

# Implementacja algorytmu wyszukującego najkrótszą ścieżkę w grafie - $A^*$ . Porównanie z DFS oraz BFS.

Monika Litwin 200586

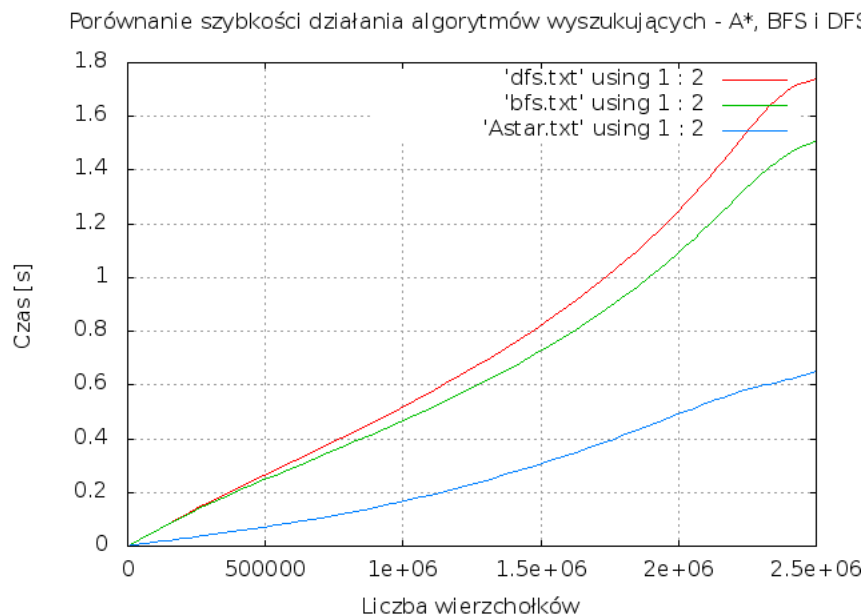
26 maja 2014

## Algorytm A\*

Jest to algorytm znajdowania najkrótszej ścieżki w grafie, w którym krawędzie posiadają wagi. Z dowolnego wierzchołka do wierzchołka przeznaczenia. Z dostępnych krawędzi, prowadzących do celu wybierana jest ta, o najmniejszej wadze. Sprawdzane jest też, czy obrana trasa jest poprawna (pomimo tego, że początkowa waga była najmniejsza, kolejne mogą być już dużo dłuższe od alternatywnych). Wówczas algorytm wybiera inną trasę. Tym sposobem A\* znajduje zawsze najkrótszą drogę. Dzieje się to szybciej niż w przypadku innych podobnych algorytmów, np. Branch & Bound, dzięki wykorzystaniu heurystyki. Jest to funkcja wyliczająca spodziewaną najkrótszą odległość między zadanymi wierzchołkami.

## Wyniki testów programu

W programie wywoływane były kolejno algorytmy wyszukiwujące - A\*, BFS i DFS z tymi samymi parametrami (początkiem i celem). Mierzony był czas ich wykonania - do odnalezienia zadanego wierzchołka. Działo się tak dla grafów o różnej wielkości. Od 50 do 2.500.000 elementów. Poniżej zamieszczam wykres oraz wyniki pomiarów.



Ilość wierzchołków	Czas - A*	Czas - BFS	Czas - DFS
50	0.00005	0.00002	0.00002
100	0.00034	0.00004	0.00003
1.000	0.0016	0.0002	0.0002
5.000	0.0021	0.0007	0.0003
10.000	0.003	0.001	0.001
50.000	0.001	0.002	0.002
100.000	0.04	0.05	0.08
500.000	0.08	0.56	0.42
1.000.000	0.04	0.3	0.49
1.500.000	0.2	0.35	0.42
2.000.000	0.4	1.11	1.14
2.100.000	0.7	1.12	1.37
2.200.000	0.52	1.32	1.44
2.300.000	0.6	1.42	1.69
2.400.000	0.62	1.47	1.72
2.500.000	0.65	1.51	1.74

Analizując wyniki pomiarów możemy zobaczyć, że dla mniejszych grafów lepsze czasy mają algorytmy DFS i BFS. Dla grafów liczących 50.000 wierzchołków i więcej ta tendencja się zmienia. Przy największych grafach - ponad 1.000.000 wierzchołków, A\* jest zauważalnie bardziej wydajny, znacznie mniej wydajny jest BFS, a najmniej DFS.

### Wnioski

Wyniki analizy są zgodne z oczekiwaniami. Dla mniejszych grafów wydajniejsze są mniej skomplikowane algorytmy DFS i BFS. Przy większych grafach widoczna jest już przewaga algorytmu A\*, który dzięki swoim dodatkowym funkcjom jest w stanie odrzucić ścieżki, które prawdopodobnie będą mało wydajne. BFS i DFS nie przeprowadzają takich szacowań, tylko przeszukują w określonej kolejności wszystkie wierzchołki, aż do napotkania zadanego.