

АЛГОРИТМ ДЕЙКСТРИ ПОШУКУ НАЙКОРОТШОГО ШЛЯХУ ГРАФА. ПРОГРАМНА РЕАЛІЗАЦІЯ

Науковий керівник: к.ф.-м.н., старший викладач Наталія СТЕФУРАК

Найефективніший алгоритм віднаходження довжини найкоротшого шляху від фіксованої вершини до будь-якої іншої є алгоритм, який запропонував у 1959 р. датський математик Е. Дейкстра.

Алгоритм Дейкстри може бути застосований для розв'язання багатьох прикладних задач, зокрема, знаходження найкоротшого шляху для GPS-навігації, оптимізації логістичних маршрутів доставки, знаходження найшвидшого шляху для передачі даних в мережах тощо.

Алгоритм працює із зваженими орієнтованими графами, які можуть мати лише додатну вагу кожного ребра. Нехай $G = (V, E)$ – орієнтований граф, $w(v_i, v_j)$ – вага дуги. Починаємо з вершин a і знаходимо віддаль від a до кожної із суміжних до неї вершин. Вибираємо вершину, віддаль від якої до вершини a є найменшою; нехай ця вершина буде v^* . Далі знаходимо віддалі від вершини a до кожної вершини, суміжної з v^* вздовж шляху, який проходить через вершину v^* . Якщо для певної з таких вершин ця віддаль менша від поточної, то заміняємо нею поточну віддаль. Знову вибираємо вершину, найближчу до a , але таку, що не вибирали раніше, і процес повторюємо.

При реалізації даного алгоритму зручно користуватись методом міток, які будуть двох типів: тимчасові та постійні. Вершини з постійними мітками групують у множину M , яку називають множиною позначених вершин. Решта вершин має тимчасові мітки, і множину таких вершин позначають через T ($T = V \setminus M$), де V – множина усіх вершин графа. Вихідній вершині присвоюється мітка 0 і мітки ∞ для усіх інших вершин. Суть алгоритму зводиться у покроковому переприсвоюванні значень міток. Величина постійної мітки дорівнює довжині найкоротшого шляху від вихідної вершини до заданої. Якщо ж мітка тимчасова, то вона дорівнює довжині найкоротшого шляху, який проходить лише через вершини з постійними мітками. Процес продовжується до тих пір поки усі вершини не матимуть постійних міток.

Даний алгоритм реалізовано на мовах C++ та Python. Основна ідея реалізації полягає у використанні пріоритетної черги для зберігання вершин, які потрібно опрацювати, та використання структури даних, яка дозволяє зберігати граф.

Одна з головних відмінностей між програмними реалізаціями алгоритму Дейкстри на мові C++ та Python полягає у тому, як кожна з цих мов реалізує деякі важливі функції, такі як структури даних та робота з пам'яттю.

На мові C++, для реалізації алгоритму Дейкстри, часто використовують стандартну бібліотеку, яка містить багато різних структур даних, таких як вектори та пріоритетні черги. Це дозволяє користувачу реалізувати алгоритм Дейкстри більш ефективно, економно використовуючи пам'ять.

У мові Python також є багато структур даних та функцій, які допомагають реалізувати алгоритм Дейкстри. Однак, Python має меншу продуктивність в порівнянні з C++, тому реалізація алгоритму може працювати повільніше на великих вхідних

даних, проте необхідно відмітити можливості цієї мови у візуальному представленні графа та виведені результатів.

Крім того, Python має більш простий та зрозумілий синтаксис, що робить його більш доступним для користувача. З іншого боку, C++ може бути більш потужним та швидким, і це робить його популярним для вимогливих застосувань, таких як наукові обчислення та розробка ігор.

Список використаних джерел:

1. Нікольський Ю.В. Дискретна математика / Ю.В.Нікольський, В.В.Пасічник, Ю.М.Щербина – К.: Видавнича група ВНУ, 2007, 387с.
2. М. Ф. Бондаренко Комп'ютерна дискретна математика: підручник / Бондаренко М. Ф., Білоус Н. В., Руткас А. Г. –Харків, «Компанія СМІТ», 2011- 485с.
3. Кривий С.Л. Дискретна математика: підручник для студентів вищ. навч. закл.– Чернівці-Київ: Видавничий дім «Букрек», 2014. – 568 с.