МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ БЕЛАРУСЬ

БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И РОБОТОТЕХНИКИ

КАФЕДРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ И ТЕХНОЛОГИЙ

Отчет

по лабораторной работе № **5**

по дисциплине

«Распределенная и параллельная обработка данных»

тема

«**Программная реализация и экспериментальное исследование метода свертывания графа распараллеленности операций**»

|  |  |  |
| --- | --- | --- |
| Исполнитель: |  | студент группы 10701117  Рашкевич В.А  Болтромюк П.А. |
|  |  |  |

### Минск, 2020

**План лабораторной работы:**

1. Изучение задачи многошагового планирования параллельных вычислений.
2. Изучение графа распараллеленности операций.
3. Изучение последовательно-параллельного синхронного планирования методом свертывания графа распараллеленности операций.
4. Разработка архитектуры программы.
5. Разработка структур данных для представления операций, графа распараллеленности операций, последовательно-параллельного плана.
6. Программная реализация алгоритма свертывания графа.
7. Отладка программы.
8. Проведение вычислительного эксперимента на конкретных исходных данных (согласовать с преподавателем).
9. Сдача лабораторной работы.

**Ход работы:**

Теоретическая часть

Граф распараллеленности операций определяется как GR=(V,R), где V – множество вершин, являющихся частичными последовательно-параллельными планами и, в частном случае, операциями; R – множество ребер, описывающих отношение распараллеленности частичных планов (операций). Граф является неориентированным. Две вершины i,k ϵ V соединены ребром, если соответствующие частичные планы выполняются параллельно, если вершины не соединены ребром, соответствующие планы выполняются параллельно. Граф является взвешенным. Каждой вершине i ϵ N ставится в соответствие два параметра: время Ti выполнения частичного плана (в частном случае, это время ti выполнения операции) и вектор bi=(bi1,…,bim) чисел bij процессоров типов j=1,…,m, необходимых для выполнения плана.

Исходный граф G0R на множестве вершин-операций строится по графу GH. В граф G0R вводится ребро (i,k), если на ориентированном графе GH существует путь, соединяющий операции i и k, в противном случае ребро не вводится.

Граф распараллеленности 𝐺𝑅0 является исходными данными для синтеза последовательно-параллельного плана методом свертывания. Свертывание графа 𝐺𝑅0 выполняется посредством пошагового склеивания пары вершин, при этом в графе появляются более сложные вершины, которым соответствуют частичные последовательно-параллельные планы и которые также метятся двумя интегрированными величинами:

1. временем Ti выполнения частичного плана;

2. вектором bi чисел процессоров различных типов, необходимых для реализации частичного плана.

Исходные данные:

1. Граф GH предшествования операций.

2. Времена ti, i ϵ N выполнения операций.

3. Время tstep шага управления.

4. Число pi доступных процессоров типа i=1,…,Types.

Результирующие данные:

1. Шаги свертывания.

2. Граф на каждом шаге.

Описание алгоритма:

Рассмотрим последовательно-параллельное планирование с целью минимизации времени решения задачи при заданных ограничениях на ресурсы. Ограничения представим вектором bmax = (b1max,…,bmmax) максимального числа используемых процессоров каждого типа. Свертывание графа GR посредством склеивания пар вершин. Пусть на шаге s планирования в графе 𝐺𝑅𝑠 для склеивания выбраны вершины x и y. В результате склеивания x и y образуется новая вершина с именем xy. Граф 𝐺𝑅𝑠 преобразуется путем удаления вершин x и y, удаления множества ребер, инцидентных этим вершинам, а также путем добавления вершины xy и добавления ребер, инцидентных этой вершине и вершинам из множества вершин, смежных с вершиной x в графе 𝐺𝑅𝑠. Если вершины x и y соединены ребром, они склеиваются параллельно, при этом новой вершине соответствует параллельный план par(x,y). Если вершины не соединены ребром, они склеиваются последователь- но, при этом новой вершине соответствует последовательный план seq(x,y). Время выполнения Txy и вектор bxy для новой вершины xy вычисляются следующим образом:

1) если вершины склеиваются параллельно, то 𝑇𝑥𝑦=max⁡(𝑇𝑥,𝑇𝑦) и 𝑏𝑥𝑦=𝑏𝑥+ 𝑏𝑦;

2) 2) если вершины склеиваются последовательно, то 𝑇𝑥𝑦=𝑇𝑥+ 𝑇𝑦 и 𝑏𝑥𝑦=max(𝑏𝑥,𝑏𝑦)=(max(𝑏𝑥1,𝑏𝑦1),…,max⁡(𝑏𝑥𝑚,𝑏𝑦𝑚)).

Параллельное склеивание вершин увеличивает число используемых процессоров, последовательное склеивание увеличивает время выполнения плана. Параллельное склеивание вершин не должно приводить к нарушению ограничений на максимальное число процессоров, описываемых вектором 𝑏𝑚𝑎𝑥. Если нарушение ограничений имеет место, ребро может быть заранее удалено из графа 𝐺𝑅𝑠. Различный порядок склеивания пар вершин приводит к построению различных последовательно-параллельных планов. Существенное влияние на качество синтезируемого плана оказывает потеря ребер, обусловленная с одной стороны, особенностями механизма склеиванием вершин и, с другой стороны, учетом ограничения на число процессоров bmax. Потеря ребра при склеивании вершин x и y происходит тогда, когда существует вершина z, смежная с вершиной x и не смежная с вершиной y. В результате такого склеивания новая вершины xy и вершина z не соединены ребром. Это означает потерю

потенциального параллелизма между вершинами xy и z, что может сказаться отрицательно на минимизации времени выполнения синтезируемого плана. В силу сказанного, предпочтительно необходимо выбирать пары вершины, склеивание которых минимизирует потерю ребер. Склеивание пар вершин и свертывание графа распараллеленности завершается тогда, когда граф включает одну вершину. Этой вершине соответствует суперпозиция функций seq и par, описывающая синтезированный последовательно-параллельный план.

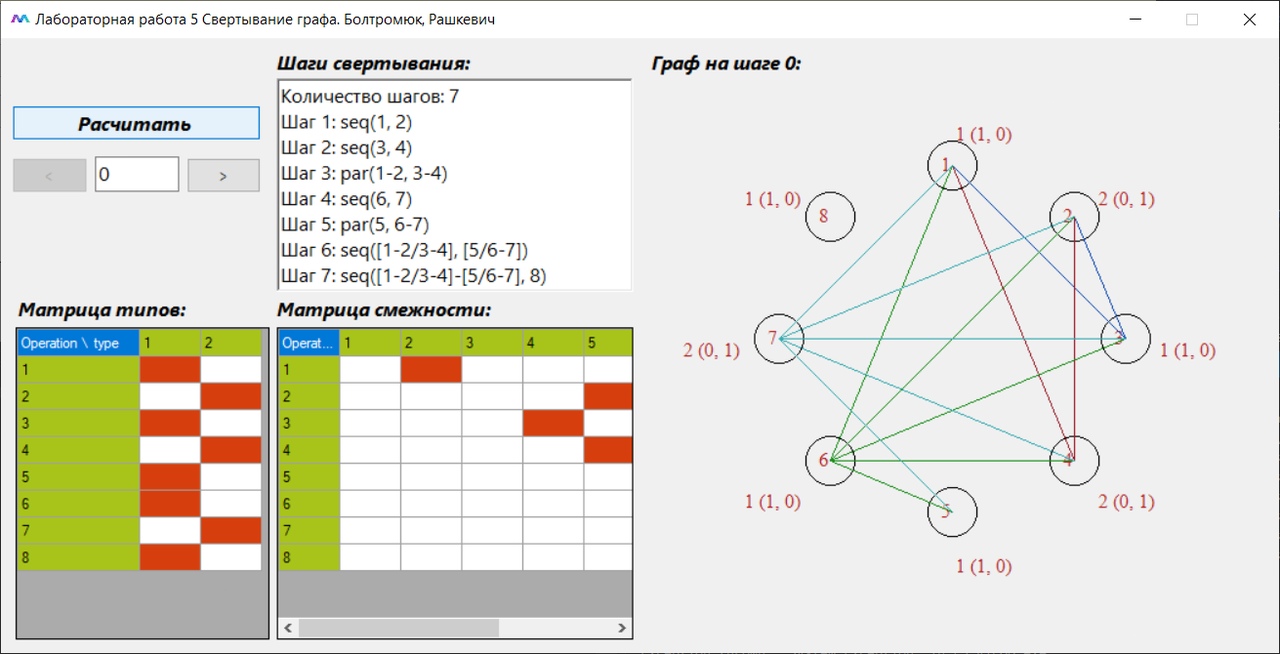


Рисунок 1-Реализация свертывания графа шаг1

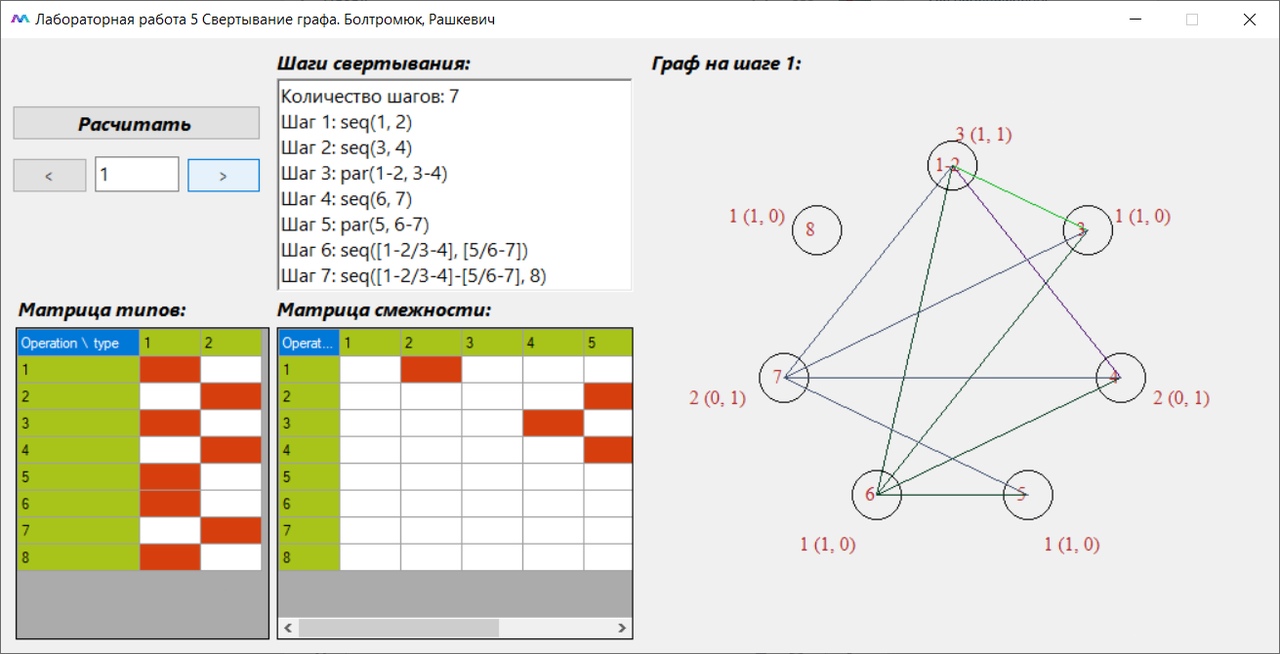


Рисунок 1-Реализация свертывания графа шаг2

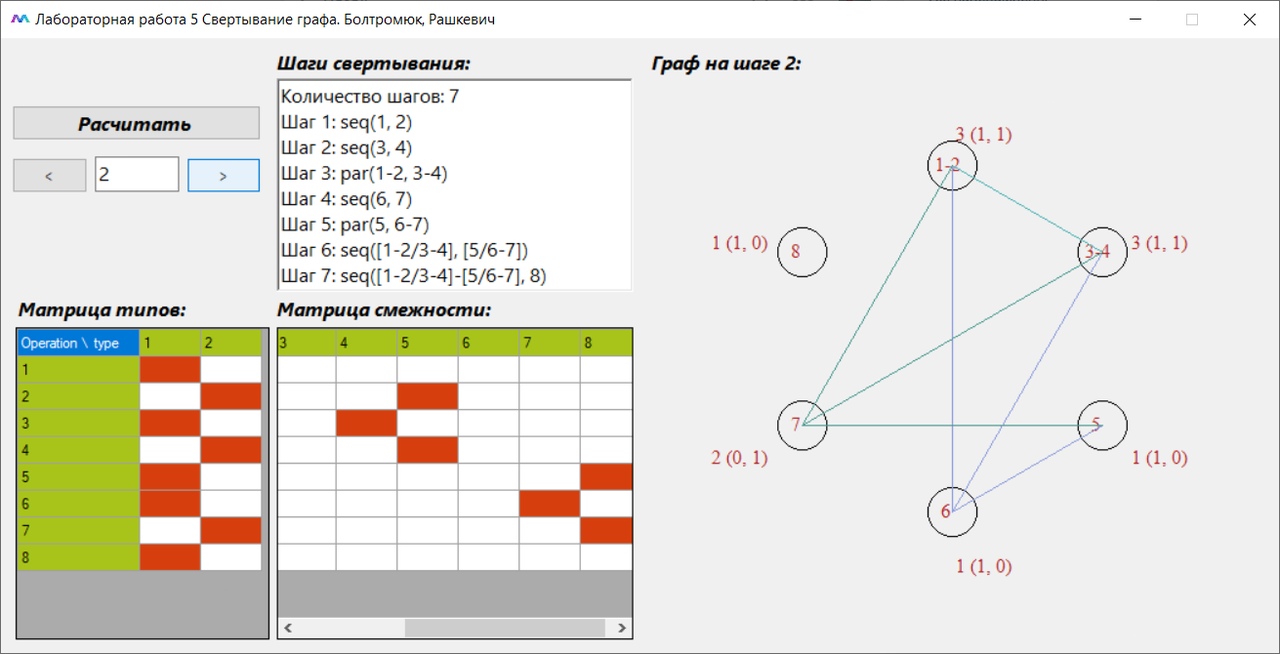


Рисунок 1-Реализация свертывания графа шаг3

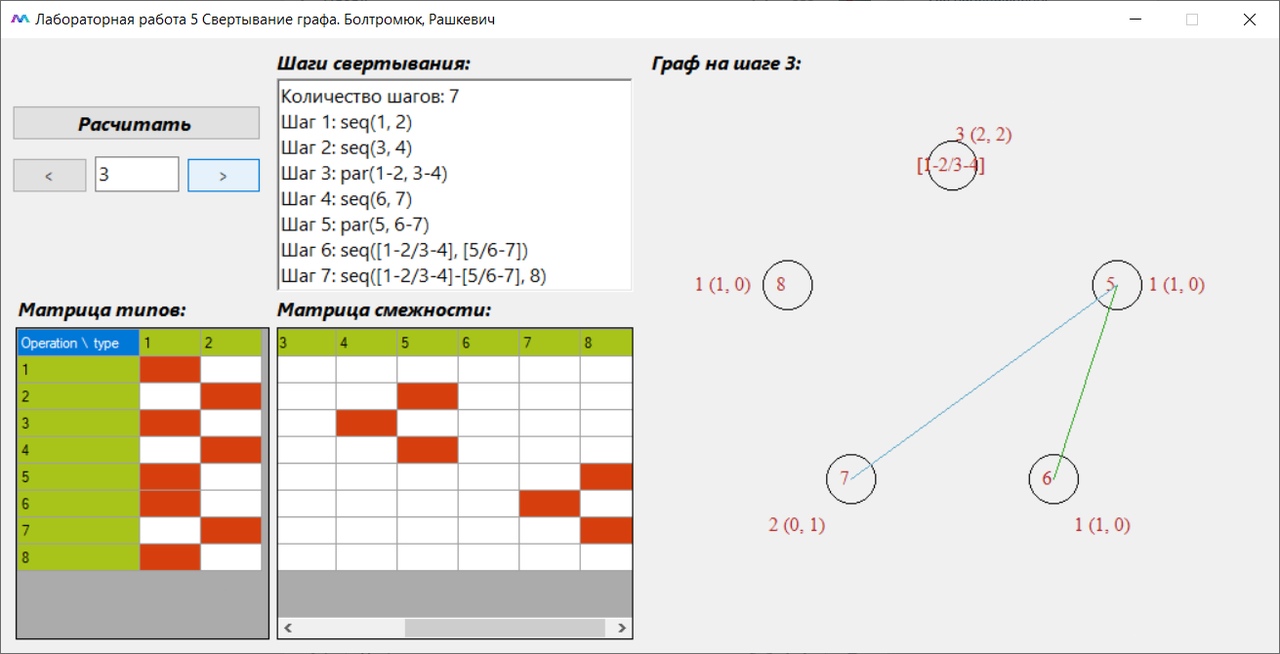


Рисунок 1-Реализация свертывания графа шаг4

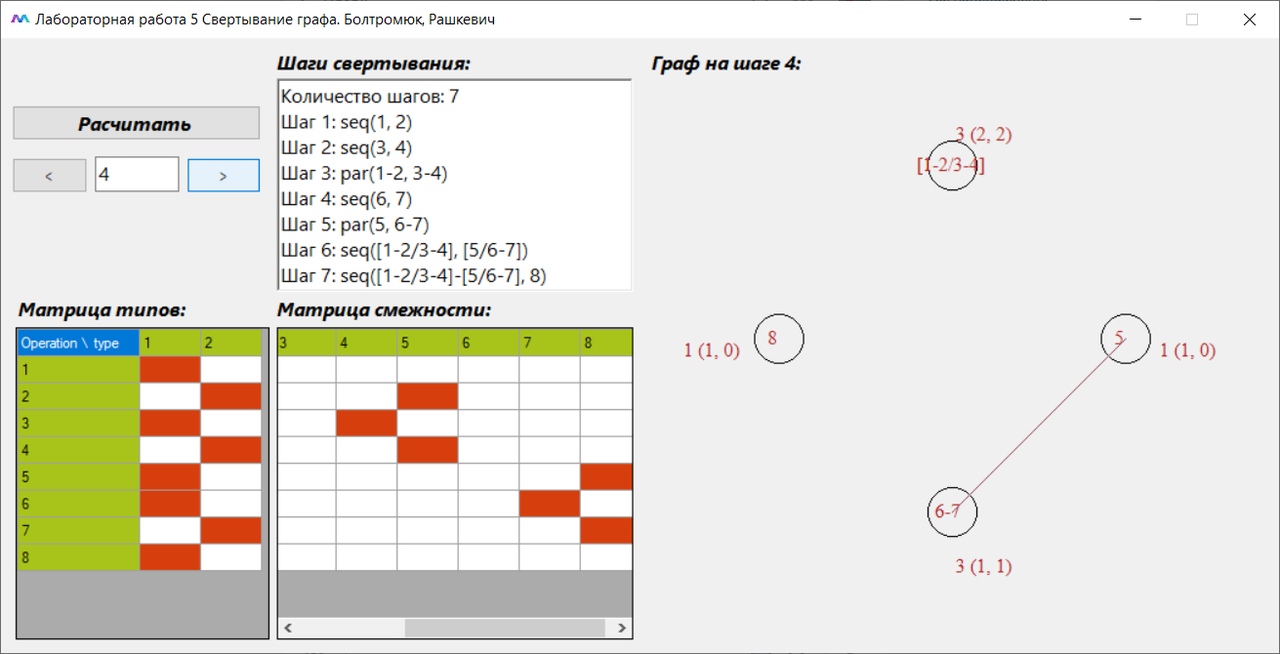


Рисунок 1-Реализация свертывания графа шаг5

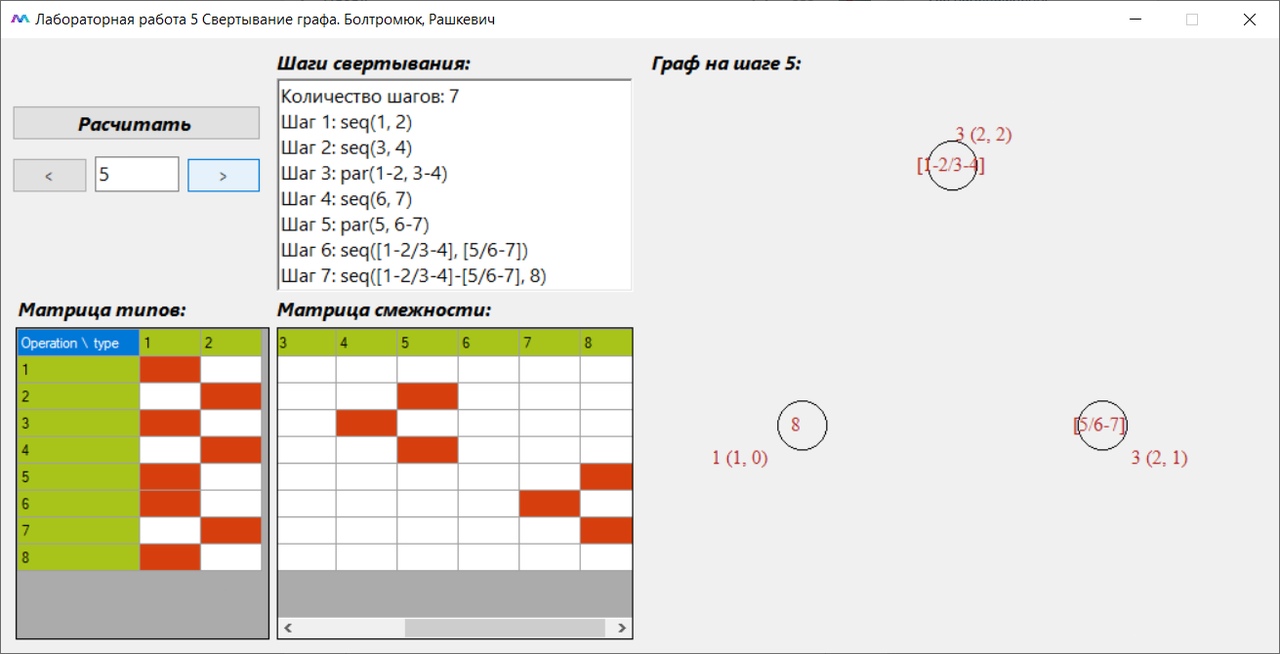


Рисунок 1-Реализация свертывания графа шаг6

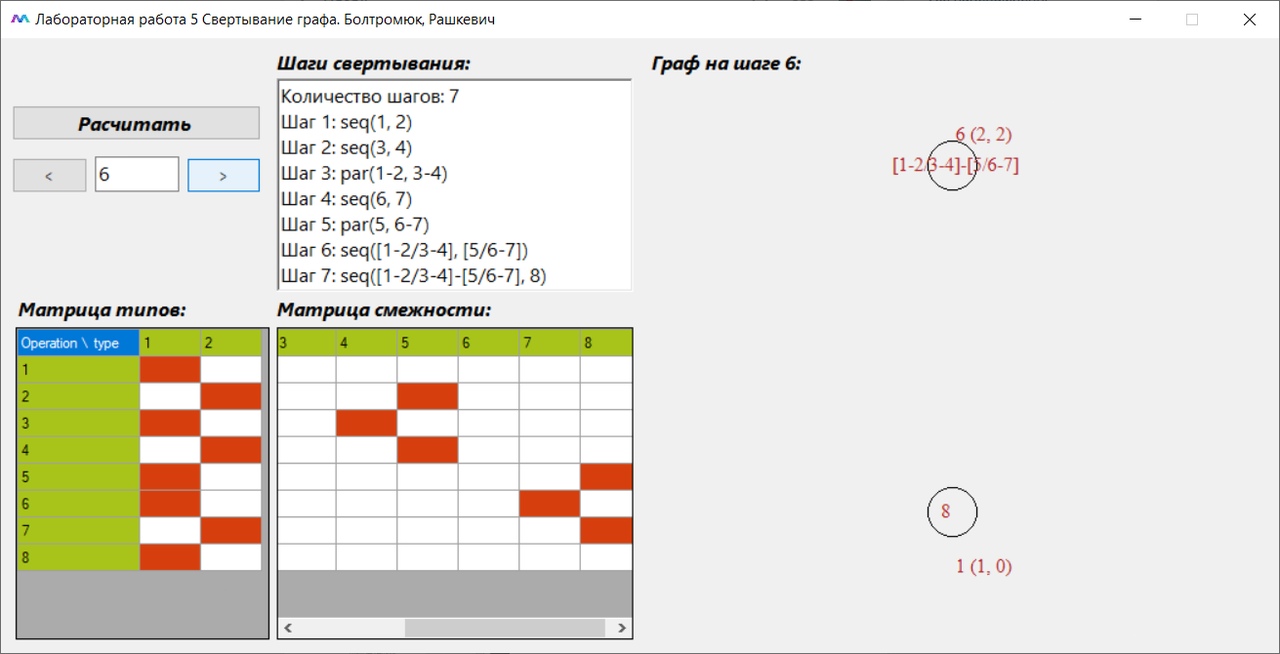


Рисунок 1-Реализация свертывания графа шаг7

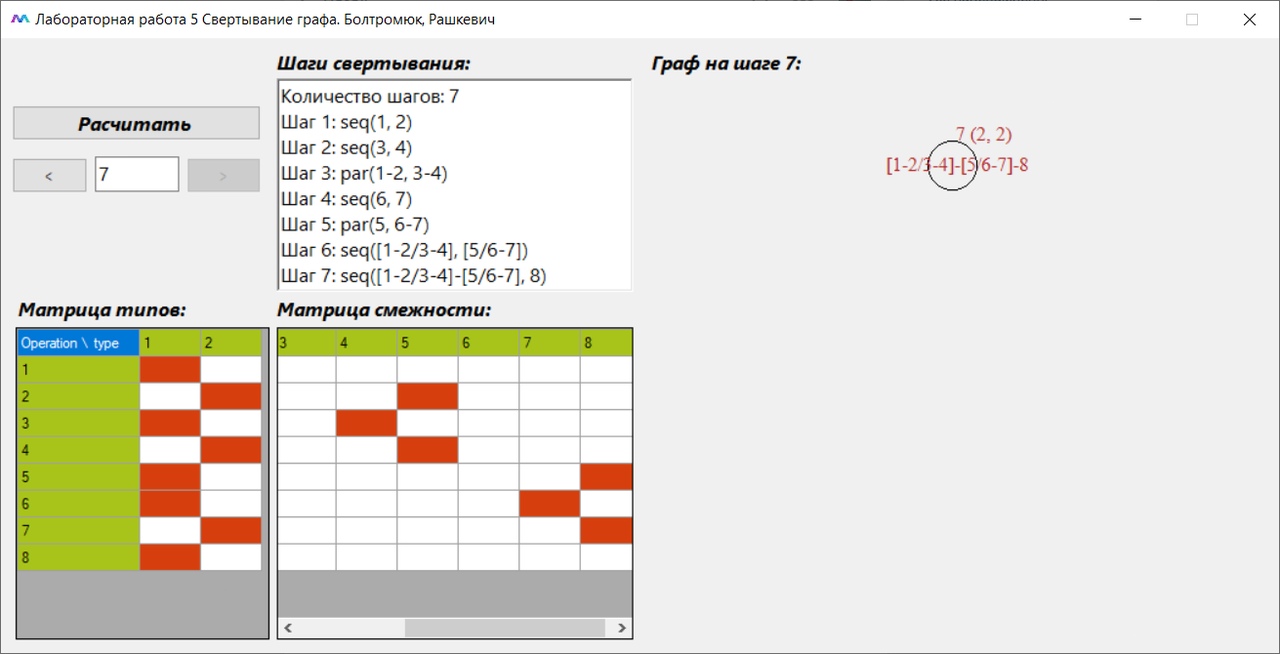


Рисунок 1-Реализация свертывания графа шаг8

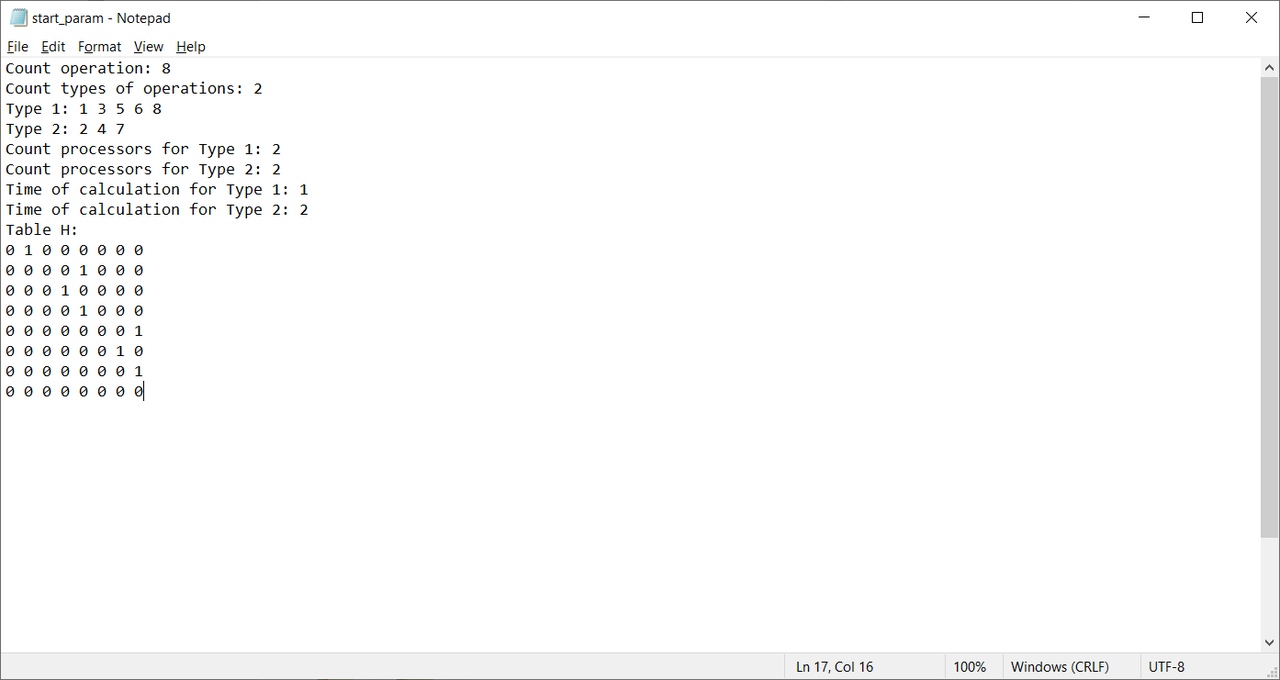


Рисунок 3 – Входные параметры

**Вывод:**

В ходе прохождения лабораторной роботы мы узнали, что из себя представляет алгоритм свертывания графа, так же реализовали его на практике