabetu.

Wywołanie – Python	Wynik
<code>droga(['D', 'E'], ['A', 'W'], ['G', 'Y'], ['X', 'Y'], ['W', 'X'], ['E', 'F'], ['F', 'G'], ['Z', 'B'], ['C', 'D'], ['B', 'C'])</code>	10
<code>droga([])</code>	-1
<code>droga(['E', 'B'], ['C', 'D'], ['D', 'E'], ['A', 'B'], ['Z', 'X'], ['X', 'C'], ['B', 'G'], ['G', 'C'])</code>	5
<code>droga(['E', 'B'], ['C', 'P'], ['P', 'R'], ['R', 'D'], ['D', 'E'], ['A', 'B'], ['Z', 'X'], ['X', 'C'], ['B', 'G'], ['G', 'W'], ['W', 'T'], ['T', 'U'], ['U', 'H'], ['H', 'S'], ['S', 'C'])</code>	8
<code>droga(['E', 'F'], ['G', 'C'], ['F', 'G'], ['C', 'D'], ['A', 'B'], ['D', 'E'], ['Z', 'H'], ['H', 'K'], ['B', 'C'])</code>	-1
<code>droga(['P', 'M'], ['R', 'M'], ['O', 'R'], ['X', 'Z'], ['M', 'K'], ['A', 'D'], ['D', 'F'], ['F', 'W'], ['K', 'X'], ['Z', 'R'], ['W', 'E'], ['E', 'K'], ['U', 'K'], ['B', 'C'], ['C', 'H'], ['H', 'Z'])</code>	7
<code>droga(['A', 'H'], ['E', 'A'], ['H', 'K'], ['D', 'E'], ['G', 'C'], ['B', 'C'], ['C', 'Z'], ['K', 'F'], ['F', 'B'], ['Z', 'A'])</code>	1
<code>droga(['H', 'B'], ['L', 'B'], ['S', 'B'], ['B', 'C'], ['I', 'C'], ['X', 'D'], ['W', 'E'], ['C', 'H'], ['Q', 'J'], ['T', 'J'], ['E', 'K'], ['M', 'K'], ['U', 'K'], ['N', 'M'], ['P', 'M'], ['R', 'M'], ['Z', 'P'], ['D', 'R'], ['O', 'R'], ['F', 'T'], ['Y', 'T'], ['J', 'V'], ['A', 'W'], ['G', 'X'], ['K', 'X'], ['V', 'K'])</code>	6