**УДК 510.52**

*Кофель Д.В., бакалавр*

*Левчук В.М., асистент*

*Полтавський національний технічний університет*

*імені Юрія Кондратюка*

**ЗНАХОДЖЕННЯ НАЙКОРОТШИХ ВІДСТАНЕЙ І ШЛЯХІВ У МЕРЕЖІ**

**Вступ**

Завдяки своєму широкому застосуванню, теорія про знаходження найкоротших шляхів останнім часом інтенсивно розвивається. Знаходження найкоротшого шляху - життєво необхідне і використовується практично скрізь, починаючи від знаходження оптимального маршруту між двома об'єктами на місцевості (наприклад, найкоротший шлях від дому до університету), в системах автопілота, для знаходження оптимального маршруту при перевезеннях, комутації інформаційного пакету в Internet і т.д. Пошук найкоротшого шляху розглядається за допомогою певного математичного об'єкту, що називається графом.

**Аналіз останніх публікацій.** Проблемі пошуку алгоритмів найкоротших шляхів та розвитку комп’ютерних технологій та їх вплив на розвиток професійних компетентностей випускників ВНЗ присвячено значну кількість навчально-методичних робіт вітчизняних авторів: В.В. Анісімов, О.В. Вітюк, А.Г. Григорович, Р.С. Гуревич, О.М. Джеджула, Ю.О. Дорошенко,

І .М. Лазурчак, В.В. Моштук, І .Д. Нищак, І .О. Петрицин, М.Ф. Юсупова та ін.

**Мета статті.** Дослідження і наведення рекомендацій реалізації алгоритмів пошуку найкоротших шляхів та їх практичні відомості застосування в практичній діяльності, та формування вмінь і навичок для їх використання

**Виклад основного матеріалу.** Теорія графів - це область дискретної математики, особливістю якої є геометричний підхід до вивчення об'єктів. Теорія графів знаходиться зараз у самому розквіті. Зазвичай її відносять до топології (тому що в багатьох випадках розглядаються лише топологічні властивості графів), проте вона перетинається з багатьма розділами теорії множин, комбінаторної математики, алгебри, геометрії, теорії матриць, теорії ігор, математичної логіки та багатьох інших математичних дисциплін. Основний об'єкт теорії графів - граф і його узагальнення. Задача про найкоротший шлях полягає в знаходженні найкоротшого шляху від заданої початкової вершини A до заданої вершини Z.

В процесі написання роботи розглядаються **задачі:**

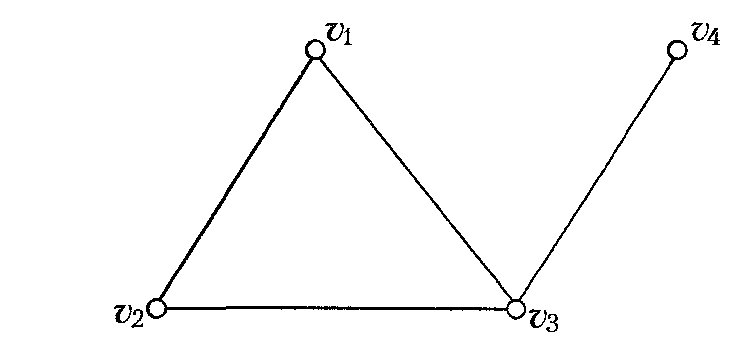
1. Для заданої початкової вершини A знайти найкоротші шляхи від A до всіх інших вершин графа.

2. Знайти найкоротші шляхи між усіма парами вершин графа.

*Простим графом* називають пару G=(V, Е), де V - непорожня скінченна множина елементів, називаних *вершинами,* Е - множина невпорядкованих пар різних елементів із V. Елементи множини Е (невпорядковані пари різних вершин) називають *ребрами*.

На рис. 1 зображено простий граф G з множиною вершин V= 

і множиною ребер E= {{ }, {}, { }, { }}



**Рис.1**

Іноді розглядають графи, у яких дві вершини можуть бути з'єднані більше ніж одним ребром. Так виникає поняття мультиграфа. *Мультиграфом* називають пару (V, Е), де V - скінченна непорожня множина вершин, а Е - сім'я невпорядкованих пар різних елементів множини V. Тут застосовано термін „сім'я" замість поняття „множина", бо елементи в Е (ребра) можуть повторюватись. Ребра, що з'єднують одну й ту саму пару вершин, називають *кратними* (або *паралельними*).

Окрім кратних ребер розглядають також петлі, тобто ребра, які з'єднують вершину саму із собою. *Псевдографом* називають пару (V, Е), де V -скінченна непорожня множина вершин, а Е -сім'я невпорядкованих пар не обов'язково різних вершин.

Розглянуті три типи графів називають неоріентованими. Псевдограф -це найзагальніший тип неорієнтованого графа, бо він може містити петлі й кратні ребра. Мультиграф -це неорієнтований граф, який може містити кратні ребра, але не може містити петель. Нарешті, простий граф - це неорієнтований граф без кратних ребер і без петель.

Розглядають також орієнтовані графи. Орієнтованим графом називають пару (V, Е), де V - скінченна непорожня множина вершин, а Е - множина впорядкованих пар елементів множини V. Елементи множини Е в орієнтованому графі називають дугами (чи орієнтованими ребрами).

Останнім часом графи і пов’язані з ними методи досліджень використовуються практично в усіх розділах сучасної математики і, зокрема, дискретної математики. Граф є математичною моделлю найрізноманітніших об’єктів, явищ і процесів, що досліджуються і використовуються в науці, техніці та на практиці.

У реальних задачах на графах часто потрібно брати до уваги додаткову

інформацію - фактичну віддаль між окремими пунктами, вартість проїзду, час проїзду тощо. Для цього використовують поняття зваженого графа.

*Зваженим* називають простий граф, кожному ребру є якого приписано дійсне число ***w(e)***. Це число називають вагою ребра e. Аналогічно означають *зважений орієнтований граф*: це такий орієнтований граф, кожній дузі ***e*** якого приписано дійсне число ***w(e)*,** називане вагою дуги.

*Довжиною шляху* в зваженому графі називають суму ваг ребер (дуг), які утворюють цей шлях. Якщо граф не зважений, то вагу кожного ребра (кожної дуги) уважають рівною 1 й отримують раніше введене поняття довжини шляху як кількості ребер (дуг) у ньому.

Існують три найбільш ефективних алгоритми знаходження найкоротшого шляху:

* алгоритм Дейкстри (використовується для знаходження оптимального маршруту між двома вершинами);
* алгоритм Форда-Бельманн**а** (для знаходження оптимального маршруту між усіма парами вершин);
* алгоритм Флойда-Воршелла (для знаходження найкоротшого шляху між парами вершин).

**Метою даного дослідження** єпрограмна реалізація алгоритму пошуку найкоротшого шляху між двома будь-якими вершинами графа. Програма повинна працювати так, щоб користувач вводив кількість вершин і довжини ребер графа, а після обробки цих даних на екран виводився найкоротший шлях між двома заданими вершинами. Необхідно передбачити різні результати пошуку, щоб програма не видавала помилок і працювала правильно. Дана програма може використовуватися в дискретній математиці для дослідження графів або в якості наочного посібника, що демонструє застосування алгоритмів на практиці.

**Алгоритм Дейкстри** - найефективніший алгоритм на графах відкритий нідерландським вченим Е. Дейкстром у 1959 році. Цей алгоритм знаходить найкоротшу відстань від однієї вершини графу до всіх решти. Алгоритм працює лише для графів у яких ребра не мають від’ємної ваги. Алгоритм широко застосовується в програмуванні і технологіях, наприклад, його використовують протоколи маршрутизації OSPF та IS-IS.

Алгоритм Дейкстри будує найкоротші шляхи, що ведуть з вихідної вершини графа до решти вершин цього графа (якщо такі є). У процесі роботи алгоритму послідовно позначаються розглянуті вершини графа. Причому вершина, позначена останньою (на даний момент) розташована ближче до вихідної вершині, ніж всі непомічені, але далі, ніж всі помічені. Спочатку позначається вихідна вершина; наступною, очевидно, буде позначена вершина, найближча до початкової, і суміжна з нею. Нехай на якомусь кроці вже позначено кілька вершин. Відомі найкоротші шляхи, що ведуть з вихідної вершини до позначених. Для кожної з непозначених вершин проробимо наступне:

1. Розглянемо всі дуги, які ведуть з позначених вершин в одну непозначену. Кожна така дуга є останньою дугою на шляху з вихідної вершини в цю непозанчену.

2. Виберемо з цих шляхів найкоротший. А потім виберемо серед них самий короткий до всіх непозначених вершин, і позначимо вершину, до якої він веде.

Алгоритм завершиться, коли будуть позначені всі досяжні вершини.

В результаті роботи алгоритму Дейкстри будується дерево найкоротших шляхів.

**Алгоритм Форда-Бельманна** - алгоритм пошуку найкоротшого шляху в зваженому графі. Знаходить найкоротші шляхи від однієї вершини графа до всіх інших. На відміну від алгоритму Дейкстри, алгоритм Беллмана—Форда допускає ребра з негативною вагою. Запропоновано незалежно Річардом Беллманом і Лестером Фордом.

Дано [орієнтований](http://uk.wikipedia.org/wiki/%D0%9E%D1%80%D1%96%D1%94%D0%BD%D1%82%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B9_%D0%B3%D1%80%D0%B0%D1%84" \o "Орієнтований граф) або [неорієнтований](http://uk.wikipedia.org/wiki/%D0%9D%D0%B5%D0%BE%D1%80%D1%96%D1%94%D0%BD%D1%82%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B9_%D0%B3%D1%80%D0%B0%D1%84" \o "Неорієнтований граф) граф  G(V, E) з ваговою функцією w:E\to \mathbb{R}. Вагою w(p) шляху p=\langle v_0, v_1, \dots, v_k \rangle назвемо суму ваг ребер, що входять в цей шлях: w(p)=\sum_{i=1}^k w(v_{i-1}, v_i).

Вхідними даними для алгоритму є граф G, вагова функція w, та стартова вершина s. Потрібно знайти найкоротші шляхи від вершини s до всіх вершин графа. Алгоритм Беллмана-Форда повертає логічне значення, яке вказує на те, чи міститься в графі цикл з негативною вагою, досяжний з витоку. Якщо такий цикл існує у графі G,алгоритм повідомляє, що найкоротших шляхів не існує. Якщо ж таких циклів немає, алгоритм видає найкоротші шляхи і їх вагу.

Сам алгоритм Форда-Беллмана представляє з себе кілька фаз. На кожній фазі проглядаються всі ребра графа, і алгоритм намагається справити релаксацію (relax, ослаблення) уздовж кожного ребра (u,v) ваги w(u,v). Релаксація вздовж ребра - це спроба поліпшити значення v.d значенням v.u+w(u,v) . Фактично це означає, що ми намагаємося поліпшити значення для вершини v, користуючись ребром (u,v) і поточним значенням для вершини u . Стверджується, що достатньо |G.V|-1 фази алгоритму, щоб коректно порахувати довжини всіх найкоротших шляхів у графі (цикли негативної ваги відсутні). Для недосяжних вершин відстань v.d залишиться нескінченністю.

Метод використовується в деяких протоколах дистанційно-векторної маршрутизації, наприклад в RIP (Routing Information Protocol - Протокол маршрутної інформації).

**Алгоритм Флойда-Воршелла** - це алгоритм динамічного програмування для знаходження найкоротших відстаней між усіма вершинами зваженого орієнтованого графа. Розроблений в 1962 році Робертом Флойдом і Стівеном Воршеллом.

Нехай вершини графа G=(V,\; E),\; |V| = n пронумеровані від 1 до n і введено позначення d_{i j}^{k} для довжини найкоротшого шляху від i до j, який окрім самих вершин i,\; j проходить тільки через вершини 1 \ldots k. Очевидно, що d_{i j}^{0} - довжина (вага) ребра (i,\;j), якщо воно існує (в іншому разі його довжина може бути позначена як \infty)

Існує два варіанти значення d_{i j}^{k},\;k \in \mathbb (1,\;\ldots,\;n):

1. Найкоротший шлях між i,\;j не проходить через вершину k, тоді d_{i j}^{k}=d_{i j}^{k-1}
2. Існує більш короткий шлях між i,\;j, що проходить через k, тоді він спочатку йде від i до k, а потім від k до j. У цьому випадку, очевидно, d_{i j}^{k}=d_{i k}^{k-1} + d_{k j}^{k-1}

Таким чином, для знаходження значення функції досить вибрати мінімум з двох позначених значень.

Тоді [рекурентна](http://uk.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BA%D1%83%D1%80%D1%81%D1%96%D1%8F" \o "Рекурсія) формула для d_{i j}^k має вигляд:

d_{i j}^0 — довжина ребра (i,\;j)

d_{i j}^{k} = \min (d_{i j}^{k-1},\; d_{i k}^{k-1} + d_{k j}^{k-1})

Алгоритм Флойда-Воршелла послідовно обчислює всі значення d_{i j}^{k}, \forall i,\; j для k від 1 до n. Отримані значення d_{i j}^{n} є довжинами найкоротших шляхів між вершинами i,\; j.

Основним завданнямавтора єпрограмна реалізація алгоритму пошуку найкоротшого шляху між двома будь-якими вершинами графа. Програма повинна працювати так, щоб користувач вводив кількість вершин і довжини ребер графа, а після обробки цих даних на екран виводився найкоротший шлях між двома заданими вершинами. Необхідно передбачити різні результати пошуку, щоб програма не видавала помилок і працювала правильно. Дана програма може використовуватися в дискретній математиці для дослідження графів або в якості наочного посібника, що демонструє застосування алгоритмів на практиці.

Отже, алгоритми пошуку найкоротших шляхів широко застосовуються і розвиваються в наш час. Кожен з нас завжди шукає найкоротший шлях для вирішення певної задачі. Правильне використання алгоритмів дає змогу знаходити найкоротший шлях з найменшими затратами часу та ресурсів. В майбутньому планую створити програму яка може використовуватися у вивчені курсу дискретної математики, як наочний приклад роботи алгоритму.

**Література:**

*1. Ю.В. Нікольський. Дискретна математика. Київ. Видавнича група BHV. 2007 р.*

*2. М.Ф. Бондаренко, Н.В. Білоус. Комп’ютерна дискретна математика. Харків. «Компанія Сміт».2004р.*

*Кофель Д.В., бакалавр*

*Левчук В.М., асистент*

*Полтавський національний технічний університет*

*імені Юрія Кондратюка*

**ЗНАХОДЖЕННЯ НАЙКОРОТШИХ ВІДСТАНЕЙ І ШЛЯХІВ У МЕРЕЖІ**

*У статті описані алгоритми пошуку найкоротших шляхів у мережі, а саме:**алгоритм Дейкстри, алгоритм Форда-Бельманна, алгоритм Флойда-Воршелла.*

*Об’єктом дослідження є сукупність вершин і зв’язки між ними, які називають графами.*

***Ключові слова:*** *граф, алгоритм, найкоротший шлях, алгоритм Дейкстри, алгоритм Форда-Бельманна, алгоритм Флойда-Воршелла.*

*Кофель Д.В., бакалавр*

*Левчук В.М., асистент*

*Полтавский национальный университет*

*имени Юрия Кондратюка*

**НАХОЖДЕНИЕ КРАТЧАЙШИХ РАССТОЯНИЙ И ПУТЕЙ В СЕТИ**

*В статье описаны алгоритмы поиска кратчайших путей в сети, а именно: алгоритм Дейкстры, алгоритм Форда-Бельманна, алгоритм Флойда-Уоршелла.*

*Объектом исследования является совокупность вершин и связи между ними, которые называют графами.*

***Ключевые слова:*** *граф, алгоритм, кратчайший путь, алгоритм Дейкстры, алгоритм Форда-Бельманна, алгоритм Флойда-Уоршелла.*

*Kofel D.V., bachelor*

*Levchuk V.M., assistant*

*Poltava National Technical Yuri Kondratyuk University*

**BEING OF THE SHORTEST DISTANCES AND PATHS IS IN NETWORK**

*In the article the described search of short*  way*s are in a network, namely:* *Dijkstra's algorithm,*  *Bellman–Ford algorithm, Floyd–Warshall algorithm.*

*The object of the study is a set of tops and the connections between them , which is a graph .*

***Keywords****: graph, algorithm, short* way*s, Dijkstra's algorithm,*  *Bellman–Ford algorithm, Floyd–Warshall algorithm.*