Краевая научно-практическая конференция

учебно-исследовательских работ учащихся 6-11 классов

«Прикладные и фундаментальные вопросы математики и физики»

прикладные вопросы математики

Выбор оптимального метода нахождения кратчайшего пути в графе

Лебедев Арсений Александрович,

11 кл., МБОУ "Лицей №1", г. Перми,

Котельникова Наталья Васильевна,

учитель информатики

Пермь. 2017.

Введение

В современном мире очень важно быстродействие и точность выполнения различных задач. Одним из способов, при котором достигается быстрое и точное выполнение задачи, является граф. Граф – это совокупность вершин графа и его ребер, на которых имеется “вес”, “вес” – это число, которое находится на ребре и равняется расстоянию от предыдущей вершины к последующей. В современном мире у графов есть множество применений. Начиная с нахождения кратчайшего пути при перевозке груза из города в город, затрачивая минимум времени, а значит минимум бензина и минимум денег и, заканчивая на различных технологиях, в которых очень важно быстро найти решение той или иной задачи. Таким образом, исследование в данной тематике является особо актуальным в настоящее время.

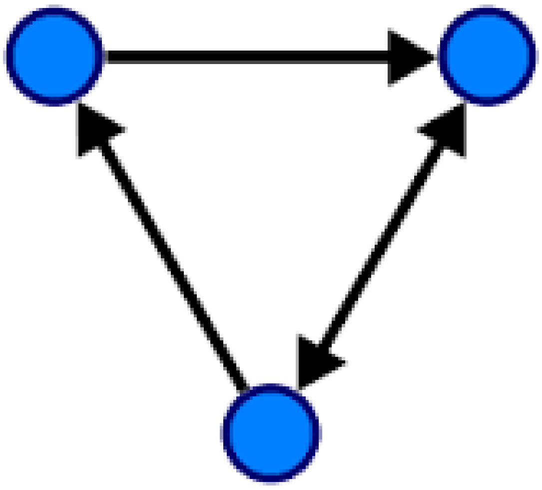


Рис. 1. Ориентированный граф

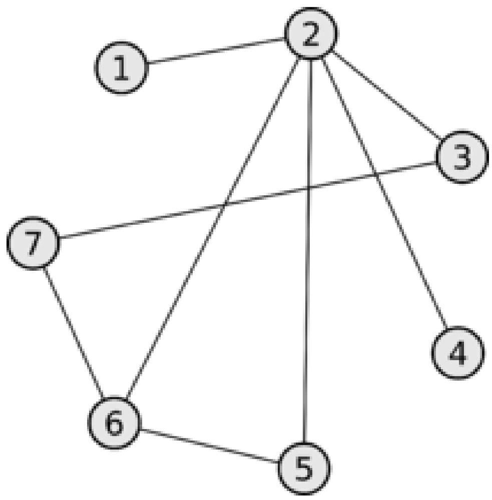


Рис. 2. Неориентированный граф

Алгоритмов нахождения кратчайшего пути в графе достаточно много: волновой алгоритм, алгоритм Левита, переборный алгоритм, алгоритм Беллмана — Форда, алгоритм Джонсона и другие. Есть более популярные методы, которые обширно применяются в наше время, например алгоритм Дейкстры или Флойда-Уоршелла.

Алгоритм Флойда-Уоршелла был опубликован в 1962 году, он используется для нахождения кратчайшего пути во взвешенном ориентированном графе, ориентированным графом называется такой граф, у которого задано направление движения по ребру. Быстродействие данного алгоритма достигает количества вершин графа в кубе, потому что нам необходимо просмотреть квадратную матрицу порядка количества вершин ровно количество вершин раз, то есть количество вершин графа в кубе. Память, которая затрачивается на данный метод, равна количеству вершин графа в квадрате.

Алгоритм Дейкстры был опубликован в 1959 году, он используется для нахождения кратчайшего пути в графе без ребер отрицательного “веса”. Данный алгоритм широко применяется в программировании и технологиях, например, его используют протоколы маршрутизации OSPF и IS-IS, чтобы выбрать наилучший путь, поэтому эти протоколы очень экономно используют пропускную способность сетей. Быстродействие данного алгоритма достигает повторения всего алгоритма квадрат раз.

Таким образом, целью данного исследования является сравнение методов или модификаций этих методов. Для достижения цели были поставлены следующие задачи:

1. Обзор литературы.
2. Сравнение различных подходов.
3. Выбор корректного метода.
4. Реализация задачи на ЭВМ при помощи современных средств.
5. Анализ результатов.

Сравнение различных подходов

Принято решение сравнивать два алгоритма, а именно – алгоритм Дейкстры и Флойда-Уоршелла. Необходимо, чтобы алгоритм быстро выполнялся и был универсальным. Были написаны две программы на языке C++ в среде разработки VisualStudio 2017. В алгоритме Флойда-Уоршелла вся матрица графа пробегается три раза, в каждом пробеге, затем в последнем пробеге определяется кратчайший путь. Данный алгоритм работает с любыми ребрами, то есть как с отрицательными, так и с положительными. Сложность данного алгоритма является кубической, то есть весь алгоритм по времени займет количество вершин графа в кубе.

```
for k = 1 to n
  for i = 1 to n
    for j = 1 to n
      W[i][j] = min(W[i][j], W[i][k] + W[k][j])
```

Рис. 3. Алгоритм Флойда-Уоршелла в простом представлении

Алгоритм Дейкстры: сначала идет проверка каждого элемента матрицы на проход к ней напрямую от первичного элемента, если таковые находятся, то записываем к этим элементам максимальное значение, в программировании это называется INT_MAX или INT16_MAX, или INT32_MAX. Затем матрица пробегается дважды, и каждый раз складывает веса ребер с каждой последующей вершиной, в конце определяется минимальный из них.

Сложность данного алгоритма является квадратичной, то есть весь алгоритм по времени займет количество вершин графа в квадрате.

Алгоритм Дейкстры сегодня особенно актуален, он используется при проложении маршрута в 2GIS и в иерархической маршрутизации. Также нередко он используется, когда необходимо выбрать менее затратный по времени и средствам путь, здесь учитываются уже два фактора, на которых стоит вся задача. Именно поэтому алгоритм Дейкстры является наиболее применяемым и востребованным.

Существуют также более быстрые и надежные алгоритмы, но их применяют редко, так как они требуют больших затрат и эти затраты не окупят быстродействие. Большим минусом алгоритма Дейкстры является то, что он не может работать с ребрами отрицательного веса.

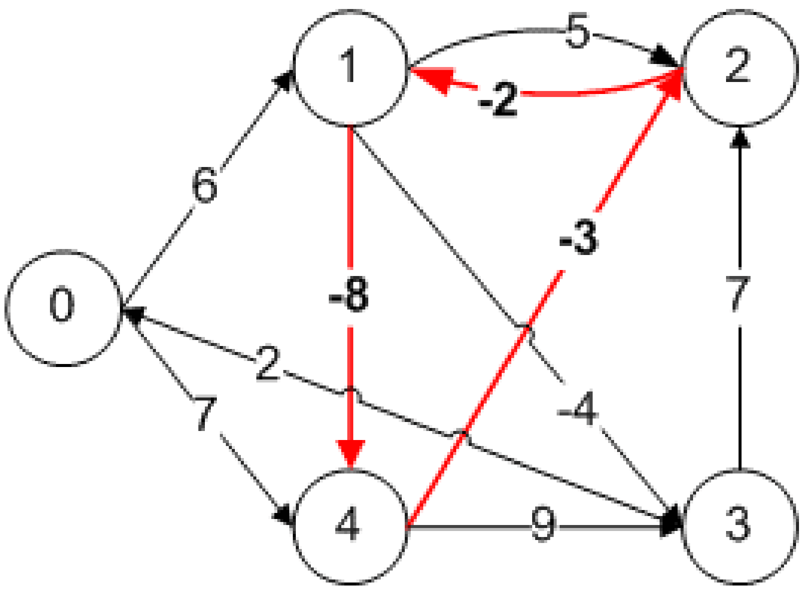


Рис. 4. Пример графа с ребрами, имеющими отрицательный “вес”

Практическая часть. Анализ результатов

В результате двадцати проверок для каждого алгоритма на быстродействие при нахождении кратчайшего пути между 700 вершинами графа алгоритмами Дейкстры и Флойда-Уоршелла, выяснилось, что алгоритм Дейкстры работает в разы быстрее, нежели алгоритм Флойда-Уоршелла. Каждая проверка это нахождение кратчайшего пути в массиве, который не имел нулей, а значит, исключается появление нулевого массива.

Рис. 5. Отношение затраченного времени к каждой проверке

Наименьшее затраченное время равно 4.254 с.

Наибольшее затраченное время равно 4.417 с.

Среднее затраченное время равно 4.3396 с.

Рис. 6. Отношение затраченного времени к каждой проверке

Наименьшее затраченное время равно 0.1407 с.

Наибольшее затраченное время равно 0.5077 с.

Среднее затраченное время равно 0.2368 с.

Также был разработан алгоритм, который справляется с задачами как алгоритм Дейкстры, но быстрее его. Было принято решение полностью отказаться от массива, в который записывались кратчайшие пути, и заносить кратчайшие расстояния в саму матрицу, как потом будет видно, это изменение повлияет на быстродействие алгоритма в лучшую сторону.

Для этой программы также был проведен тест на быстродействие, ниже представлен график.

Рис. 7. Отношение затраченного времени к каждой проверке

Наименьшее затраченное время равно 0.1267 с.

Наибольшее затраченное время равно 0.2913 с.

Среднее затраченное время равно 0.2010 с.

После измерения быстродействия алгоритма, необходимо было его проверить на реальных задачах, например на задачах из ЕГЭ.

Ниже приведены задачи из ЕГЭ в качестве примеров.

	A	B	C	D	E	F
A		4				
B	4		6	3	6	
C		6			4	
D		3			2	
E		6	4	2		5
F					5	

Рис. 8. Задача из ЕГЭ, найти кратчайшее расстояние от A к F

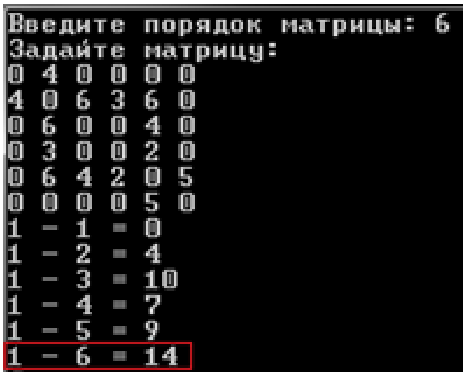


Рис. 9. Решение задачи из ЕГЭ из Рис. 6.

Кратчайшее расстояние от A до F равно 14, теперь это надо проверить, ниже приведен граф, построенный по этой матрице.

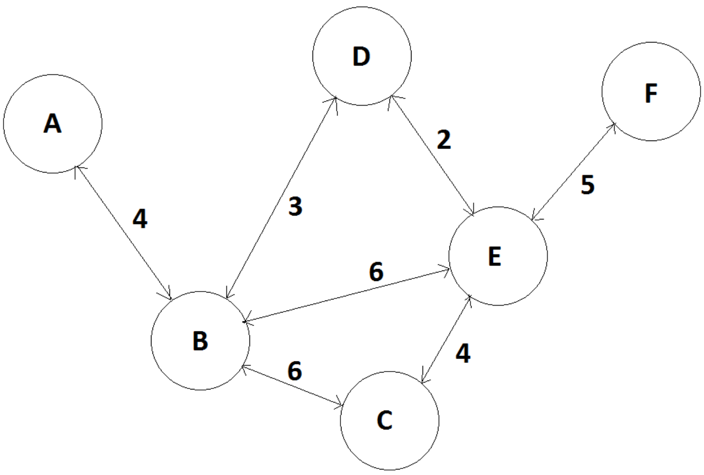
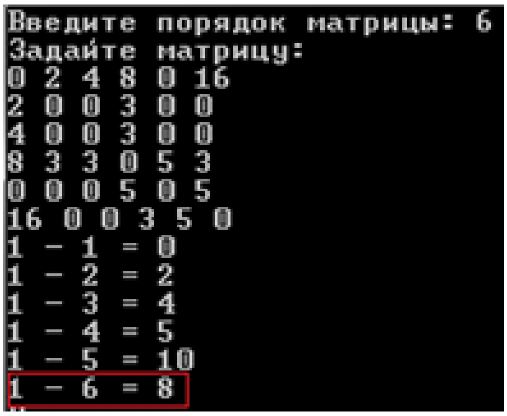


Рис. 10. Граф, построенный по матрице из Рис. 6.

Кратчайший путь будет такой: A-B-D-E-F, а сумма “веса” его ребер равна 14, значит программа посчитала все верно.

	A	B	C	D	E	F
A		2	4	8		16
B	2			3		
C	4			3		
D	8	3	3		5	3
E				5		5
F	16			3	5	

Рис. 11. Задача из ЕГЭ, найти кратчайшее расстояние от A к F



```
Введите порядок матрицы: 6
Задайте матрицу:
0 2 4 8 0 16
2 0 0 3 0 0
4 0 0 3 0 0
8 3 3 0 5 3
0 0 0 5 0 5
16 0 0 3 5 0
1 - 1 = 0
1 - 2 = 2
1 - 3 = 4
1 - 4 = 5
1 - 5 = 10
1 - 6 = 8
```

Рис. 12. Решение задачи из ЕГЭ из Рис. 9.

Кратчайшее расстояние от A до F равно 8, теперь это надо проверить, ниже приведен граф, построенный по этой матрице.

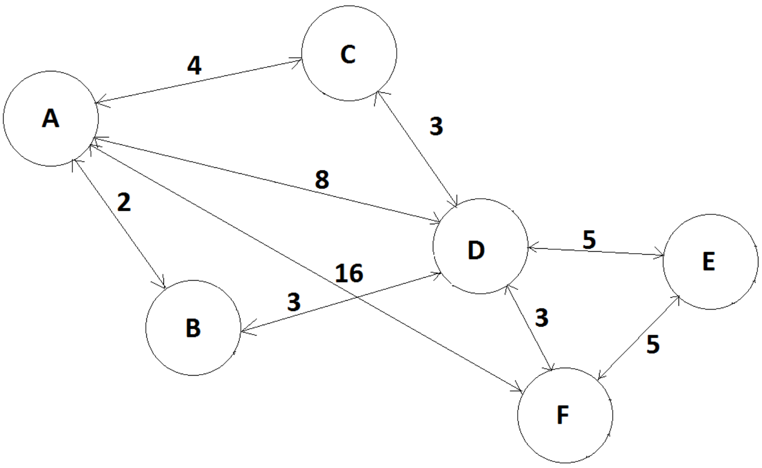


Рис. 13. Граф, построенный по матрице из Рис. 9.

Кратчайший путь будет такой: A-B-D-F, а сумма “веса” его ребер равна 8, значит программа посчитала все верно.

Заключение

Были изучены два самых популярных и рабочих алгоритма для нахождения кратчайшего пути в графе от любой из вершин к любой другой. Была изучена необходимая литература. Также проведена реализация алгоритмов на ЭВМ, сравнение быстродействия алгоритмов, создан более быстрый аналог алгоритму Дейкстры, был проведен анализ результатов. В итоге получилось, что алгоритм Дейкстры быстрее, чем алгоритм Флойда-Уоршелла. Алгоритм Дейкстры очень часто применяется в современном мире, так как он достаточно быстрый и не сильно затратный.

Список литературы

1) goo.gl/NZ2eoE
2) goo.gl/16b5mg
3) goo.gl/mua7EX
4) goo.gl/4z16hU
5) goo.gl/7VC9mY
6) goo.gl/p71mv6