

Algorytmy grafowe – minimalne drzewo rozpinające grafu

Zadanie 1

- Zaimplementuj algorytm Dijkstry-Prima poszukiwania minimalnego drzewa rozpinającego grafu (MST).
- Wklej kod źródłowy z komentarzami opisującymi realizowane kroki

Zadanie 2

- Jakie własności grafu mogą być istotne, z punktu widzenia działania algorytmu
- Zdefiniuj dane oraz zaprezentuj uzyskane wyniki dla powyższych przypadków

Zadanie 3

- Opisz krótko (w dwóch zdaniach) ideę działania algorytmu Kruskala
- Który krok algorytmu ma największą złożoność obliczeniową

Zadanie 4

Podaj interpretację praktyczną problemu MST

- Jakie własności problemu rzeczywistego są wyrażane za pomocą wag grafu
- Jakie własności problemu należy jeszcze uwzględnić aby model w pełni odzwierciedlał problem rzeczywisty
- Czy możliwa jest modyfikacja alg. Dijkstry-Prima (podać sposób / lub przyczynę braku możliwości) aby rozwiązać nim omawiany problem rzeczywisty

Algorytm

DPA (Dijkstra-Prima Algorytm).

```
G: Graf
V: zbiór wierzchołków G
a: funkcja wag krawędzi a[u,v]
s: wierzchołek startowy
Q: zbiór wierzchołków nienależących do MST
A: zbiór krawędzi MST
alfa[u]: poprzednik wierzchołka u w MST
beta[u]: waga krawędzi łączącej u z MST (z wierzchołkiem alfa[u])
N[u]: lista sąsiedztwa wierzchołka u
```

u, u^* : bieżący i ostatnio wybrany wierzchołek
suma: sumaryczna waga krawędzi MST

DPA(G, a, s)

```
01 suma ← 0
02 A ← ∅
03 dla każdego  $u \in V$ 
04     alfa[u] ← 0
05     beta[u] ← ∞
06 Q ← V
07 beta[s] ← 0
08 Q ← Q - {s}
09  $u^* \leftarrow s$ 
10 dopóki Q ≠ ∅
11     dla każdego ( $u \in Q$  i  $u \in N[u^*]$ )
12         jeśli  $a[u, u^*] < \beta[u]$  to alfa[u] ←  $u^*$ ; beta[u] ←  $a[u, u^*]$ 
13     dla każdego  $u \in Q$ 
14          $u^* \leftarrow \arg \min(\beta[u])$ 
15     Q ← Q - { $u^*$ }
16     A ← A + {alfa[ $u^*$ ],  $u^*$ }
17     suma ← suma +  $a[\text{alfa}[u^*], u^*]$ 
18 zwróć (A, suma)
```

Sprawozdanie

Na platformie UPEL (Sprawozdanie 3) w terminie ćwiczeń (Sprawozdanie 3 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 .