



POLITECHNIKA RZESZOWSKA

im. Ignacego Łukasiewicza

WYDZIAŁ MATEMATYKI I FIZYKI STOSOWANEJ

Tomasz Nowak

Grupa P05

**Projekt Algorytmy i struktury danych nr. 3**

Rzeszów 2023

# 1. Wstęp

Na repozytorium trzymam swoją całą pracę: <https://github.com/Tnovyloo/Projekt-Studia>

Założeniem zadania projektu było stworzenie grafu w języku programowania oraz stworzenia funkcji do odczytywania informacji z grafu. Podpunkty zadania były następujące:

- Wypisz wszystkich sąsiadów dla każdego wierzchołka grafu (sąsiad wierzchołka  $w_1$  to ten wierzchołek do którego prowadzi krawędź z  $w_1$ )
- Wypisz wszystkie wierzchołki, które są sąsiadami każdego wierzchołka
- Wypisz stopnie wychodzące wszystkich wierzchołków
- Wypisz stopnie wchodzące wszystkich wierzchołków
- Wypisz wszystkie wierzchołki izolowane
- Wypisz wszystkie pętle
- Wypisz wszystkie krawędzie dwukierunkowe

## 2. Pseudokod dla funkcji

Funkcja\_zadanie1 (graf -> słownik)

Dla 'w' w kluczach 'graf'

Jeżeli graf[w]

Wypisz: 'w', 'graf[w]

Jeżeli Nie

Wypisz: Wierzchołek 'w' nie ma sąsiadów

Funkcja\_zadanie2(graf -> słownik)

Ile\_sąsiadów = ilość( klucze w grafie) - 1

Lista\_sąsiadów = []

Dla 'w' w kluczach 'graf'

x = 0

w\_nazwa = tekst(w)

Dla 'w1' w kluczach 'graf'

Jeśli 'w\_nazwa' jest w wartościach kluczu 'w1'

x += 1

Jeśli x == Ile\_sąsiadów

Lista\_sąsiadów += 'w'

Funkcja\_zadanie3(graf -> słownik)

Dla 'w' w kluczach 'graf'

wypisz 'w', 'wartości dla klucza 'w''

Funkcja\_zadanie4(graf -> słownik)

Dla 'w' w kluczach 'graf'

tekst\_w = tekst(w)

wychodzace = []

Dla 'w1' w kluczach 'graf'

Jeśli 'tekst\_w' jest w 'wartościach dla klucza 'w1''

wychodzace += w1

Funkcja\_zadanie5(graf -> słownik)

Izolowane = []

Dla 'w' w 'graf'

Jeśli wartości dla klucza 'w' są puste

w\_tekst = tekst(w)

wychodzace = []

Dla 'w1' w kluczach grafu

Jeśli 'w\_tekst' w wartościach dla klucza 'w1'

wychodzace += w1

Jeśli 'wychodzace' jest puste

izolowane += w

Funkcja\_zadanie6(graf -> słownik)

“ Proszę o wejście na Githuba ( <https://github.com/Tnovyloo/Projekt-Studia> )

Tam wytłumaczyłem swój problem

Funkcja\_zadanie7(graf -> słownik)

dwa\_kierunki = []

Dla 'w' w kluczach 'graf'

Dla 'w1' w wartościach klucza 'w'

aktualny = wartości klucza 'w1'

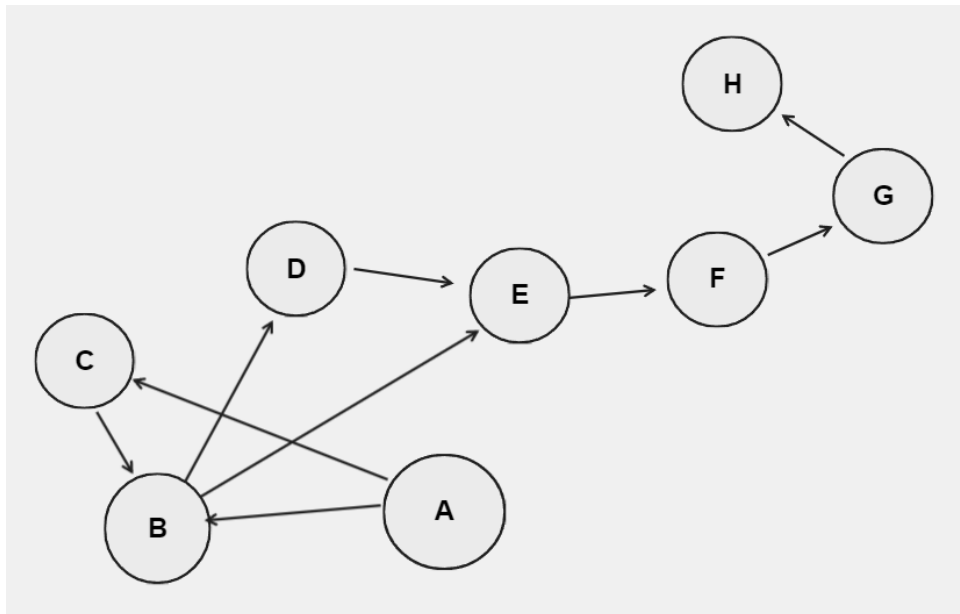
Jeśli 'w' jest w 'aktualny'

dwa\_kierunki += w, w1

### 3. Program

Na repozytorium trzymam swoją całą pracę: <https://github.com/Tnovyloo/Projekt-Studia>

Zadanie 1 dla Grafu:



### Zadanie 1

Dla wierzchołka A sąsiadami są: ['B', 'C']

Dla wierzchołka B sąsiadami są: ['D', 'E']

Dla wierzchołka C sąsiadami są: ['B']

Dla wierzchołka D sąsiadami są: ['E']

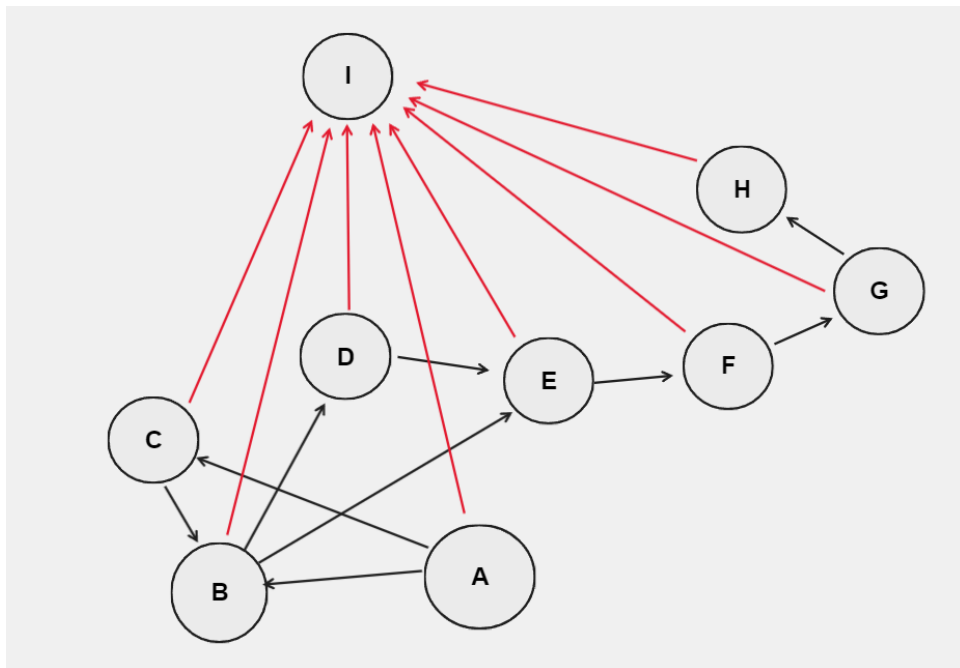
Dla wierzchołka E sąsiadami są: ['F']

Dla wierzchołka F sąsiadami są: ['G']

Dla wierzchołka G sąsiadami są: ['H']

Wierzchołek H nie ma sąsiadów

Zadanie 2 dla grafu:



### Zadanie 2

Wierzchołki będące sąsiadami wszystkich wierzchołków to ['I']

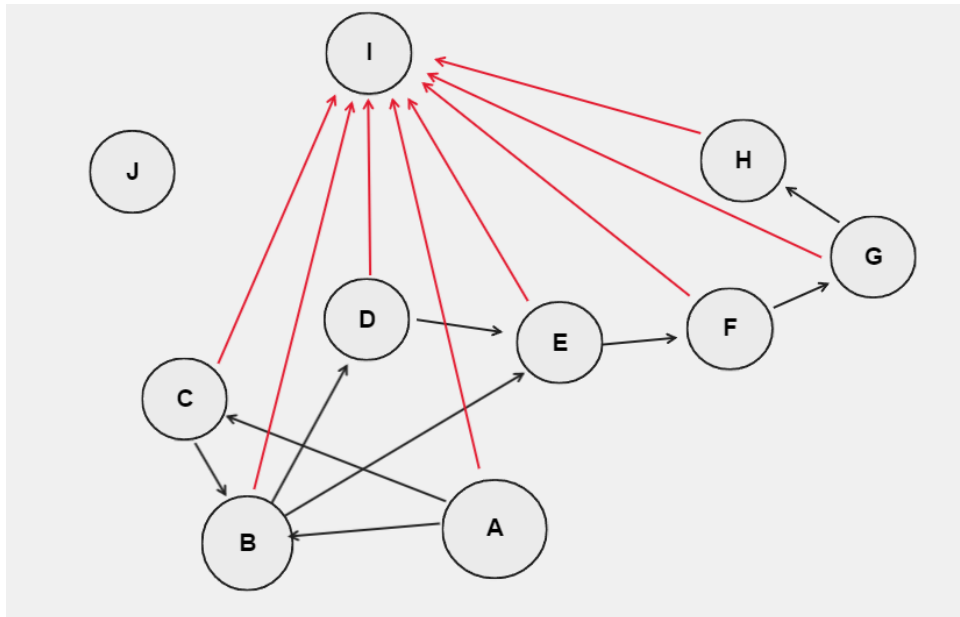
Zadanie 3 dla powyższego grafu:

```
Zadanie 3
Stopnie wychodzące wierzchołka A
A - ['B', 'C', 'I'] - Ilość 3
Stopnie wychodzące wierzchołka B
B - ['D', 'E', 'I'] - Ilość 3
Stopnie wychodzące wierzchołka C
C - ['B', 'I'] - Ilość 2
Stopnie wychodzące wierzchołka D
D - ['E', 'I'] - Ilość 2
Stopnie wychodzące wierzchołka E
E - ['F', 'I'] - Ilość 2
Stopnie wychodzące wierzchołka F
F - ['G', 'I'] - Ilość 2
Stopnie wychodzące wierzchołka G
G - ['H', 'I'] - Ilość 2
Stopnie wychodzące wierzchołka H
H - ['I'] - Ilość 1
Stopnie wychodzące wierzchołka I
I - [] - Ilość 0
```

Zadanie 4 dla powyższego grafu:

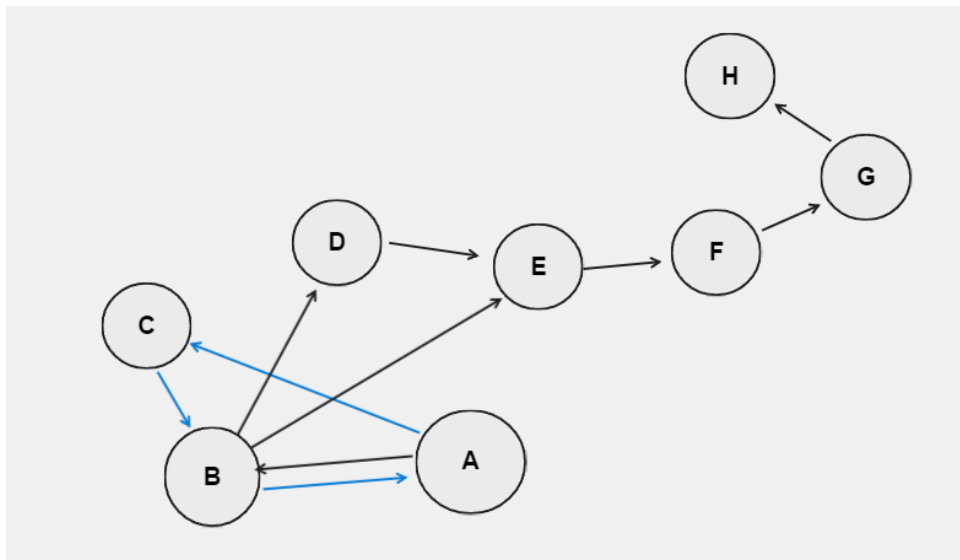
```
Zadanie 4
Wchodzace stopnie do wierzchołka A to []
Wchodzace stopnie do wierzchołka B to ['A', 'C']
Wchodzace stopnie do wierzchołka C to ['A']
Wchodzace stopnie do wierzchołka D to ['B']
Wchodzace stopnie do wierzchołka E to ['B', 'D']
Wchodzace stopnie do wierzchołka F to ['E']
Wchodzace stopnie do wierzchołka G to ['F']
Wchodzace stopnie do wierzchołka H to ['G']
Wchodzace stopnie do wierzchołka I to ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H']
```

Zadanie 5 dla grafu:



```
Zadanie 5  
Wierzchołki izolowane to: ['J']
```

Zadanie 6 dla grafu:



```
Zadanie 6  
Pętla ['B', 'A', 'C']  
Pętla ['B', 'A']
```

Zadanie 7 dla powyższego grafu:



Zadanie 7

Dwukierunkowe: [['A', 'B'], ['B', 'A']]