О.П. РЯБЫЧИНА1, Ю.А.ГИСИЧ2

**ПРОГРАММНАЯ РЕАЛИЗАЦИЯ АЛГОРИТМА ДЕЙКСТРЫ**

*1 Учреждение образования «Белорусская государственная академия связи», г. Минск, Республика Беларусь, преподаватель*

*2 Учреждение образования «Белорусская государственная академия связи», г. Минск, Республика Беларусь, студент*

Алгоритм Дейкстры используется для поиска кратчайшего маршрута от одной из вершин графа до всех остальных. Алгоритм широко применяется в программировании и технологиях, например, его используют протоколы маршрутизации OSPF и IS-IS. Концептуально эти протоколы схожи, оба поддерживают переменную длину маски, могут использовать групповую рассылку для обнаружения соседних [маршрутизаторов](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%80%D1%88%D1%80%D1%83%D1%82%D0%B8%D0%B7%D0%B0%D1%82%D0%BE%D1%80) посредством hello-пакетов, и могут работать с аутентификацией для обмена маршрутами.

OSPF изначально был создан, как протокол для маршрутизации в IP сетях, IS-IS же работает поверх канального уровня модели OSI, поэтому он не привязан к конкретному протоколу сетевого уровня. Также IS-IS не использует протокол IP для доставки сообщений, содержащих информацию о маршрутизации (LSA и прочее).

Главным сходством этих протоколов является то, что они оба используют алгоритм Дейкстры для расчета наилучшего пути. Данный алгоритм применяется для нахождения кратчайшего пути для взвешенного ориентированного графа без дуг и ребер отрицательной длины [1].

Алгоритм Дейкстры представляет собой итерационный процесс. Для понимания работы алгоритма введем такое понятие как пометка вершины. Пометкой вершины называется значение равное суммарному весу ребер от вершины-истока до исследуемой вершины. Вершина-исток определяет начало пути, а вершина-сток конец соответственно.

Пометка вершины может иметь два состояния: постоянное и временное. Пометка вершины становится постоянной, когда принимает минимальное возможное значение, т. е. не может стать лучше. При чем на каждой итерации одна из пометок вершины становится постоянной [2].

Перед началом выполнения алгоритма задаем пометки для всех вершин. Пометка для вершины-истока равна нулю, для всех остальных вершин бесконечности, т. к. еще не были определены расстояния от вершины-истока до всех остальных вершин графа.

Выбираем наименьшую пометку, в первой итерации это будет ноль, и делаем пометку данной вершины постоянной. Рассматриваем смежные с ней вершины на предмет уменьшения пометок. Для обновления пометки необходимо найти минимальное значение между старой пометкой и суммой двух значений, а именно, пометки постоянной вершины на данной итерации и расстояния (вес ребра) от этой вершины до остальных вершин графа. Если новая пометка для вершины оказывается больше, то пометка не обновляется [3].

Если пометка была уменьшена, то необходимо записать из какой вершины мы пришли в текущую в соответствующую таблицу.

Следующий шаг – из всего множество вершин с временной пометкой выбрать наименьшую и перевести ее в статус постоянной. Повторяем действие описанные раннее до тех пор, пока наша вершина-сток не примет значение постоянной метки.

Чтобы построить маршрут, необходимо проанализировать значение ячеек таблицы, в которой указано, из какой вершины в какую мы пришли.

Для программной реализации данного алгоритма использовался объектно-ориентированный язык Java. Для поиска кратчайшего пути необходимо заполнить матрицу смежности, а также указать вершину-исток и вершину-сток. Запуск программы осуществляется нажатием кнопки «Построить путь». После чего программа выдаст результат, а именно полный путь с указанием всех вершин и значением кратчайшего расстояния, также программа отобразит время, за которое был рассчитан путь.

Демонстрация работы программы показана на рисунке 1.

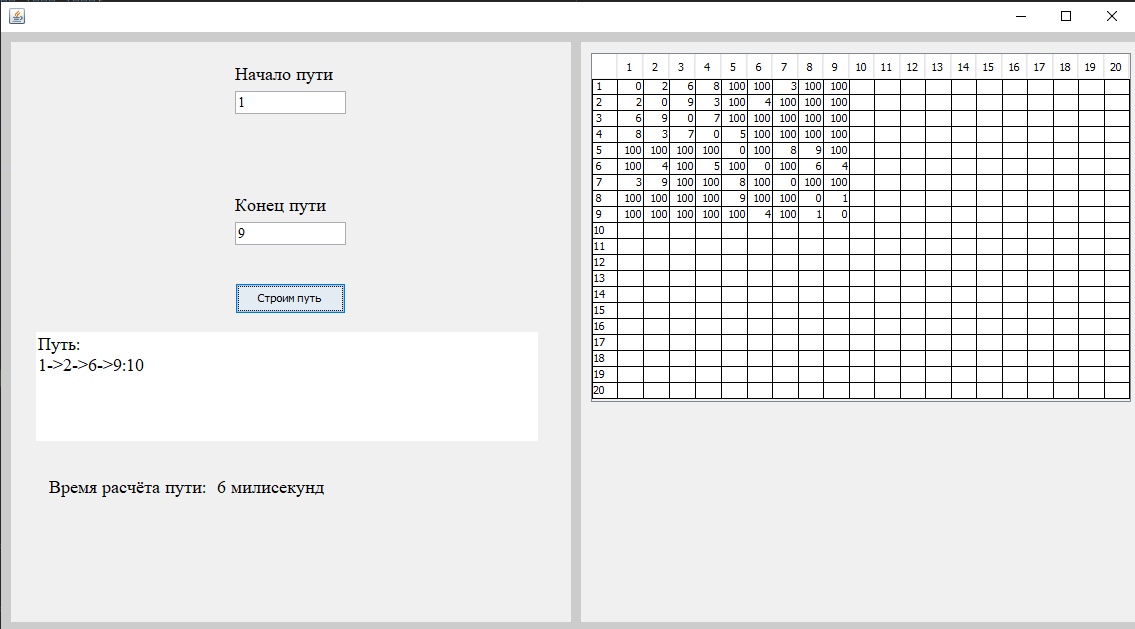


Рисунок 1– Программная реализация алгоритма Дейкстры

Таким образом, данная программа может использоваться для поиска кратчайшего пути при работе с протоколами маршрутизации OSPF и IS-IS.

ЛИТЕРАТУРА

1. Алгоритм Дейкстры – Википедия [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Алгоритм\_Дейкстры

2. Ревотюк, М. П. Регулярный встречный поиск кратчайших путей на графах / М. П. Ревотюк, Н. В. Хаджинова, А. К. Пушкина // Информационные технологии и системы 2014 (ИТС 2014) : материалы междунар. науч. конф. БГУИР, Минск, Беларусь, 29 окт. 2014 г. / редкол.: Л. Ю. Шилин [и др.]. – Минск : БГУИР, 2014. – С. 306–307.

3. Ковалев, М. Я. Абсолютная устойчивость в задачах оптимального выбора с фиксированным прошлым / М. Я. Ковалев // Докл. Нац. акад. наук Беларуси. – 2018. – Т. 62, № 2. – С. 147–150.