

Algorytmy grafowe – najkrótsza ścieżka w grafie

Zadanie 1

- Zaimplementuj wybrany algorytm poszukiwania najkrótszej ścieżki:
 - Alg. Dijkstry – 3 pkt.
 - Alg. Floyda-Warshalla – 4 pkt.
 - Alg. Bellmana-Forda – 4 pkt.
 - Alg. A* – 5 pkt.
 - Alg. Johnsona – 6 pkt.
- Wklej kod źródłowy z komentarzami opisującymi realizowane kroki
- Algorytm ma zwracać długość minimalnej ścieżki oraz listę wierzchołków przez które przechodzimy

Zadanie 2

- Jakie własności grafu mogą być istotne, z punktu widzenia działania algorytmu
- Zdefiniuj dane oraz zaprezentuj uzyskane wyniki dla powyższych przypadków
- Przyjmij liczbę wierzchołków grafu $n=10$

Zadanie 3

Określ złożoność obliczeniową algorytmu

- Czasową / pamięciową
- Optymistyczną / pesymistyczną / średnią

Algorytm

Algorytm Dijkstry.

```
G: Graf
V: zbiór wierzchołków G
a: funkcja wag krawędzi a(u,v)
s: wierzchołek startowy - opcjonalnie k: wierzchołek końcowy
Q: zbiór wierzchołków, bez wyznaczonej min ścieżki
p[u]: poprzednik wierzchołka u
d[u]: aktualny koszt osiągnięcia u (z wierzchołka s przez p[u])
N[u]: lista sąsiedztwa wierzchołka u
u, u*: bieżący i ostatnio wybrany wierzchołek

Dijkstra(G,a,s):                                     //Dijkstra(G,a,s,k)

01  dla każdego u ∈ V
02      d[u] ← ∞
03      p[u] ← 0
04  Q ← V
05  Q ← Q-{s}
06  d[s] ← 0
07  u* ← s
08  dopóki Q ≠ ∅                                     //u* ≠ k
09      dla każdego (u ∈ Q i u ∈ N[u*])
10          jeśli d[u*]+a(u*,u)<d[u] to d[u] ← d[u*]+a(u*,u); p[u] ← u*
11      dla każdego u ∈ Q
12          u* ← arg min (d[u])
13      Q ← Q-{u*}
14  zwróć (dla każdego u ∈ V: d[u])                 //zwróć (d[k])
```

Sprawozdanie

Na platformie UPEL (Sprawozdanie 4) w terminie ćwiczeń (Sprawozdanie 4 wstępne) - pliki edycji doc / pdf / txt - kod, zrzuty ekranów, pliki wynikowe.

Zadania domowe

Przygotowanie do kolejnych - ćwiczeń 4 – znajomość algorytmów poszukiwania najkrótszej ścieżki w grafu .