

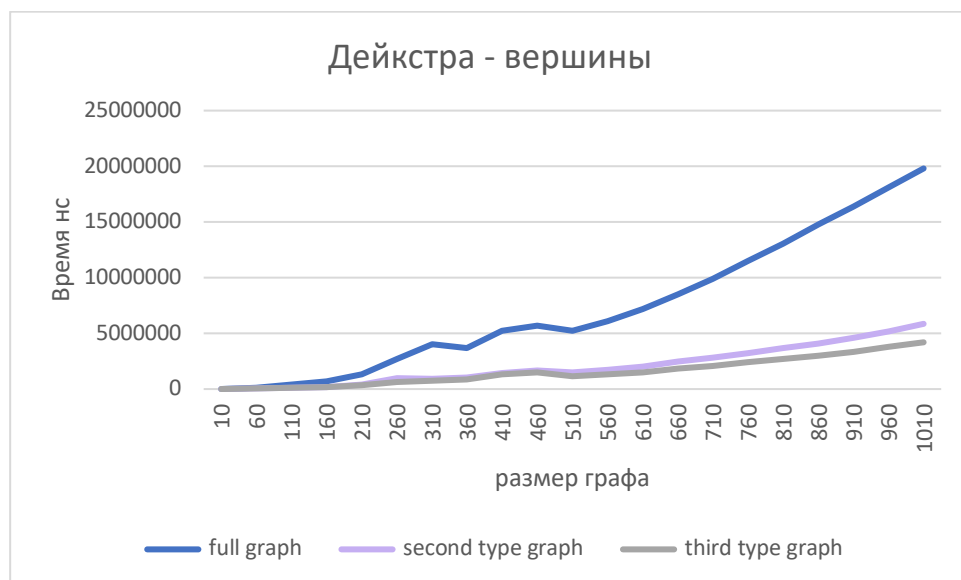
Отчет по 4 домашнему контрольному заданию

Галкина Таисия Олеговна БПИ213

Необходимо было построить несколько графиков (отдельно для каждого алгоритма по времени и относительно количества вершин и количества ребер и относительно трех типов графов). Каждый алгоритм для каждого графа запускался по 5 раз, а потом суммарное затраченное время на данный алгоритм и данный граф делился на 5, чтобы усреднить время и «сгладить» выбросы.

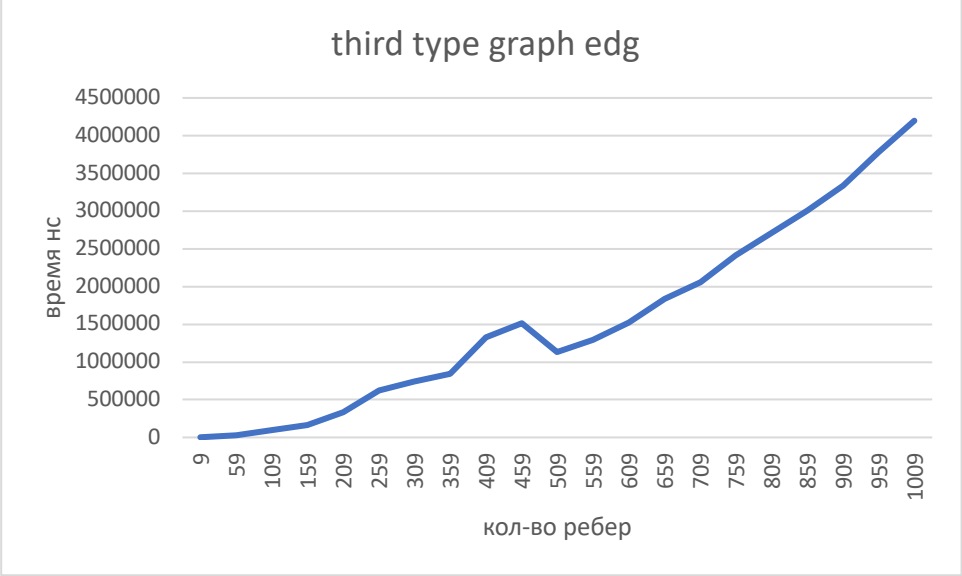
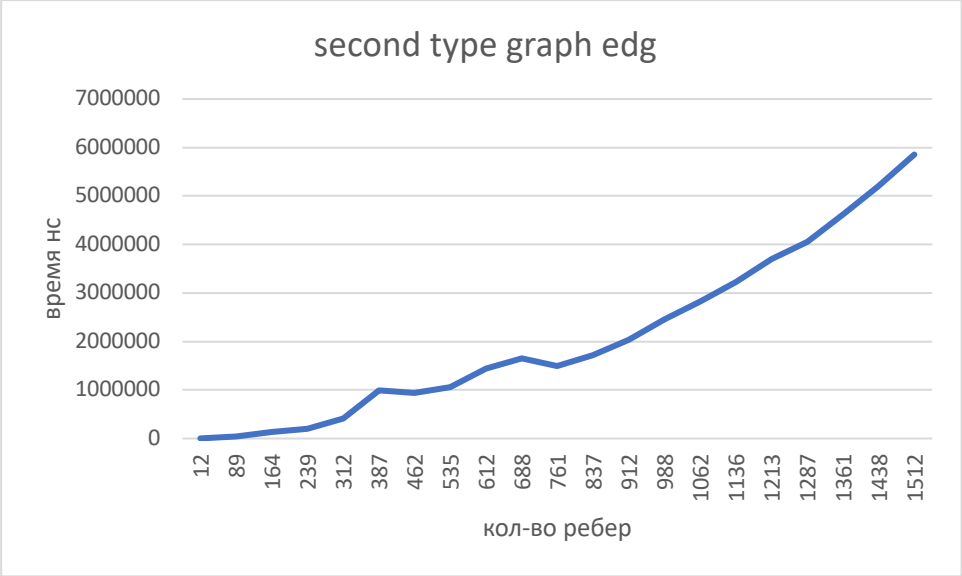
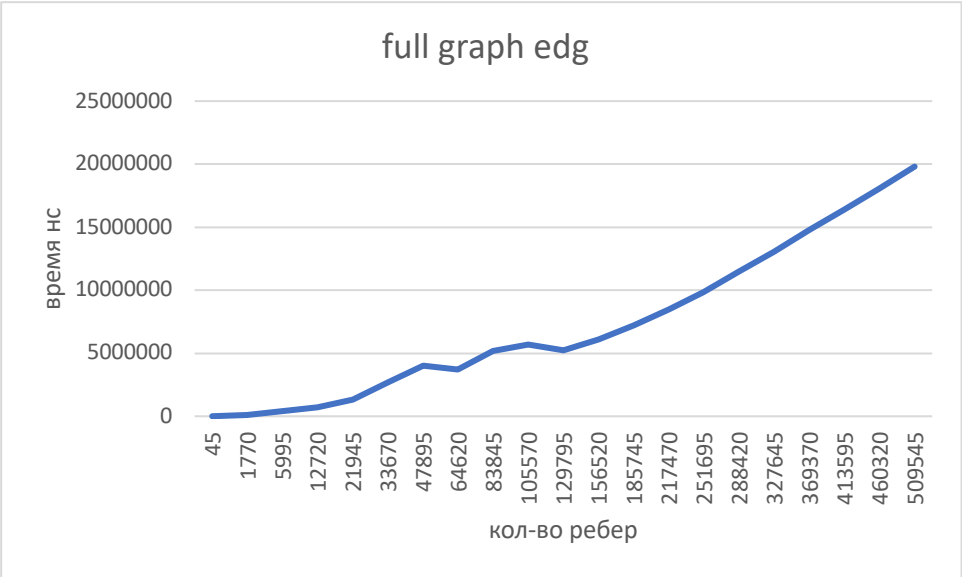
Проанализируем алгоритм Дейкстры:

Из графика видно, что медленнее всего алгоритм работает на полном графе, на остальных типах графов алгоритм работает примерно одинаково (дерево лучше всего). Это так. Поскольку данный алгоритм зависит от количества ребер ($O(E \cdot \log(V))$)



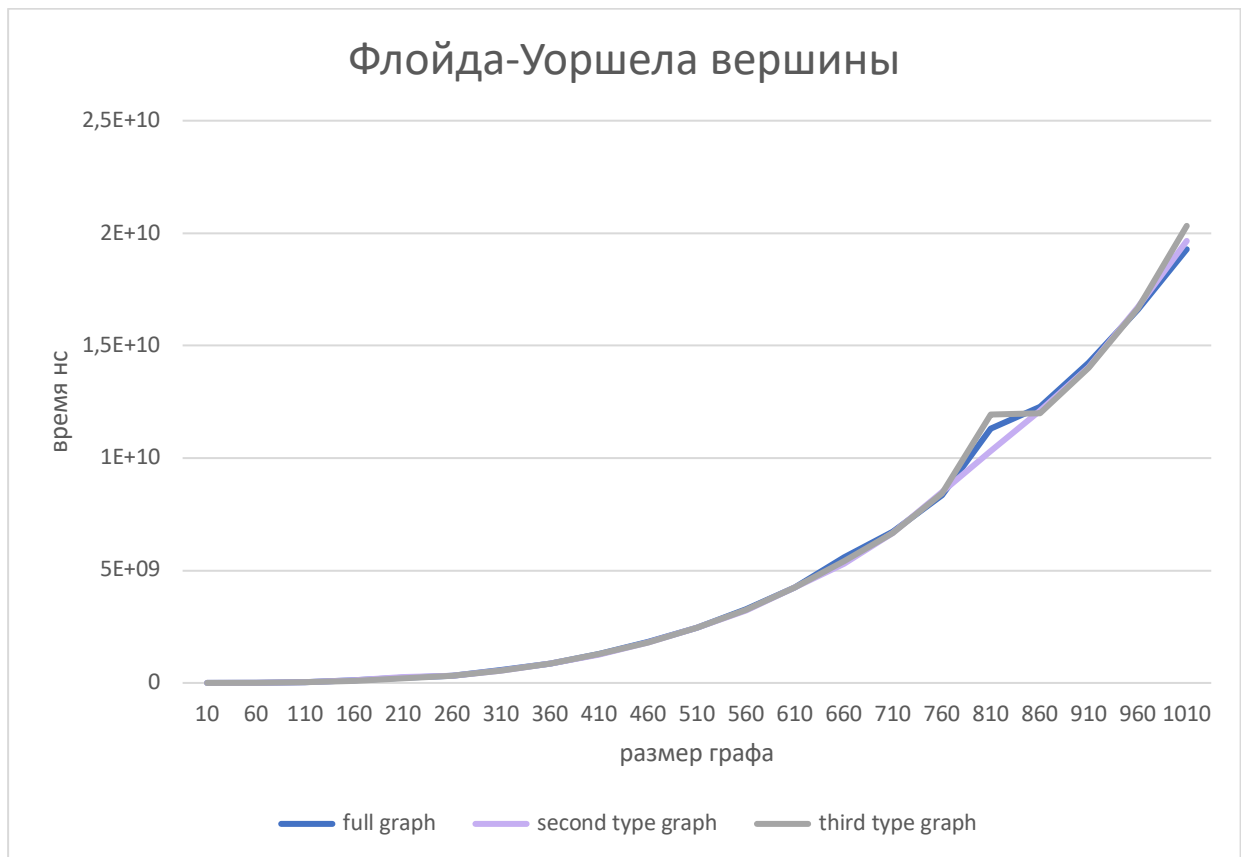
Рассмотрим графики зависимости времени от количества ребер:

Тут тоже можно заметить, что чем больше ребер, тем дольше выполняется алгоритм

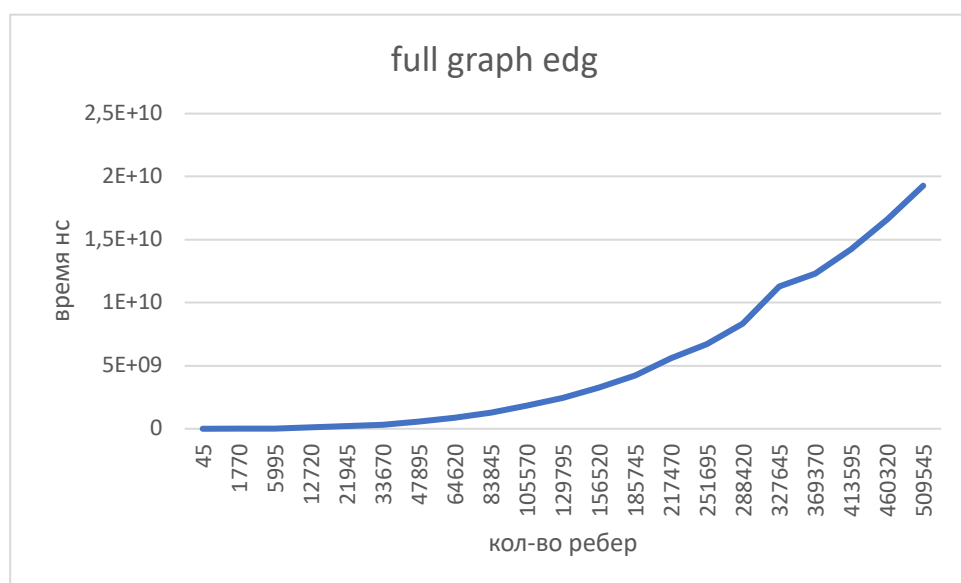


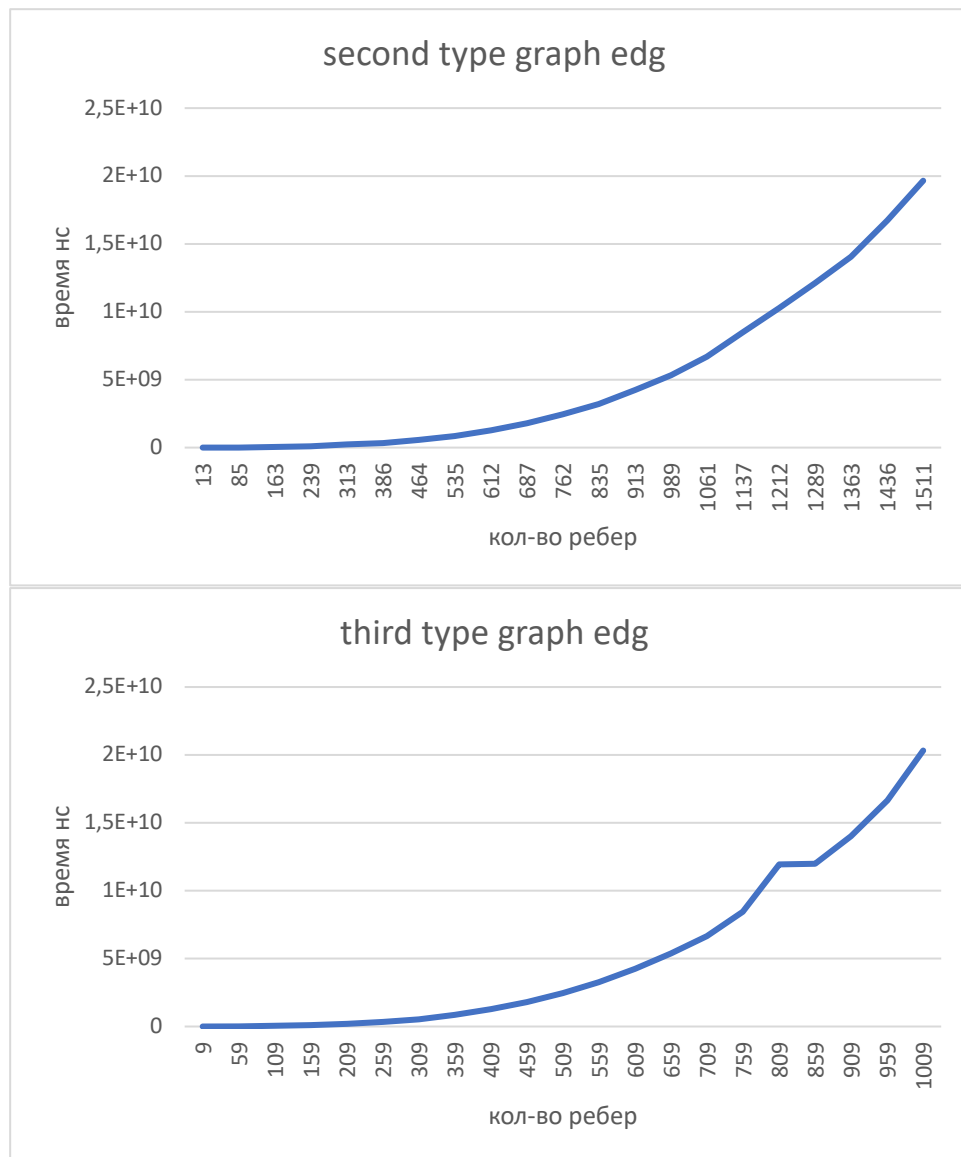
Проанализируем алгоритм Флойда-Уоршела:

Его асимптотика зависит только от количества вершин($O(V^3)$), поэтому на итоговом графике можно увидеть как три линии «сливаются» в одну



Рассмотрим графики зависимости времени от количества ребер:





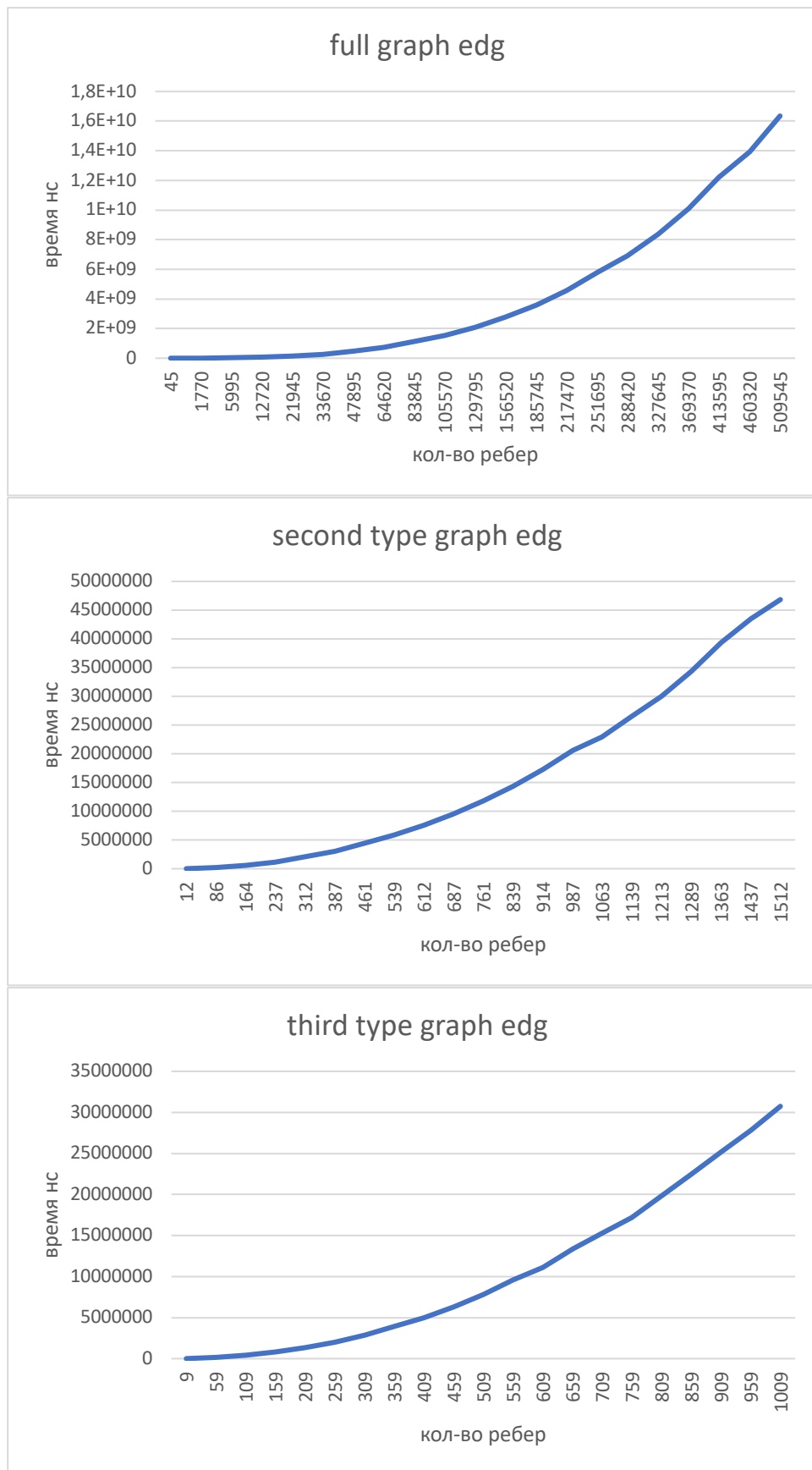
В нашем случае больше вершин -> значит больше ребер, а значит графики растут верно

Проанализируем алгоритм Форда-Беллмана:

Асимптотика данного алгоритма зависит и от количества ребер и от количества вершин($O(VE)$). Получается, чем больше ребер и вершин, тем алгоритм медленнее. Как видно из графика хуже всего и работает как раз алгоритм на полном графе, в то время как на остальных видах графа время выполнения примерно одинаковое:



Рассмотрим зависимость времени от количества ребер:

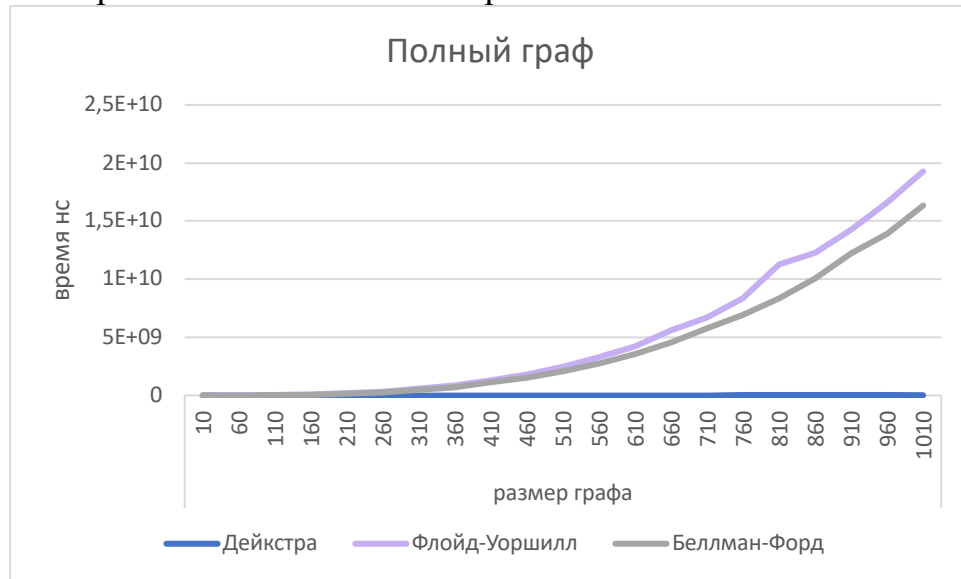


Можно заметить, что асимптотика будет лучше там, где ребер меньше

Теперь проанализируем для каждого типа графа различные алгоритмы:

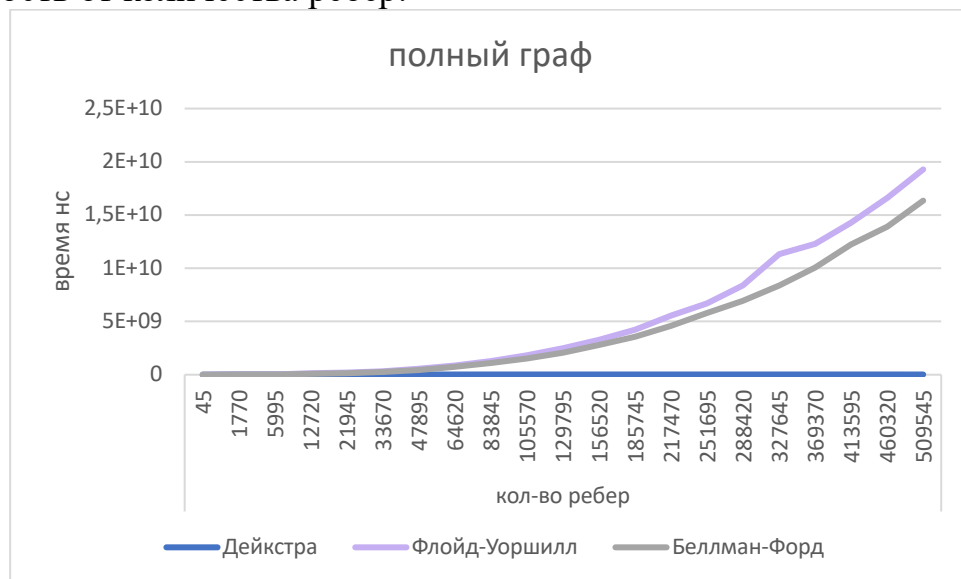
Полный граф:

Зависимость времени от количества вершин:



Заметим, что быстрее всего работает алгоритм Дейкстры, что подтверждает ранее сказанную асимптотику. Хуже всего работает алгоритм Флойда-Уоршелла, тоже подтверждает ранее сказанную асимптотику.

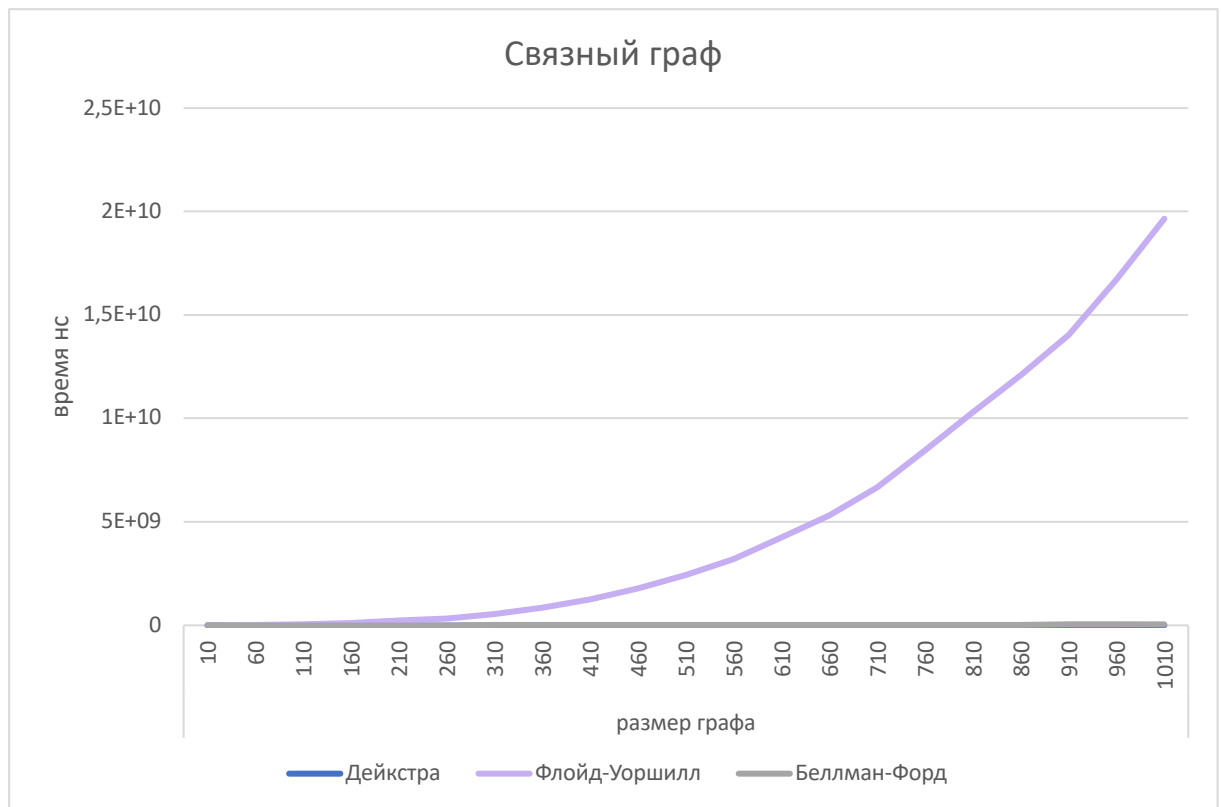
Зависимость от количества ребер:



Можно заметить, что график достаточно схож с прошлым. Итог тоже такой же – лучше всего работает Дейкстра, а хуже Флойд-Уоршелл. Это опять же подтверждает асимптотику.

Связный граф:

Зависимость времени от количества вершин:



Заметим, что быстрее всего работает алгоритм Дейкстры и Беллман-Форд, что подтверждает раннее сказанную асимптотику. Хуже всего работает алгоритм Флойда-Уоршелла, поскольку он зависит от количества вершин, а предыдущие алгоритмы — от количества ребер.

Зависимость от количества ребер:



Можно заметить, что график достаточно схож с прошлым. Итог тоже такой же — лучше всего работает Дейкстра, а хуже Флойд-Уоршелл. Это опять же подтверждает асимптотику. Если детальнее посмотреть на данные, то можно еще заметить, что в начале Дейкстра работает лучше, чем Беллман-Форд.

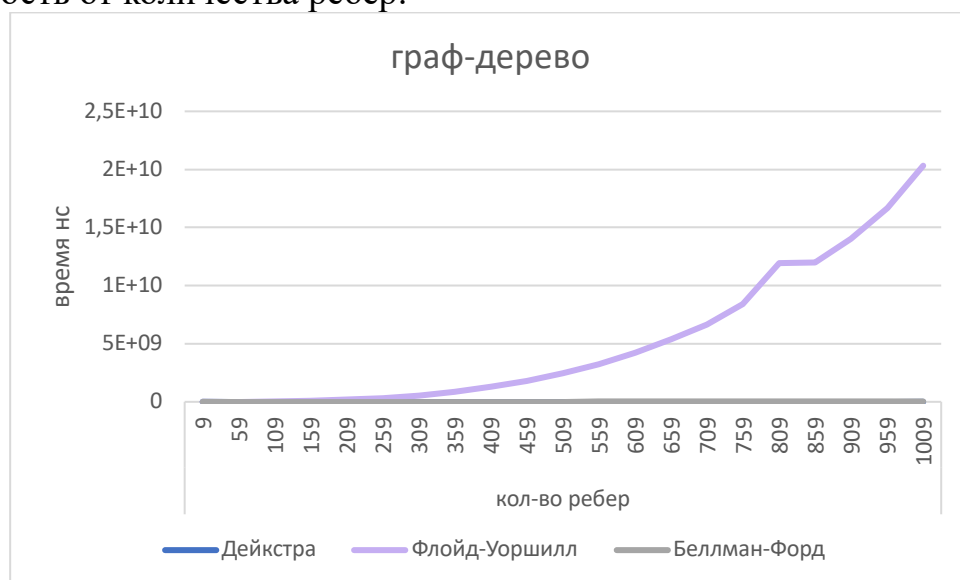
Дерево:

Зависимость времени от количества вершин:



Заметим, что быстрее всего работает алгоритм Дейкстры и Беллман-Форд, что подтверждает раннее сказанную асимптотику. Хуже всего работает алгоритм Флойда-Уоршелла, поскольку он зависит от количества вершин, а предыдущие алгоритмы и от количества ребер, которых тут меньше (такая значительная разница поскольку асимптотика Флойда-Уоршелла кубическая). Если детальнее посмотреть на данные, то можно еще заметить, что Дейкстра работает лучше, чем Беллман-Форд. Но тут второй алгоритм все «ближе» к Дейкстре, но логарифм от вершин «спасает» Дейкстру.

Зависимость от количества ребер:



Можно заметить, что график достаточно схож с прошлым. Итог тоже такой же – лучше всего работает Дейкстра, а хуже Флойда-Уоршелла. Это опять же подтверждает асимптотику. Аналогичная ситуация «конкуренции» происходит между Дейкстрой и Беллманом-Фордом.

Итог:

Все асимптотические данные подтвердились на графиках.

Если необходимо найти кратчайший путь между двумя вершинами, то логично взять алгоритм Дейкстры или Беллмана-Форда. При таком запросе алгоритм Флойда-Уоршелла покажется бесполезным. Однако если необходимо будет узнать кратчайшие пути между всеми вершинами, то тут как раз лучше всего и будет алгоритм Флойда-Уоршелла, когда как первые два уже не будут так хороши.

// в коде еще реализован алгоритм Джонсона, но я не успела внести данные, но по-идее он должен быть чуть хуже дейкстры с Беллманом Фордом ибо он их содержит вот. Но явно лучше Флойда-Уоршелла. Хорошего дня!)