



Laboratoria Algorytmów i Struktur Danych

Projekt 4 - sprawozdanie

# Algorytmy z powracaniem

Prowadzący: **Dominik Piotr Witczak**

Przygotowali: **Jan Czyżewski 160377,**  
**Albert Łapa 160381**

Grupa: **LAB 13**

Kierunek: **Informatyka, zajęcia pt. 8:00-9:30**



---

**POLITECHNIKA POZNAŃSKA**

---

## 1 | Tworzenie grafu

Program pozwala na wybranie trybu wprowadzania grafu między Hamilton i nie-Hamilton.

```
fred@Fredputer:~/aisd$ /bin/python3 /home/fred/aisd/AiSD/main.py --hamilton
Enter the number of nodes for the Hamiltonian graph: 10
Enter the saturation for the Hamiltonian graph (30 or 70): 70
Enter a command: print
[[0. 1. 0. 1. 0. 1. 1. 1. 0. 1.]
 [1. 0. 1. 1. 0. 0. 1. 1. 1. 0.]
 [0. 1. 0. 1. 1. 1. 1. 1. 1. 1.]
 [1. 1. 1. 0. 1. 1. 1. 0. 1. 0.]
 [0. 0. 1. 1. 0. 1. 1. 1. 1. 1.]
 [1. 0. 1. 1. 1. 0. 1. 1. 1. 0.]
 [1. 1. 1. 1. 1. 1. 0. 0. 1. 1.]
 [1. 1. 1. 0. 1. 1. 0. 0. 1. 0.]
 [0. 1. 1. 1. 1. 1. 1. 1. 0. 1.]
 [1. 0. 1. 0. 1. 0. 1. 0. 1. 0.]]
```

Rysunek 1: Generacja Hamilton

```
fred@Fredputer:~/aisd$ /bin/python3 /home/fred/aisd/AiSD/main.py --non-hamilton
Enter the number of nodes for the non-Hamiltonian graph: 10
Enter a command: print
[[0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 1. 0. 1. 1. 1. 0. 0. 1.]
 [0. 1. 0. 1. 0. 1. 1. 1. 1. 1.]
 [0. 0. 1. 0. 0. 1. 0. 0. 1. 0.]
 [0. 1. 0. 0. 0. 0. 1. 1. 1. 1.]
 [0. 1. 1. 1. 0. 0. 0. 0. 1. 0.]
 [0. 1. 1. 0. 1. 0. 0. 1. 0. 1.]
 [0. 0. 1. 0. 1. 0. 1. 0. 1. 0.]
 [0. 0. 1. 1. 1. 1. 0. 1. 0. 0.]
 [0. 1. 1. 0. 1. 0. 1. 0. 0. 0.]]
```

Rysunek 2: Generacja Nie-Hamilton

Program wyposażono w przejrzyste menu **help**.

```
Enter a command: help
--- Help ---
Commands:
Help - display this message
Quit - exit the program
Print - display the graph in matrix representation
Euler - find an Eulerian cycle in the graph
tikz - export the graph to a LaTeX file
```

Rysunek 3: Menu programu

## 2 | Funkcje programu

```
fred@Fredputer:~/aisd$ /bin/python3 /home/fred/aisd/AiSD/main.py --hamilton
Enter the number of nodes for the Hamiltonian graph: 10
Enter the saturation for the Hamiltonian graph (30 or 70): 70
Enter a command: euler
No Eulerian cycle exists.
```

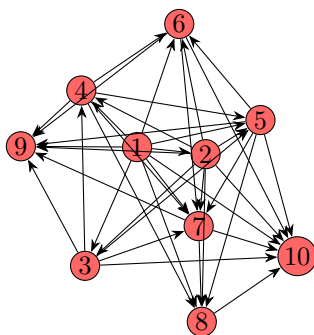
Rysunek 4: Euler

### 3 | Wizualizacje grafów

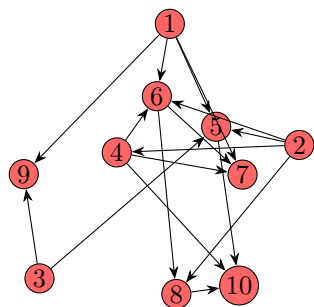
Ekport grafów jest możliwy do tikzpicture za pomocą komendy wraz z własną nazwą. Wygenerowany graf skaluje się w zależności od ilości węzłów.

```
Enter a command: tikz
Enter the filename for the LaTeX file: sussybaka
```

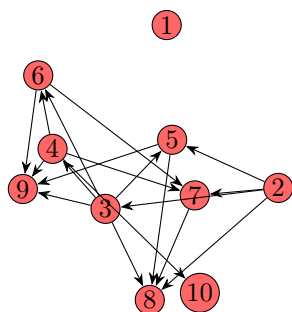
Rysunek 5: Działanie komendy tikz



Rysunek 6: Wizualizacja Hamilton Saturacja 70%



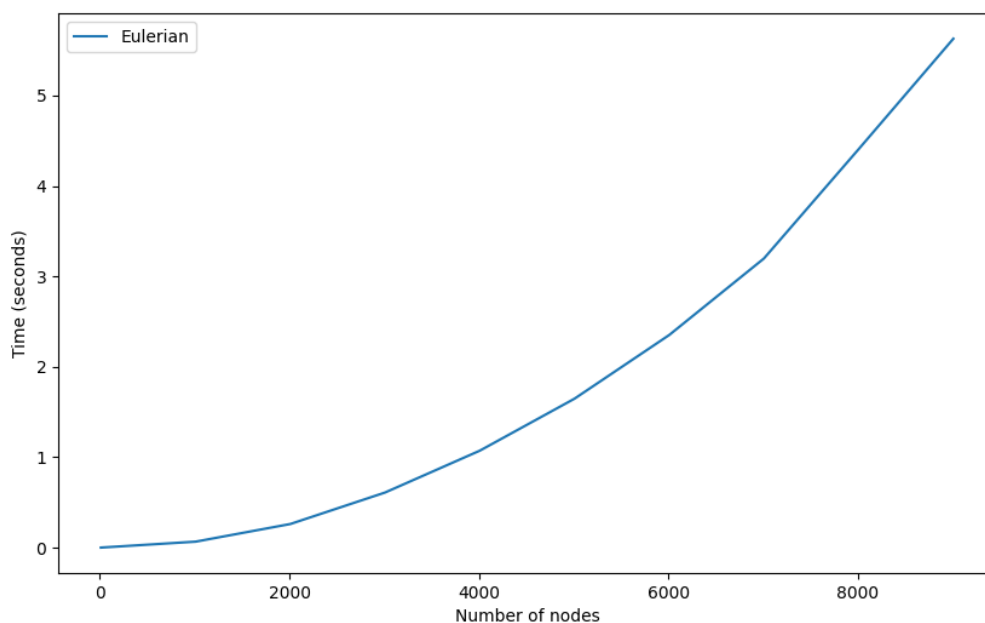
Rysunek 7: Wizualizacja Hamilton Saturacja 30%



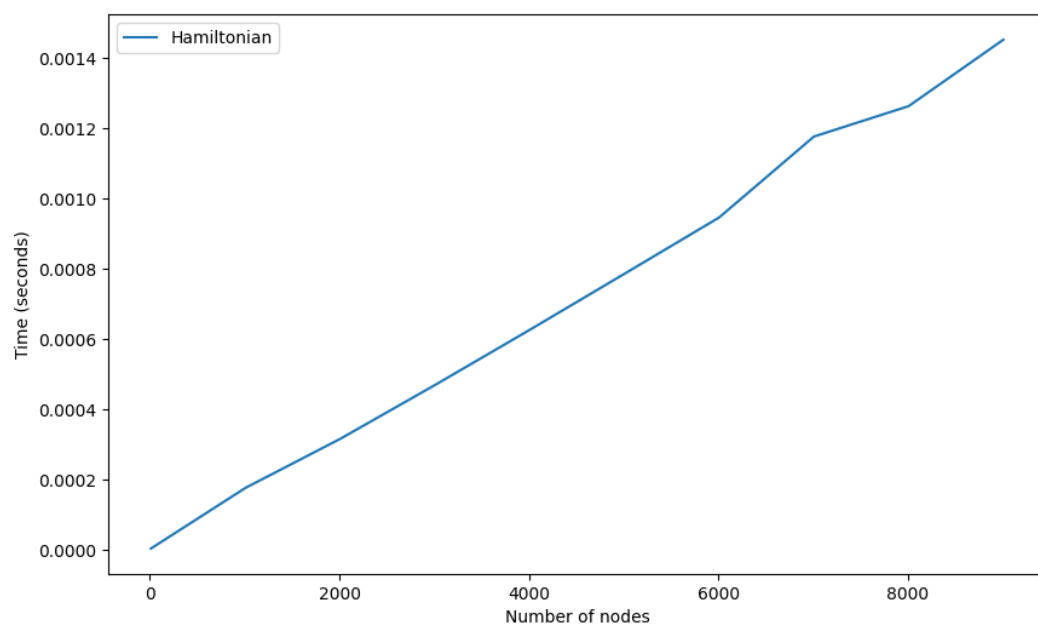
Rysunek 8: Wizualizacja Nie-Hamilton

## 4 | Wykresy

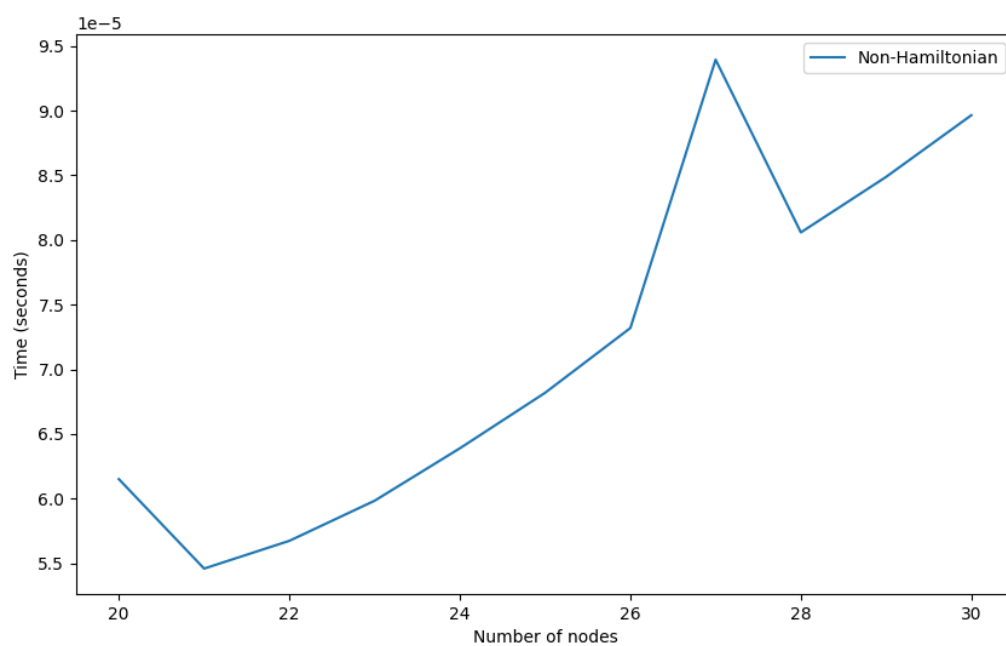
Efektywność algorytmów sprawdzono przy pomocy dostarczonych plików typu benchmark. Komputer testujący wyposażony był w jednostkę obliczeniową Intel Core i5 8500 wraz z 16 GB pamięci operacyjnej. Komputer pracował pod kontrolą systemu operacyjnego Windows 11 Pro 23H2 w środowisku wirtualnym WSL Ubuntu. Testy wykonano na algorytmach napisanych w języku Python w wersji 3.12.2.



Znajdowanie cyklu Eulera



Znajdowanie cyklu Hamiltona



Znajdowanie cyklu Hamiltona w Nie-Hamilton

## 5 | Obserwacje

Rodzaj cyklu	Niskie nasycenie grafu	Wysokie nasycenie grafu
Cykl Eulera w grafie Hamiltonowskim	Znalezienie cyklu Eulera może być trudne lub niemożliwe. Wynika to z faktu, że cykl Eulera wymaga, aby wszystkie wierzchołki miały parzysty stopień.	Prawdopodobieństwo istnienia cyklu Eulera wzrasta, ponieważ zwiększa się szansa, że wszystkie wierzchołki będą miały parzysty stopień.
Cykl Hamiltona w grafie Hamiltonowskim	Znalezienie cyklu Hamiltona może być łatwiejsze, ponieważ mniejsza liczba krawędzi oznacza mniej możliwych ścieżek do sprawdzenia.	Algorytmy mogą mieć trudności ze względu na dużą liczbę możliwych ścieżek, co zwiększa złożoność obliczeniową. Jednakże, ponieważ graf jest Hamiltonowski, cykl Hamiltona zawsze istnieje i algorytm go znajdzie, choć może to zająć więcej czasu.
Cykl Hamiltona w grafie nie Hamiltonowskim	Brak cyklu Hamiltona jest często oczywisty z powodu niewystarczającej liczby krawędzi do utworzenia takiego cyklu. Algorytmy szybko stwierdzają brak cyklu Hamiltona.	Algorytmy mogą potrzebować więcej czasu na potwierdzenie braku cyklu Hamiltona. Duża liczba krawędzi tworzy wiele możliwych ścieżek do sprawdzenia, co zwiększa złożoność obliczeniową. Algorytm musi przeanalizować wiele możliwych kombinacji, aby potwierdzić, że żaden cykl Hamiltona nie istnieje.

Tabela 1: Obserwacje związane z działaniem algorytmów znajdowania cyklu Eulera i Hamiltona w grafach o różnym nasyceniu.