|  |  |  |
| --- | --- | --- |
| Prowadzący:  Dr inż. Łukasz Jeleń | Projektowanie Algorytmów i Metody  Sztucznej Inteligencji | Termin zajęć:  Śr 7:30 |
| Wojciech Gołębiowski  241477 | **„Grafy”** | Data oddania sprawozdania:  22.05.2019 r. |

1. **Wprowadzenie**

Program ma za zadanie porównać ze sobą czasy budowania minimalnego drzewa rozpinającego grafów o różnych gęstościach i ilościach wierzchołków. Grafy będą zawierały losowe wagi krawędzi i wierzchołków, pełne i o gęstościach 75%, 50% i 25%. Grafy są reprezentowane przez macierz sąsiedztwa i listę sąsiedztwa.

1. **Opis badanego algorytmu**

Algorytm Kruskala  
Buduje minimalne drzewo rozpinające z zastosowaniem klastrów grupując wierzchołki. Początkowo wszystkie wierzchołki należą do osobnych klastrów, krawędzie są przechowywane w kolejce priorytetowe w której wagi są kluczami. Dla wszystkich krawędzi usuwa minimum z kolejki priorytetowej i jeżeli początek i koniec wierzchołka nie należą do tego samego klastra to dodajemy je do drzewa rozpinającego i łączymy klastry zawierające początkowy i końcowy wierzchołek w jeden. Złożoność tego algorytmu to O(m log n) gdzie m to ilość krawędzi, a n to ilość wierzchołków.

1. **Omówienie przebiegu eksperymentów**

W poniższych tabelach zaprezentowano czasy w jakich algorytm w danym przypadku się wykonał

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MACIERZ SĄSIEDZTWA | | | | |
| Gęstość grafu | 100 | 75 | 50 | 25 |
| |V| | t[s] | | | |
| 5 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0,001 | 0 | 0 |
| 50 | 0,06 | 0,035 | 0,016 | 0,006 |
| 100 | 0,95 | 0,537 | 0,249 | 0,06 |
| 500 | 1130,883 | 577,089 | 263,134 | 41,833 |
| 1000 | 15894,81 | 10882,34 | 4591,296 | 853,119 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| LISTA SĄSIEDZTWA | | | | |
| Gęstość grafu | 100 | 75 | 50 | 25 |
| |V| | t[s] | | | |
| 5 | 0,007 | 0 | 0,001 | 0 |
| 10 | 0,008 | 0 | 0 | 0 |
| 50 | 0,069 | 0,04 | 0,021 | 0,004 |
| 100 | 0,993 | 0,548 | 0,238 | 0,06 |
| 500 | 1042,003 | 589,171 | 294,385 | 53,416 |
| 1000 | 14266,17 | 10613,49 | 4147,903 | 1096,021 |

Na poniższych wykresach zaprezentowano w jakim czasie algorytm wykonał się w zależności od przypadku i implementacji grafu.

Na poniższych wykresach pokazano zależności czasu od gęstości grafów i ich implementacji.

1. **Wnioski**
   1. Z wykresów widać, że implementacja nie ma bardzo dużego znaczenia przy wyznaczaniu minimalnego drzewa rozpinającego, choć w większości przypadków reprezentacja przez macierz sąsiedztwa jest szybsza.
   2. Największe różnice w czasach wykonywania algorytmu występują przy zwiększaniu gęstości grafu, ponieważ wtedy algorytm musi przejść przez większą ilość krawędzi „niepotrzebnych”.
2. **Literatura**

* **Data Structures and Algorithms in C++, Michael T. Goodrich, 2004**