

# Algorytmy lista 3

Piotr Puszczynski

December 15, 2021

## 1 dijkstra

Algorytm o złożoności  $O(n^2)$ . Zaimplementowany przy pomocy kolejki priorytetowej. Kroki algorytmu:

1. ustawienie wszystkich elementów w tablicy kosztów na nieskończoność
2. ustawienie w tablicy kosztów 0 w indeksie wierzchołka startowego
3. wrzucenie tego indeksu z kosztem do kolejki
4. pętla trwająca dopóki kolejka jest niepusta
5. w pętli sprawdza się wszystkich sąsiadów
6. dla każdego sąsiada sprawdza się czy nowy koszt dojścia do tego wierzchołka jest mniejszy niż aktualny
7. jeżeli nie przechodzimy do następnej iteracji / jeżeli tak aktualizujemy tablicę kosztów i dodajemy wierzchołek z jego kosztem do kolejki

## 2 dial

Algorytm o złożoności  $O(n * w + m)$ . Zaimplementowany przy pomocy tablicy struktur Bucket. Kroki algorytmu:

1. stworzenie tablicy bucketow o długości  $v * w + 1$   $v$  - liczba wierzchołków,  $w$  - największy koszt krawędzi
2. ustawienie wszystkich elementów w tablicy kosztów na nieskończoność
3. ustawienie w tablicy kosztów 0 w indeksie wierzchołka startowego
4. dodanie do bucketa o indeksie 0 indeks wierzchołka startowego
5. zaczynamy pętlę po wszystkich bucketach
6. wyciągamy po kolei wszystkie elementy z bucketa
7. po wszystkich sąsiadach sprawdza się czy nowy koszt jest mniejszy od poprzedniego
8. jeżeli tak aktualizujemy tablicę kosztów i dodajemy indeks do bucketa o indeksie równym kosztu dojścia do tego wierzchołka / jeżeli nie przechodzimy do kolejnego elementu w buckecie

### 3 radix heap

Algorytm o złożoności  $O(n \log(w) + m)$ . Zaimplementowany przy pomocy tablicy struktur Bucket. W tym algorytmie buckety mają niejawnie zdefiniowane przedziały kosztów rosnące wykładniczo (np. B1 : 0, B2 : 1, B3 : 2-3, B4 : 4-7, B5 : 8-15, B6 : 16-31, ...). Można zdefiniować sobie metodę obliczającą odpowiedni indeks dla podanego inta. Jest to zmodyfikowany algorytm działa. różnica polega na tym, że idąc po bucketach wyciągamy z niego wierzchołki tylko wtedy gdy przedział kosztów bucketa jest równy jeden, lub w bucketcie jest jeden element. W przypadku gdy bucket jest pusty, pomijamy go, a w przypadku kiedy nie spełnia pierwszego warunku i liczba elementów jest większa od 1 algorytm przenosi wierzchołki do bucketów od 0 do aktualnego bucketa - 1. Do przestawienia wierzchołków potrzeba obliczyć minimalny koszt dla indeksów z aktualnego bucketa i przy wstawianiu do odpowiedniego bucketa używamy zaimplementowanej wcześniej metody obliczającej odpowiedni indeks podając (int - min). Po przechodzeniu na początek pętli, czyli do bucketa o indeksie 0 i powtarzamy procedurę.