Dijkstra Damian Kamiński 04.05.2021

initDistance()

Metoda wykonuje operacje inicjalizacji. Operacja przypisania ∞ zostanie wykonana dla każdego wierzchołka. Na samym końcu przypisujemy wierzchołkowi startowemu odległość 0 co w sumie da nam złożoność O(|[1]V|).

relax()

Metoda wykonuje operacje relaksacji. Złożoność czasowa uzależniona jest od złożoności dodania elementu do kolejki.Stąd złożoność metody wynosi O(log[2]n).

Dijkstra()

Metoda wykonuje algorytm Dijkstry, szuka najkrótszej drogi pomiedzy wierzchołkiem startowym a pozostałymi.

Na samym początku wykonuje się initDistance() co zajmie |V| operacji oraz O(1) operacji na dodanie wierzchołka startowego do kolejki. Kolejny etap to zagnieżdżone pętle while. Wewnętrzna pętla wykona się [3]|E_{adj}| razy, odwiedzi wierzchołki z którymi jest połączony wierzchołek ściągnięty z kolejki. Wewnątrz tej pętli znajduje się metoda relax(), która jak wiemy wykona się w czasie O(logn). Zewnętrzna pętla wykona się |E| razy, przejdzie po wszytkich krawędziech. W tej pętli prócz wenętrznej pętli jest jeszcze metoda pop(), która ściąga element z kolejki I naprawia strukturę kopca. Wykona się ona w czasie O(logn). A więc:

$$|V|+O(1)+|E|\cdot\left(O\left(\log_2 n\right)+|E_{adj}|\cdot O\left(\log_2 n\right)\right)$$

$$|V|+|E|\cdot\left(O\left(\log_2 n\right)+|E_{adj}|\cdot O\left(\log_2 n\right)\right)$$

W kolejce w jednym momencie może być maksymalnie |E| elementów, a więc

$$|V|+|E|\cdot|E_{adj}|\cdot O\left(\log_2|E|\right)$$

$$O\left(|V|+|E|\log_2|E|\right)$$

A więc całkowita złożoność czasowa metody wyniesie O(|V| + |E|$\log_2$|E|). Złożoność pamięciowa wyniesie O(|E|), bo potrzeba nam miejsca na kolejke prioretytową, a ta w możliwym momencie szczytowym będzie przetrzymywać |E| elementów.

1 Ilość wierzchołków w grafie
2 Ilość elementów w kolejce
3 Ilość krawędzi między wierzchołkami