

# Algorytm Bellmana Forda

## Bartłomiej Stąpór

### 1)Opis

W pliku ShortestPath.py znajduje się napisana przeze mnie implementacja algorytmu Bellmana Forda w języku Python. W pliku \_\_init\_\_.py są testy jednostkowe. Aby uruchomić program należy uruchomić plik \_\_init\_\_.py w folderze Projekt. Jeśli korzysta się z PyCharm, to należy uruchomić plik znajdujący się w folderze Tests > \_\_init\_\_.py. Funkcja ShortestPath zwraca predykat (poprzednio odwiedzony wierzchołek) oraz minimalny dystans.

### 2)Pseudokod algorytmu:

Bellman-Ford( $G, w, s$ ):

**dla każdego** wierzchołka  $v$  w  $V[G]$  **wykonaj**

$d[v] = \text{nieskończone}$

$\text{poprzednik}[v] = \text{niezdefiniowane}$

$d[s] = 0$

**dla i od 1 do  $|V[G]| - 1$  wykonaj**

**dla każdej** krawędzi  $(u, v)$  w  $E[G]$  **wykonaj**

**jeżeli**  $d[v] > d[u] + w(u, v)$  **to**

$d[v] = d[u] + w(u, v)$

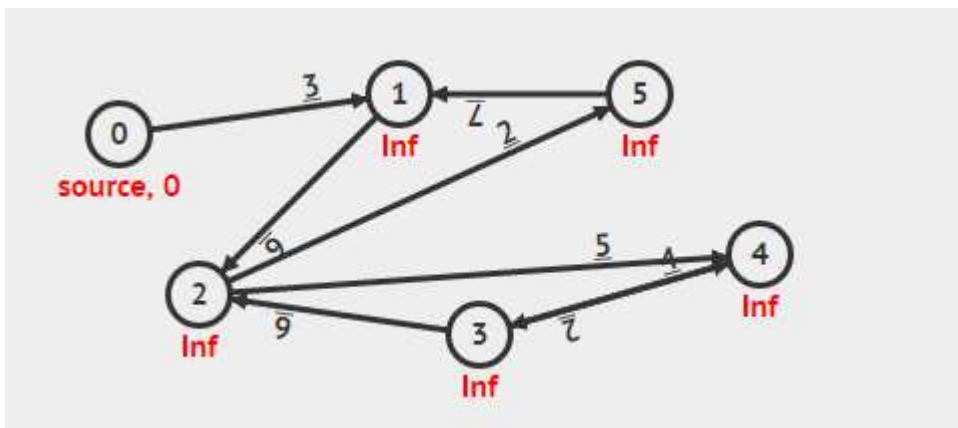
$\text{poprzednik}[v] = u$

Działanie programu porównywałem z algorytmem Bellmana Forda Online(<https://visualgo.net/sssp>). Wyniki się pokrywają.

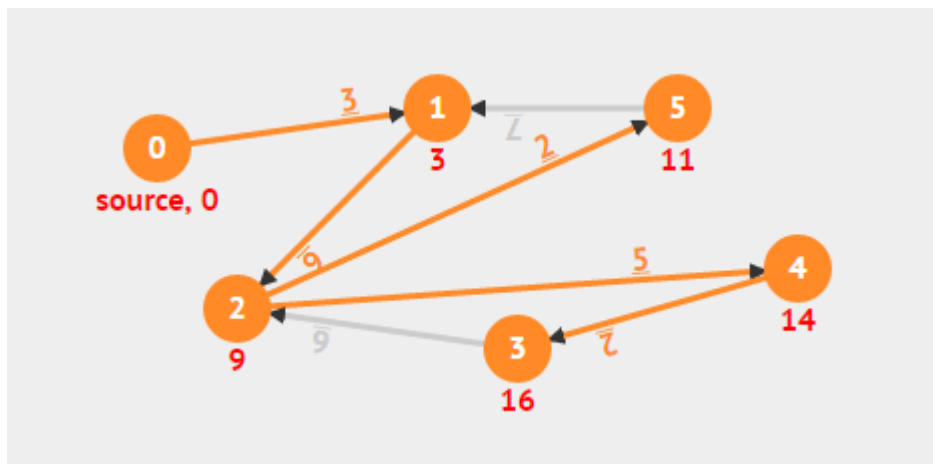
### 3)Testy

#### a)testPositiveEdges

*\*Before*

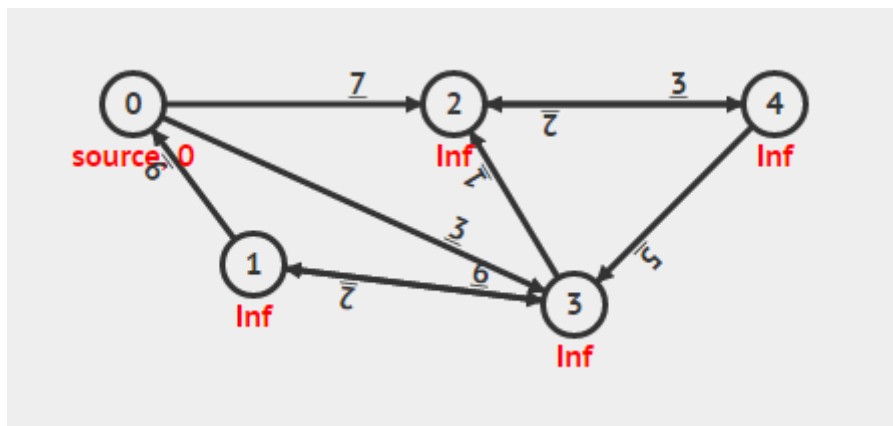


*\*After*

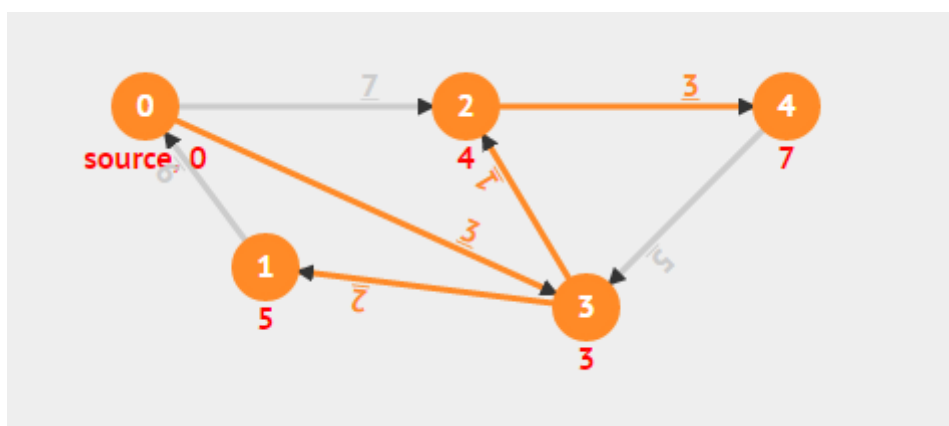


**b)testPositiveEdges2**

*\*Before*

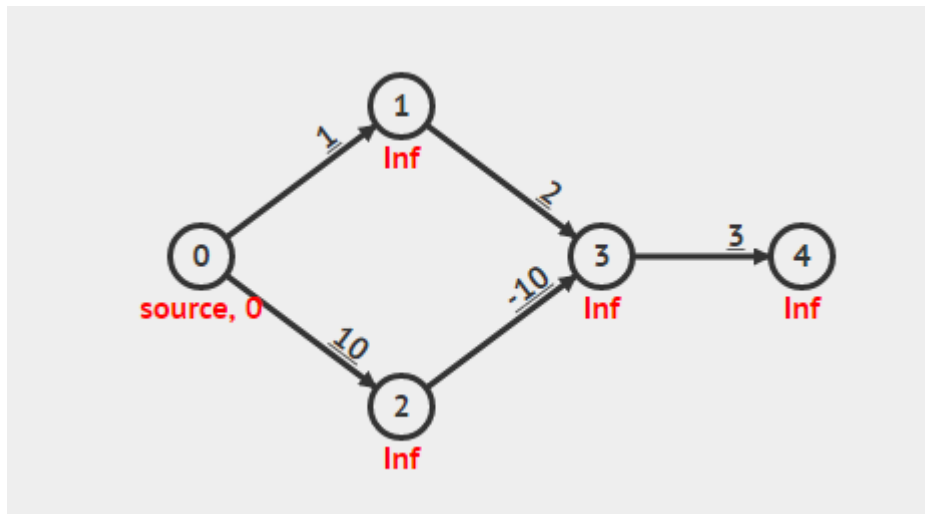


*\*After*

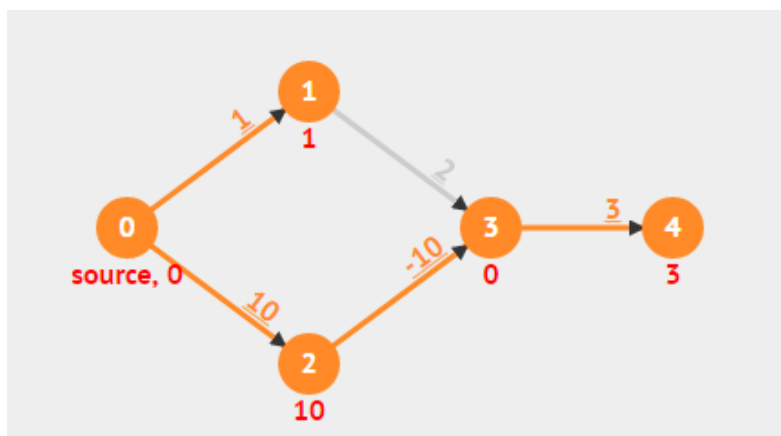


### c)testNegativeEdges

*\*Before*

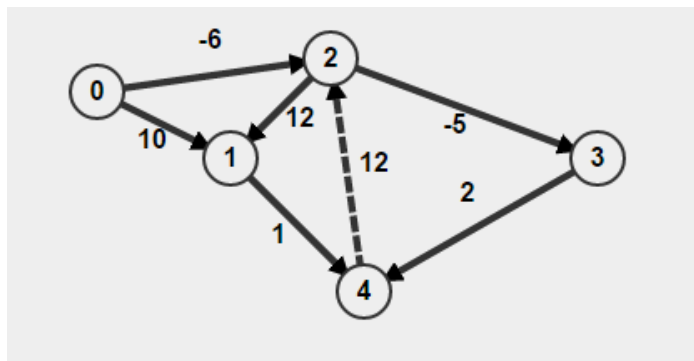


*\*After*

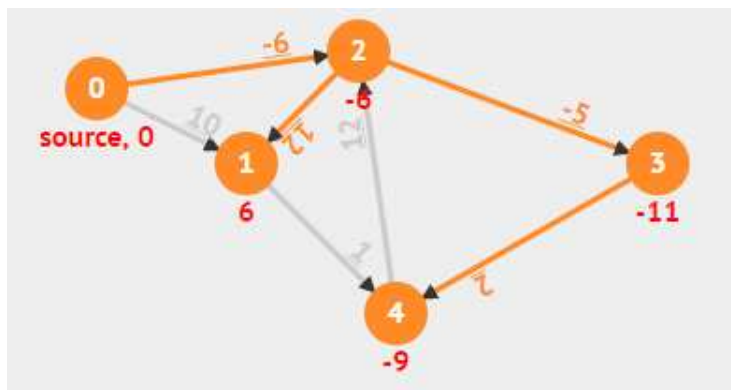


### d)testNegativeEdges2

*\*Before*

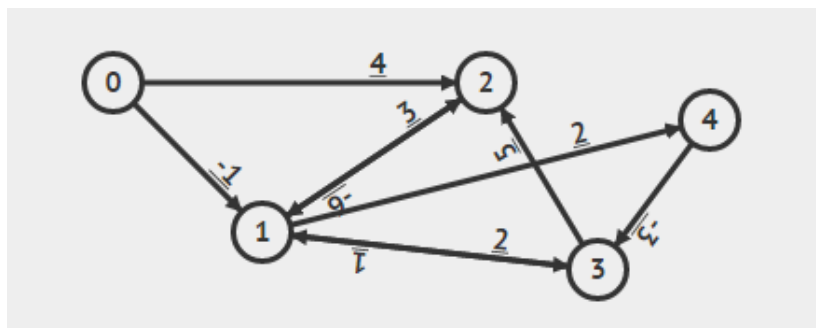


\*After

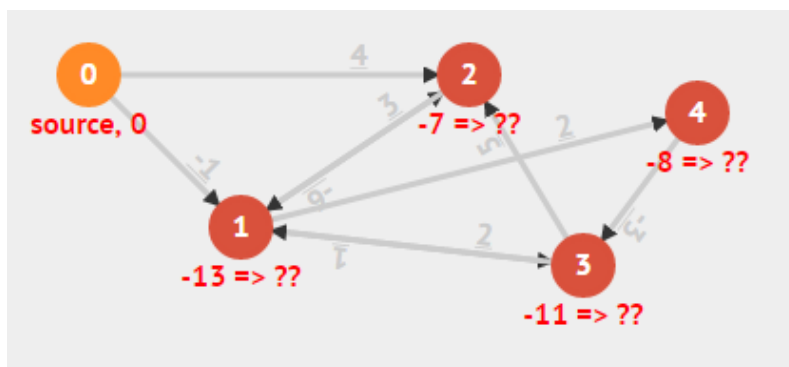


e)testNegativeCycle

\*Before

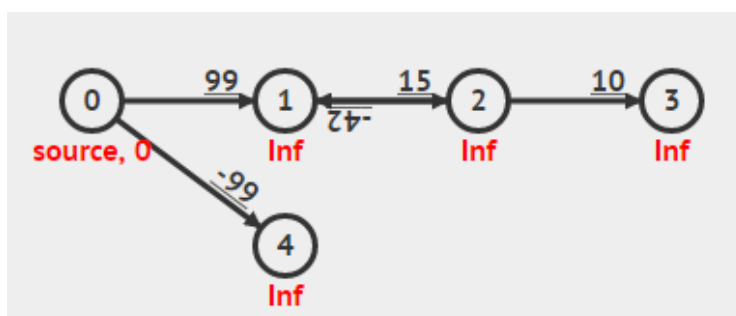


\*After



f)testNegativeCycle2

\*Before



*\*After*

