Частное учреждение образования

«Колледж бизнеса и права»

|  |  |
| --- | --- |
|  | УТВЕРЖДАЮ  Ведущий методист колледжа  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Е.В. Паскал  « \_\_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 года |
| Специальность 2-40 01 01 «Программное обеспечение информационных технологий» | Учебная дисциплина «Основы алгоритмизации и программирование» |

**Лабораторная работа № 40**

**Инструкционно-технологическая карта**

Тема: Разработка постановки задачи приложения, реализующего алгоритм Форда-Фалкерсона.

Цель: Научиться разрабатывать постановку задачи приложения, реализующего алгоритм Форда-Фалкерсона; научиться создавать граф и алгоритм поиска максимального потока в сети в этом графе.

Время выполнения: 2 часа.

1. **Порядок выполнения работы**
2. Изучить теоретические сведения к лабораторной работе.
3. Разработать на языке С++ программу вывода на экран решения задачи в соответствии с вариантом индивидуального задания, указанным преподавателем.
4. Отлаженную, работающую программу сдать преподавателю. Работу программы показать с помощью самостоятельно разработанных тестов.
5. Ответить на контрольные вопросы.
6. **Теоретические сведения**

**Алгоритм Форда-Фалкерсона**

Этот алгоритм был впервые предложен в 1956 г. До того времени задача решалась с помощью методов линейного программирования, что было крайне неэффективно. Алгоритм является **псевдополиномиальным** и имеет оценку O(n\*m log U), где m = |E|, n = |V|, U = max(Cij). Алгоритм начинает свою работу с **нулевого потока** и на каждой своей итерации увеличивает поток в сети. На каждом шаге находится увеличивающая величину потока цепь. Поток увеличивается вдоль дуг этой цепи, пока она не станет **насыщенной**.

Увеличивающей цепью является цепь из **источника** в **сток**, все дуги которой допустимы. Дугу из вершины a в вершину b назовем **допустимой**, если выполняется одно из следующих условий:

1) f(e) < c(e) и дуга согласованна;

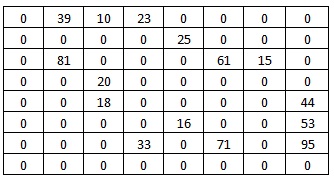
2) f(e) > 0 и дуга несогласованна.

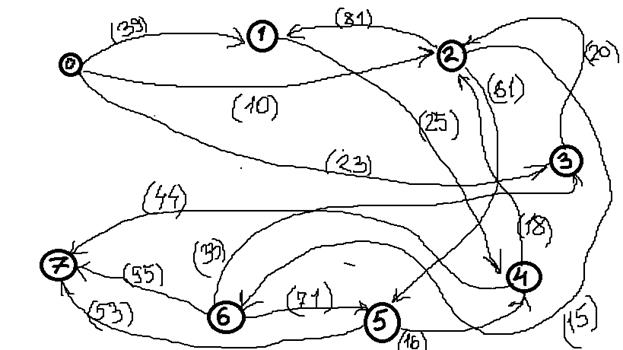
По увеличивающей цепи можно пустить поток величины Q, где Q = min{q(ei), 1 ≤ i ≤ l} и q(e) = {с(e) – f(e), если дуга согласованна, f(e), если дуга не согласованна}. Для того, чтобы увеличить величину потока сети на Q, необходимо увеличить на Q поток на каждой согласованной дуге цепи и уменьшить на каждой несогласованной. В своей работе Форд и Фалкерсон доказали, что поток в сети, для которой нельзя построить увеличивающую цепь, является максимальным. Для нахождения увеличивающей цепи ими был предложен “Метод расстановки пометок”. Процесс расстановки меток начинается в источнике сети и заканчивается в ее стоке. Как только сток оказался помеченным, мы можем говорить о существовании увеличивающей цепи из источника в сток. Метка, “наносимая” на вершины сети, содержит необходимый минимум информации, достаточный для того, чтобы восстановить эту цепь и определить величину, на которую можно изменить поток в ней. Вершина сети может находиться в одном из 3-х состояний: “непомеченная”, “помеченная” и “просмотренная”.

**Реализация алгоритма Форда-Фалкерсона**

Рассмотрим решение задачи нахождения максимального потока в транспортной сети с помощью алгоритма Форда-Фалкерсона и построим разрез сети S. Исходные данные: дана сеть S(X,U), х0 – исток сети; х7 – сток сети, где х0∈X; х7∈X. Значения пропускных способностей дуг ri,j заданы по направлению ориентации дуг: от индекса i к индексу j.

r[0,1] = 39; r[4,7] = 44; r[6,3] = 33; r[5,7] = 53; r[0,2] = 10; r[4,2] = 18; r[6,7] = 95; r[5,4] = 16; r[0,3] = 23; r[2,5] = 61; r[2,1] = 81; r[6,5] = 71; r[1,4] = 25; r[2,6] = 15; r[3,2] = 20.

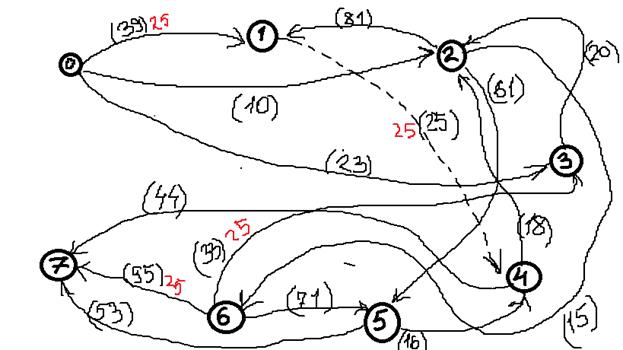




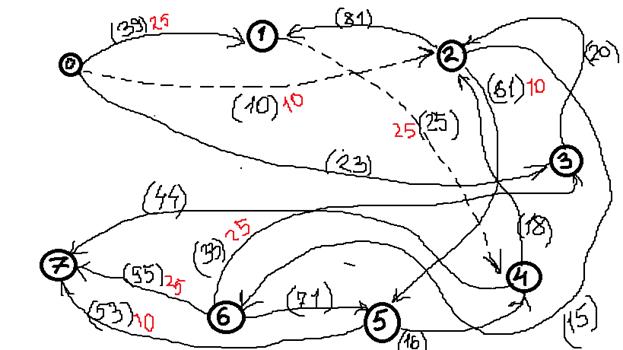
1. Зададим на сети нулевой поток (на всех дугах величина потока ϕij = 0). **Нулевой поток** – это начальный допустимый поток на сети. Значение потока на каждой дуге uij будем указывать за скобками пропускной способности дуги). Значение потока, равное нулю, не указываем.

2. Выбираем на сети (произвольно) путь, ведущий из вершины x0 в вершину x7: X0-X1-X4-X6-X7.

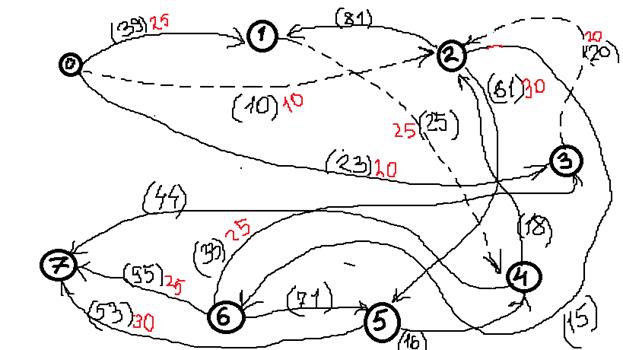
3. Находим min(39,25,33,95) = 25 и увеличиваем поток на эту величину. Ребро Х1-Х4 помечаем как рассмотренное.



4. Выбираем еще один путь, например, Х0-Х2-Х5-Х7, находим min(10,61,53) = 10 и увеличиваем поток на эту величину. Ребро Х0-Х2 помечаем как рассмотренное.



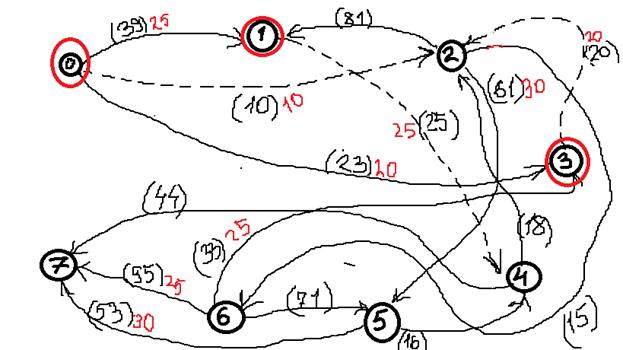
5. Выбираем еще один путь, например, Х0-Х3-Х2-Х5-Х7, находим min(23,20,61-10,53-10) = min(23,20,51,43) = 20 и увеличиваем поток на эту величину. Ребро Х3-Х2 помечаем как рассмотренное.



6. Более путей от Х0 до Х7 нет, суммируем увеличения потока: 25+10+20 = 55. Вывод: максимальный поток равен 55.

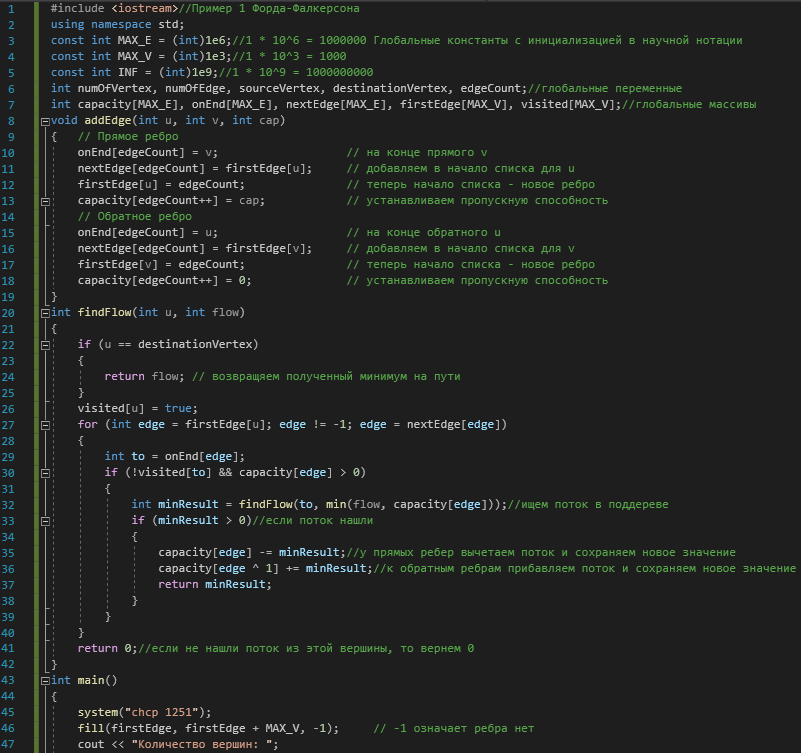
2) Построить разрез сети S. Процедура «пометок вершин». Начальное состояние: все вершины не имеют пометок.

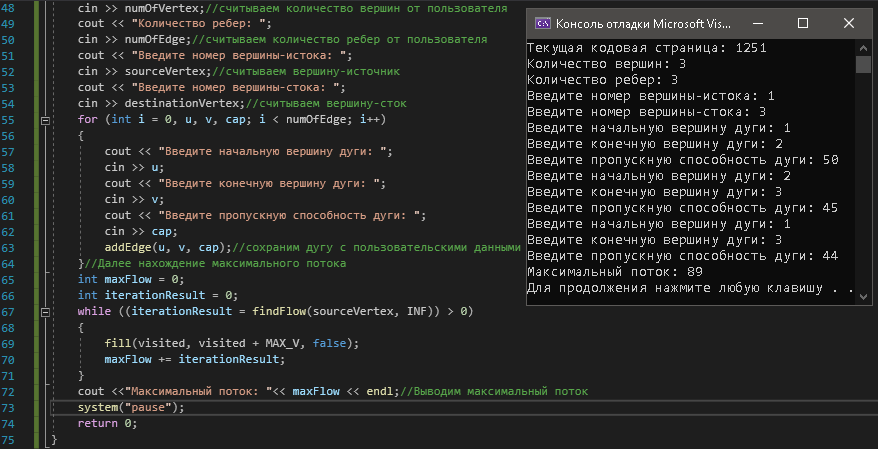
Вершине Х0 приписывается пометка. Всем вершинам xi∈{Гs}, для которых дуга (x0;xi) не насыщена, присваиваются пометки (красные круги).

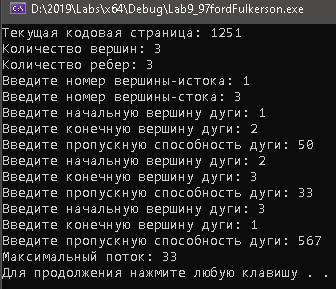


Определяем дуги минимального разреза: это дуги, начала которых находятся в помеченных вершинах, а концы – в непомеченных вершинах. Это дуги: u0,2, u1,4, u3,2. Таким образом, минимальный разрез данной сети T = (u0,2, u1,4, u3,2). Вычисление величины максимального потока Фmax = c0,2, c1,4, c3,2 = 10+25+20 = 55.

1. **Пример выполнения программы**







1. **Задания по вариантам**

Создать и отобразить (в письменном или электронном виде) свой взвешенный граф, состоящий из заданного количества вершин и дуг. Уметь проиллюстрировать работу алгоритма Форда-Фалкерсона по поиску максимального потока в сети на своем графе.

Граф должен быть проиллюстрирован в графическом виде и в виде матрицы, например, как показано ниже:

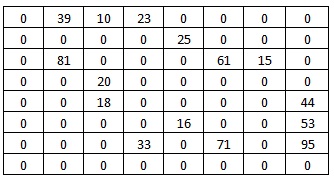


Рисунок 1 – Пример матрицы для хранения весов ребер графа. Названия строк и столбцов проименуйте соответствующими номерами вершин графа

Таблица 1 – Варианты индивидуальных заданий

|  |  |  |
| --- | --- | --- |
| Вариант № | Количество вершин | Количество ребер |
| 1 | 7 | 12 |
| 2 | 8 | 11 |
| 3 | 9 | 10 |
| 4 | 7 | 13 |
| 5 | 8 | 12 |
| 6 | 9 | 11 |
| 7 | 7 | 14 |
| 8 | 8 | 13 |
| 9 | 9 | 12 |
| 10 | 7 | 15 |
| 11 | 8 | 14 |
| 12 | 9 | 13 |
| 13 | 7 | 14 |
| 14 | 8 | 15 |
| 15 | 9 | 14 |

1. **Контрольные вопросы**
2. Объясните, что такое «исток в графе».
3. Объясните, что такое «сток в графе».
4. Какая дуга, согласно алгоритму Форда-Фалкерсона, считается «допустимой»?
5. Что такое «нулевой поток»?
6. Дайте определение понятию «максимальный поток в сети».
7. Зачем может понадобиться находить максимальный поток в сети графа?
8. Опишите алгоритм Форда-Фалкерсона. В чем его суть?

**Литература**

**Дейтел,** Х.М. Как программировать на С++ / Х.М. Дейтел, П.Дж. Дейтел . – М. : Бином-Пресс , 2018 . – 1456 с.

**Павловская**, Т.А. С++. Объектно-ориентированное программирование : практикум / Т.А. Павловская, Ю.А. Щупак . – СПб. : Питер , 2019 . – 265 с.

**Страуструп**, Б. Язык программирования С++ / Б. Страуструп . – СПб. : Бином-Пресс , 2019 . – 1054 с.

Преподаватель Шаляпин Ю.В.

|  |
| --- |
| Рассмотрено на заседании цикловой  комиссии ПОИТ № 10  Протокол №\_\_\_\_от «\_\_\_»\_\_\_\_\_\_\_\_\_\_2021 г.  Председатель ЦК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |