Министерство образования Республики Беларусь

Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

Факультет информационных технологий и управления

Кафедра интеллектуальных информационных технологий

**Лабораторная работа №2**

по дисциплине

«Модели решения задач в интеллектуальных системах»

Вариант 9

Выполнили:

студенты гр. 121702 Мойсевич А. В.

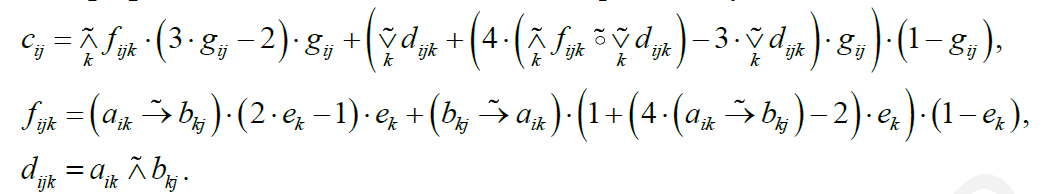
Проверил: Ивашенко В. П.

Минск, 2024

**Тема**: Реализация модели решения задач в интеллектуальных системах.

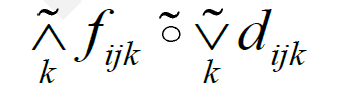
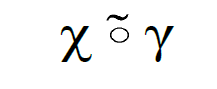
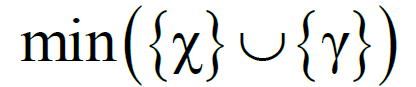
**Цель:** Реализовать и исследовать модель решения на ОКМД архитектуре задачи вычисления матрицы значений.

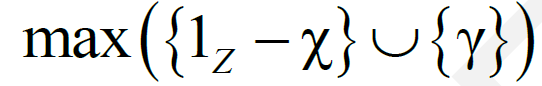
**Дано:** Сгенерированные матрицы A, B, E, G заданных размерностей pxm, mxq, 1xm, pxq соответственно со значениями в рекомендуемом диапазоне [-1;1].





Согласно варианту:

 :  =

=



**Задание:** Получить матрицу значений С, соответствующую размерности *p*x*q.*

**Дополнительные теоретические сведения:**

*Коэффициент ускорения:* , где:

–время решения задачи на первой(последовательной) архитектуре;

– время решения задачи на другой (параллельной архитектуре).

*Эффективность вычислительных систем:*  , где:

– количество процессорных элементов в системе (совпадает с количеством этапов конвейера).

*Коэффициент расхождения:* , где:

— время (суммарная длина программы) решения задачи на (параллельной) архитектуре.

— среднее время (средняя длина программы) решения задачи на той же архитектуре.

**Описание программы:**

Программа позволяет пользователю задавать следующие параметры:

1. Параметры матриц – p, q, m;
2. Время выполнения операций сложения, вычитания, умножения, сравнения;
3. Количество процессорных элементов n.

Программа включает в себя следующие методы:

1. Арифметические:
   1. sum(a, b)
   2. mult(a, b)
   3. diff(a, b)
   4. compare(a, b)

Вычисляют сложение, умножение, разность и сравнение двух величин, увеличивая счетчик вызова соответствующей функции на 1.

1. Утилитарные:
   1. check\_input(str)
   2. print\_matrix(matr, name)
   3. fill\_matrix(m,p,q)

Валидируют ввод, заполняют и выводят матрицы.

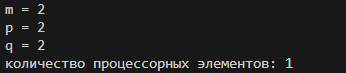
1. Методы расчета значений искомой матрицы:
   1. find\_impl(u,v) – находит импликацию, используя формулу по варианту (max(x, y)).
   2. find\_compose(f, d) – находит композицию, используя формулу по варианту (min(x, y)).
   3. find\_kf(i, j) – находит конъюнкцию из по формуле и считает изменение при нахождении и дизъюнкции .
   4. find\_dd(i, j) – находит дизъюнкцию из по формуле и считает изменение при нахождении и дизъюнкции .
   5. find\_cij(i, j) – находит , используя функции find\_kf(i, j), find\_dd(i, j), find\_compose(f, d) для композиции конъюнкции и дизъюнкции . Считает изменение
   6. find\_C(x, y, m) содержит в себе все перечисленные выше функции в качестве подфункций. Вычисляет импликации  и , находит ,  и их композицию, по формуле: 

находит каждый элемент матрицы .

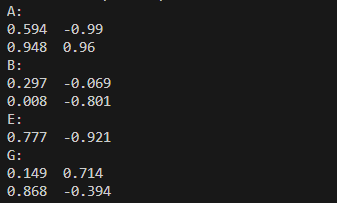
* 1. main() содержит цикл ввода, прекращающийся когда введенные пользователем значения окажутся корректными, инициирует заполнение матриц, нахождение матрицы С и их вывод, отвечает за подсчет большей части показателей.

**Тестирование программы:**

Введем необходимые параметры. Время выполнения каждой операции по умолчанию равно 1.

****

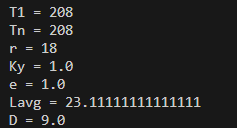
Сгенерированные матрицы A, B, E, G:



Итоговая матрица С:



Вычисленные параметры:



Проверка выполнения:

= min(, )

= min(, ) = 0.297

= min(, ) = -0.99

 = 1 - (1 - )(1 - ) = -0.39897

= max( *1 -* , ) \* (2 - 1) \* + max( *1 -* , ) \* (1 + (4\*max( *1 -* , ) - 2))\*(1 - )

= max( *1 -* , ) \* (2 - 1) \* + max( *1 -* , ) \* (1 + (4\*max( *1 -* , ) - 2))*(1 - ) = 0.406*\*0.554\*0.777 + 0.703\*(1+0.777(4\*0.406 - 2))\*(1- 0.777) = 0.28573457111

= max( *1 -* , ) \* (2 - 1) \* + max( *1 -* , ) \* (1 + (4\*max( *1 -* , ) - 2))\*(1 - ) = 1.99\*(-2.842)\*(-0.921) + 0.992\*(1-0.921(4\*1.99 - 2))\*(1+0.921) = 5.20878918 - 8.5546869491 = -3.3458977691

 = \* = -0.95603866403

=  \* \* (3 - 2) + ( + \* (4 \* min(, ) - 3\*) )\*(1 - )

=  \* \* (3 - 2) + ( + \* (4 \* min(, ) - 3\*) )\*(1 - ) = -0.95603866403 \* 0.149\*(3\*0.149 - 2) + (-0.39897 + 0.149\*(4\*( -0.95603866403) - 3\*(-0.39897))\*(1-0.149) = -0.4514309864146989

Программа вычисляет значения правильно.

**Графики:**

График зависимости коэффициента ускорения от r:

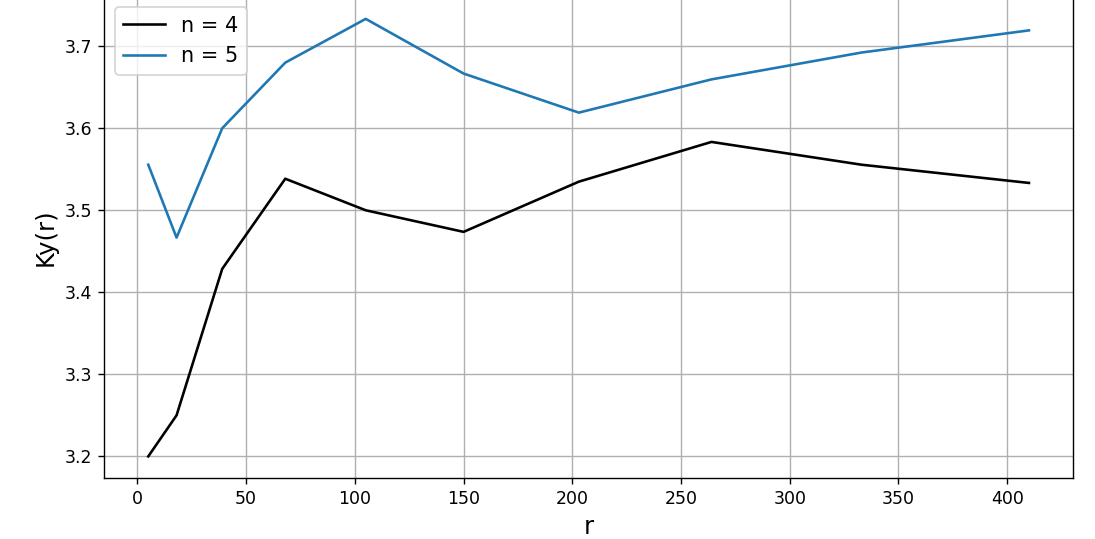


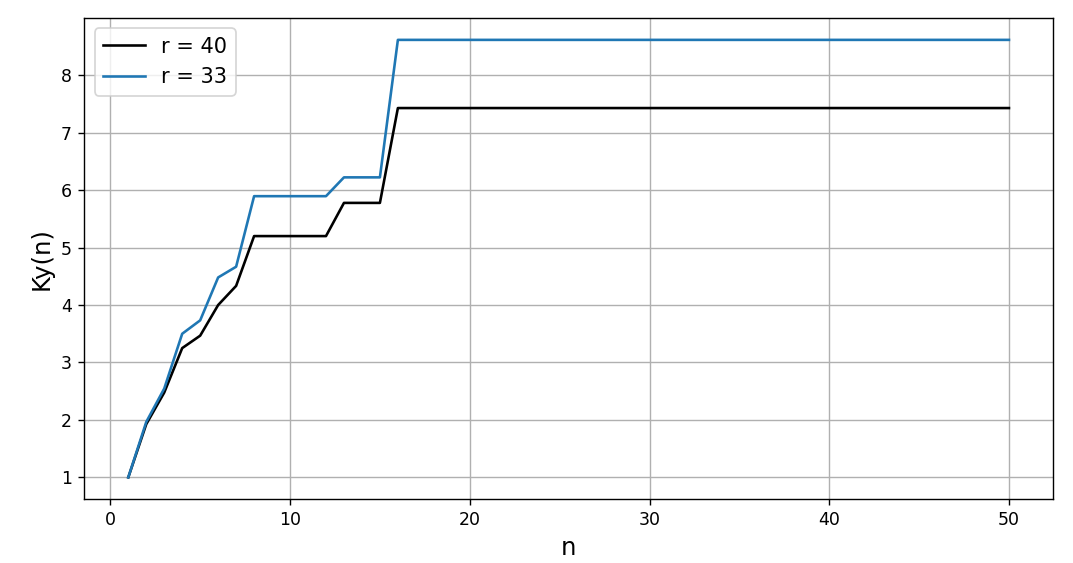
График зависимости Ку от n при r = 40:

График зависимости эффективности e от r:

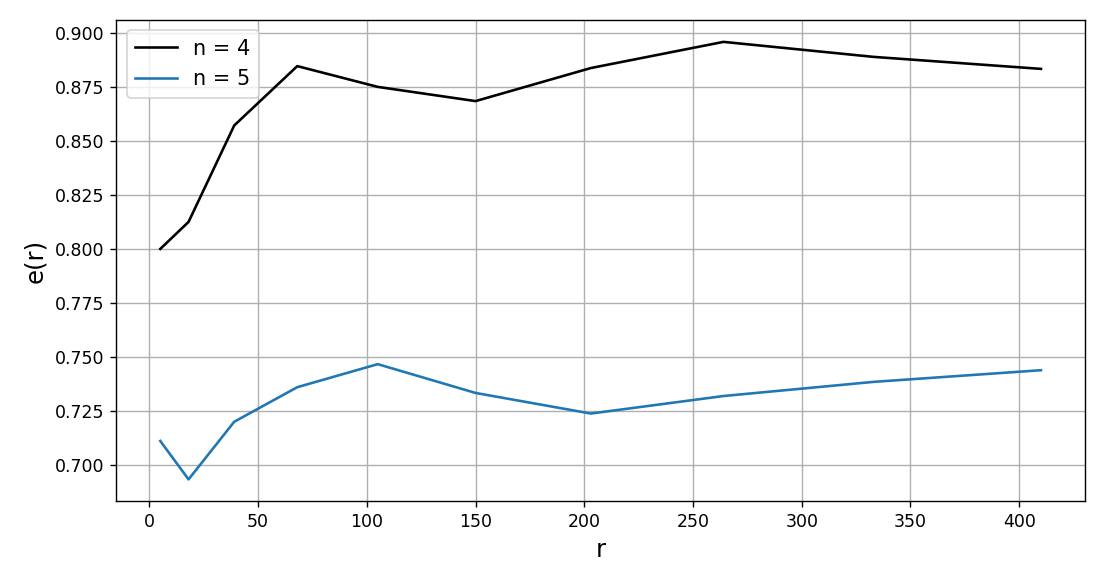


График зависимости е от n при r = 40:

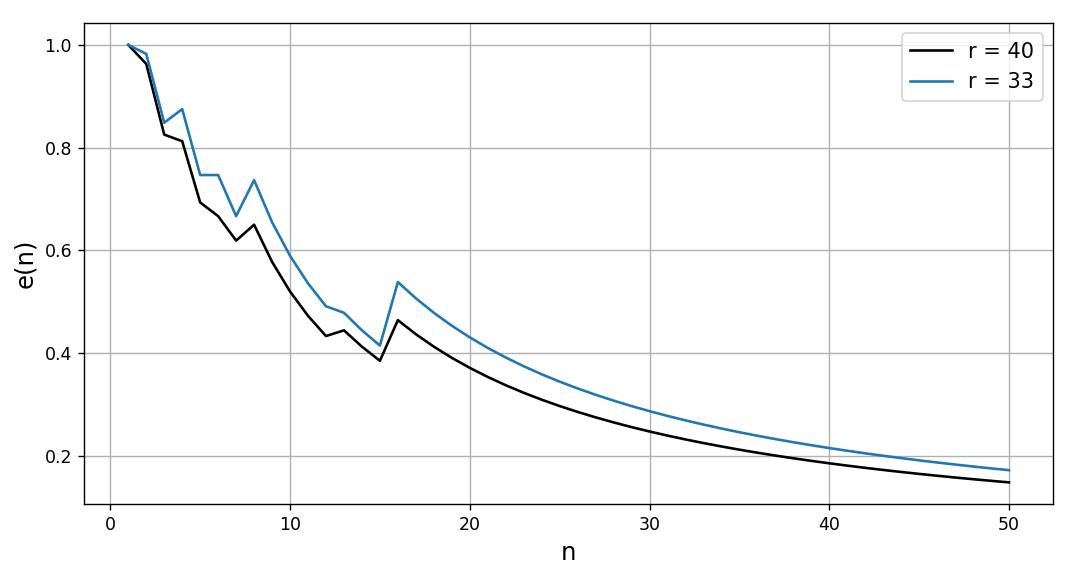
****

График зависимости коэффициента расхождения D от r:

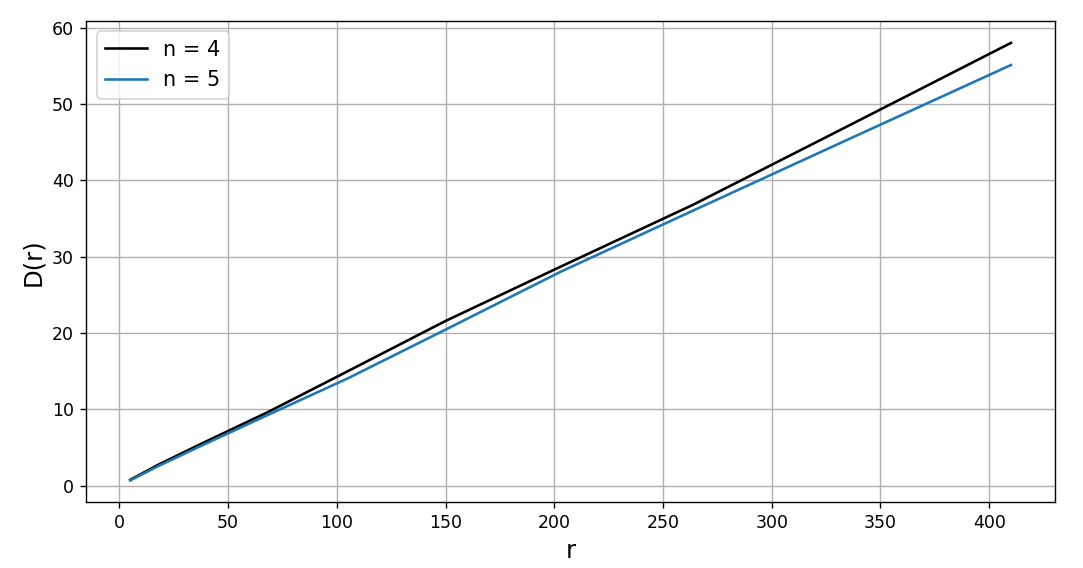
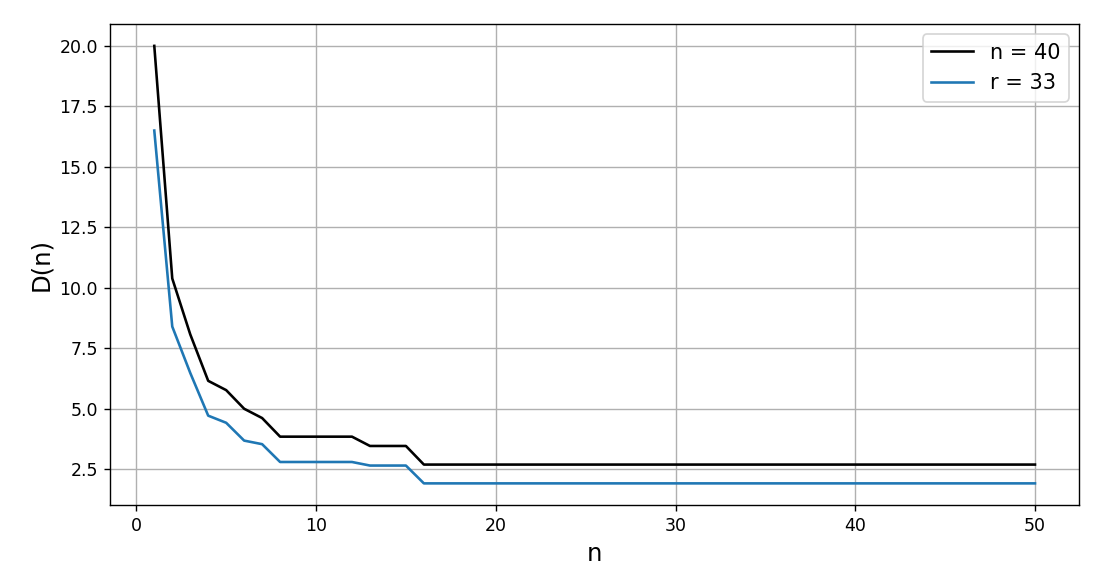
****

График зависимости коэффициента расхождения D от n:

****

**Вопросы:**

1. *Объяснить асимптоты на графиках:*

На асимптотой является y, равный значению процессорных элементов.

На графике видно наличие горизонтальной асимптоты, т.к в какой то момент график перестает изменяться вовсе.

Асимптотами графиков являются 0 и 1 соответственно.

Для графиков существуют горизонтальные асимптоты, при этом чем меньше ранг, тем меньше асимптота.

График асимптот не имеет.

2. *Спрогнозировать, как изменится вид графиков при изменении параметров модели*:

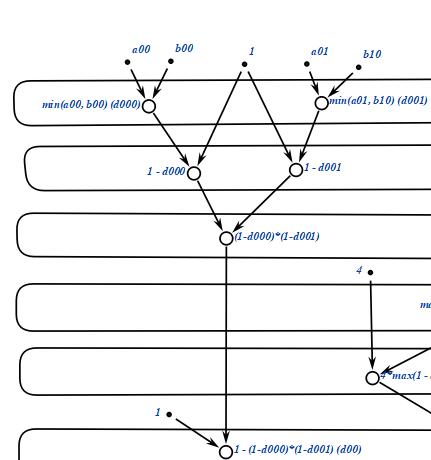
График (r) при увеличении ранга задачи растет, и точки приближаются к горизонтальной асимптоте. График (n) при увеличении количества процессорных элементов растет до определенного момента. Далее он перестает изменяться и на графике образуется плато.

График e(r,n): при увеличении количества процессорных элементов значение эффективности уменьшается, а при увеличении ранга задачи – к асимптоте = 1.

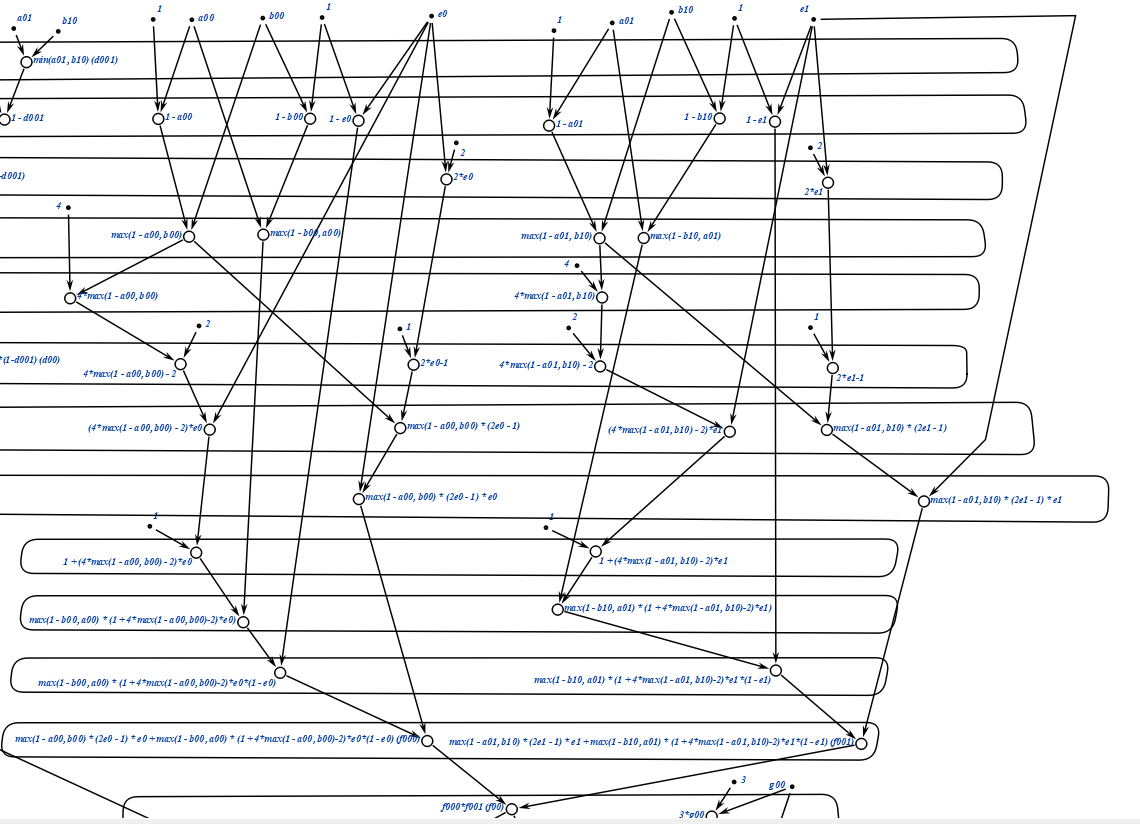
Значение D(r,n) при увеличении ранга бесконечно растет, при увеличении количества процессорных элементов график падает.

**Информационный граф**

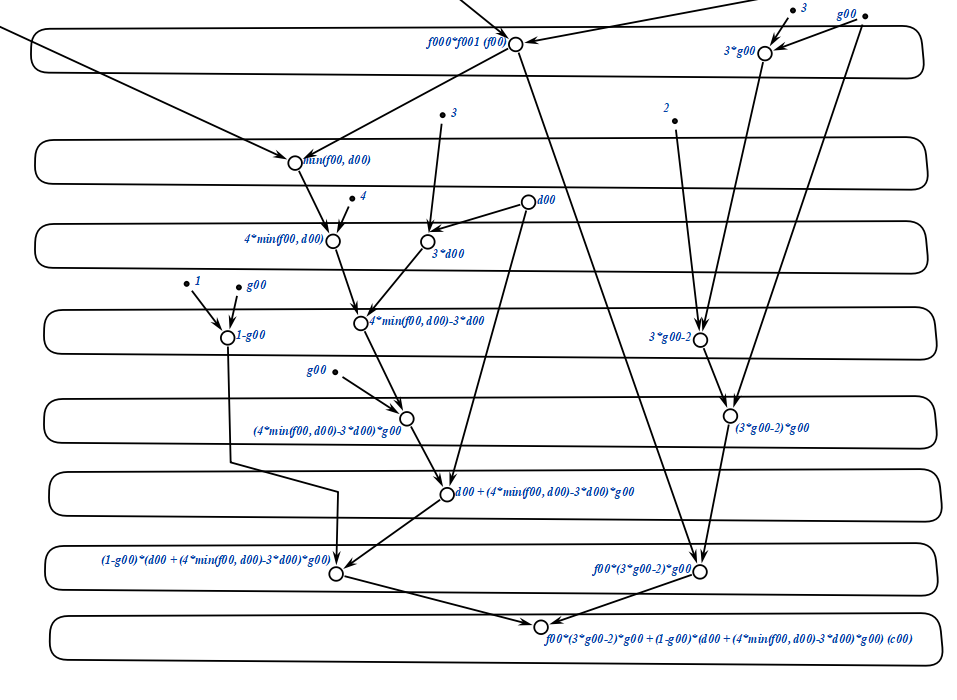
Фрагмент информационного графа для вычисления :

****

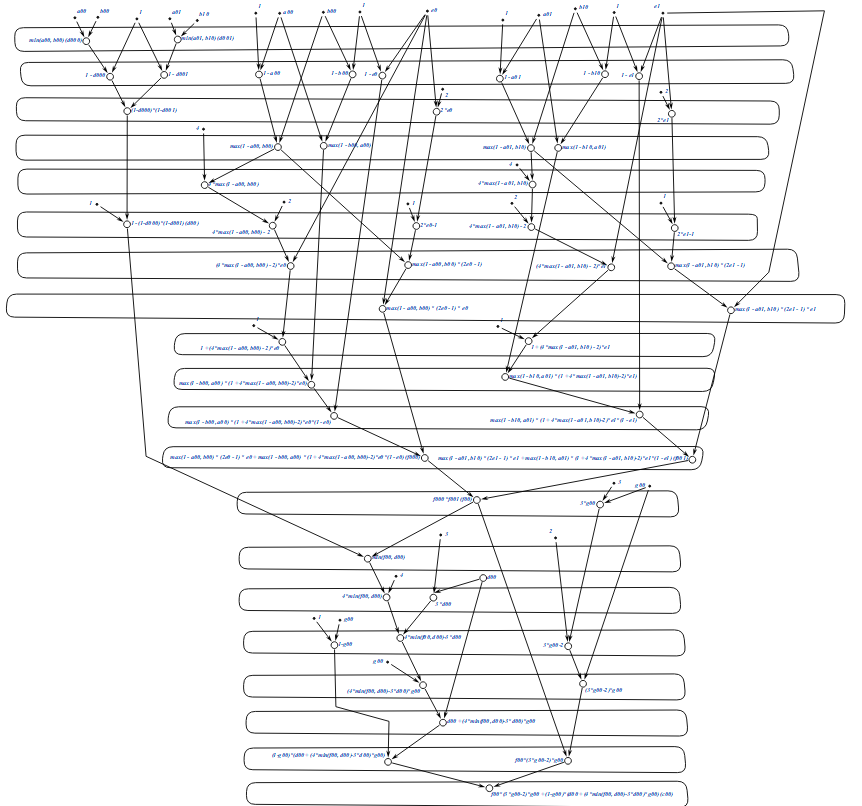
Фрагмент информационного графа для вычисления :



Фрагмент информационного графа для вычисления :



Фрагмент информационного графа с ярусами для вычисления :



**Вывод:** в результате выполнения лабораторной работы была реализована и исследована модель решения на ОКМД архитектуре задачи вычисления матрицы значений. Была выполнена отладка и тестирование модели, построен информационный граф. Были созданы и проанализированы графики двух характеристик конвейерной архитектуры: коэффициент ускорения и эффективность.

**Вклад:**

Витковская С. И. – разработка алгоритма поиска матрицы значений, тестирование и отладка системы, составление графиков.

Мойсевич А. В. – расчет показателей, тестирование и отладка системы, составление информационного графа.

**Использованные источники:**

1. Модели решения задач в интеллектуальных системах. В 2 ч. Ч.1: Формальные модели обработки информации и параллельные модели решения задач: учеб.-метод. пособие/ В. П. Ивашенко. – Минск : БГУИР, 2020. – 79 с.
2. Методы и алгоритмы обработки данных [Электронный ресурс]. Режим доступа: https://studref.com/636037/ekonomika/vychislitelnyy\_konveyer (дата обращения 05.03.2024).