

Laboratoria: piątek, 8:00
Grupa: 13
Informatyka Wydział informatyki i telekomunikacji.

Algorytmy i Struktury Danych
Prowadzacy:
Dominik Witczak

Sprawozdanie do

Projektu 3 Sortowanie topologiczne

Autor:
Marcin Wrzaskowski
nr indeksu:
160329

1 Uzasadnienie wyboru reprezentacji grafu (macierz sąsiedztwa):

1.1 Prostota i przejrzystość:

Macierz sąsiedztwa to dwuwymiarowa tablica, gdzie wiersze i kolumny odpowiadają wierzchołkom, a komórki wskazują na istnienie krawędzi. Jest to prosta i intuicyjna struktura.

1.2 Szybki dostęp do krawędzi:

Sprawdzenie, czy istnieje krawędź między dwoma wierzchołkami, jest operacją $O(1)$, co zapewnia szybki dostęp do informacji o krawędziach.

1.3 Łatwość implementacji algorytmów:

Algorytmy grafowe, takie jak DFS, BFS czy algorytmy najkrótszej ścieżki, są łatwe do zaimplementowania z macierzą sąsiedztwa, co poprawia efektywność czasową.

1.4 Efektywność dla grafów gęstych:

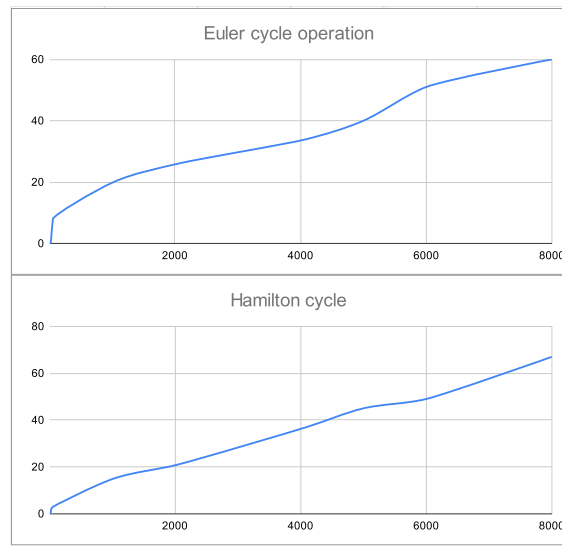
Macierz sąsiedztwa jest szczególnie efektywna dla grafów gęstych, gdzie liczba krawędzi jest bliska maksymalnej liczbie krawędzi ($O(V^2)$).

1.5 Łatwe zarządzanie wagami krawędzi:

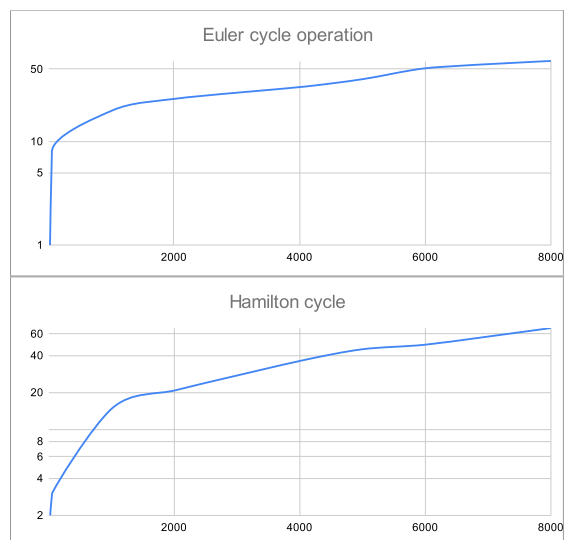
Dla grafów ważonych, wartości w macierzy mogą bezpośrednio reprezentować wagi krawędzi, co upraszcza zarządzanie dodatkowymi informacjami.

2 Wykresy zależności: (dla grafów hamiltonowskich o nasyceniu 30) $t = f(n)$

2.1 Skala liniowa t(ms):

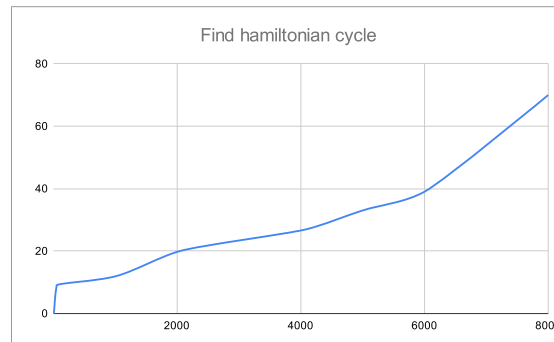


2.2 Skala logarytmiczna $t(\text{ms})$:

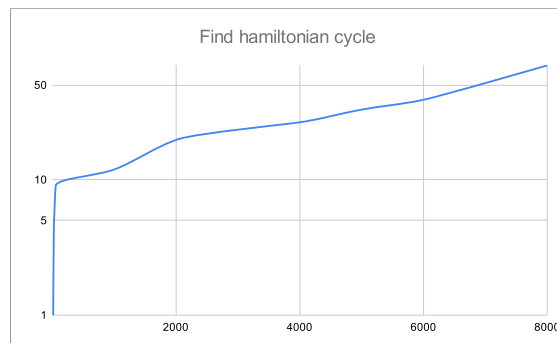


3 Wykresy zależności: (dla grafów nie hamiltonowskich o nasyceniu 50) $t = f(n)$

3.1 Skala liniowa t(ms):



3.2 Skala logarytmiczna $t(\text{ms})$:



4 Podsumowanie:

Nauczyłem się

1. Generować grafy.
2. Implementować algorytmy które operują na tych strukturach.
3. O grafach jako strukturach danych.
4. Jak zaimplementować algorytmy BFS i DFS.
5. Jak znajdować krawędzie w grafach.
6. Wypisywania na ekran.
7. Implementować algorytmy grafowe dla różnych reprezentacji maszynowych grafów.

Spis treści

1	Uzasadnienie wyboru reprezentacji grafu (macierz sąsiedztwa):	2
1.1	Prostota i przejrzystość:	2
1.2	Szybki dostęp do krawędzi:	2
1.3	Łatwość implementacji algorytmów:	2
1.4	Efektywność dla grafów gęstych:	2
1.5	Łatwe zarządzanie wagami krawędzi:	2
2	Wykresy zależności: (dla grafów hamiltonowskich o nasyceniu 30) $t = f(n)$	3
2.1	Skala liniowa $t(\text{ms})$:	3
2.2	Skala logarytmiczna $t(\text{ms})$:	4
3	Wykresy zależności: (dla grafów nie hamiltonowskich o nasyceniu 50) $t = f(n)$	5
3.1	Skala liniowa $t(\text{ms})$:	5
3.2	Skala logarytmiczna $t(\text{ms})$:	6
4	Podsumowanie:	6