

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 курсу «МРЗвИС»  
на тему:  
«Реализация модели решения задачи на ОКМД архитектуре»**

Выполнили  
студенты группы  
821701:  
Козлов К.А  
Пашкевич В.Л

Проверили:  
Орлова А.С.  
Крачковский Д.Я.

Минск 2020

**Цель:**

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

**Задание:**

Дано: сгенерированные матрицы  $A$ ,  $B$ ,  $E$ ,  $G$  заданных размерностей  $p \times m$ ,  $m \times q$ ,  $1 \times m$ ,  $p \times q$  соответственно со значениями в рекомендуемом диапазоне  $[-1;1]$ .

$$c_{ij} = \wedge_k f_{ijk} * (3 * g_{ij} - 2) * g_{ij} + \left( \vee_k d_{ijk} + \left( 4 * \left( \wedge_k f_{ijk} \circ \vee_k d_{ijk} \right) - 3 * \vee_k d_{ijk} \right) * g_{ij} \right) * (1 - g_{ij})$$

$$f_{ijk} = (a_{ik} \rightarrow b_{kj}) * (2 * e_k - 1) * e_k + (b_{kj} \rightarrow a_{ik}) * \left( 1 + \left( 4 * (a_{ik} \rightarrow b_{kj}) - 2 \right) * e_k \right) * (1 - e_k)$$

$$d_{ijk} = a_{ik} \wedge b_{kj}$$

Получить:  $C$  – матрицу значений соответствующей размерности  $p \times q$ .

**Вариант: 3**

$$\wedge_k f_{ijk} = \prod_k f_{ijk}$$

$$\vee_k d_{ijk} = 1 - \prod_k (1 - d_{ijk})$$

$$\wedge_k f_{ijk} \circ \vee_k d_{ijk} = \min \left( \left\{ \wedge_k f_{ijk} \right\} \cup \left\{ \vee_k d_{ijk} \right\} \right)$$

$$a_{ik} \rightarrow b_{kj} = a_{ik} * (1 - b_{kj}) + 1$$

$$b_{kj} \rightarrow a_{ik} = b_{kj} * (1 - a_{ik}) + 1$$

$$a_{ik} \wedge b_{kj} = a_{ik} * b_{kj}$$

**Исходные данные:**

1.  $n$  – количество процессорных элементов в системе;
2.  $p, m, q$  – размерность матриц;
3.  $t_i$  – время выполнения  $i$  операции над элементами матриц.

**Описание модели:**

Для подсчета времени  $T_1$  подсчитываются количества вызовов различных операций (сумма, разность и т.д.), а затем время одной операции умножается на количество вызовов данной операции, полученные значения суммируются. Для подсчёта  $T_n$  необходимо установить зависимости между выполняемыми операциями. Если операции не являются зависимыми и их можно считать на различных процессорах, то время выполнения такой операции будет  $\frac{c}{n} * t$ , где  $t$  – время выполнения такой операции,  $c$  – количество вызовов данной операции,

$n$  – количество процессоров, на которых выполняется операция. Если операции являются зависимыми, то время выполнения будет  $(t_1+t_2+\dots+t_n)$ , то есть сумма времен зависимых операций. При этом следует учесть, что результат некоторых зависимых операций можно получить, распараллелив её выполнение. Коэффициент ускорения вычислялся по формуле  $K_y = \frac{T_1}{T_n}$ , эффективность по формуле  $e = \frac{K_y}{n}$ . Для подсчета коэффициента расхождения задачи необходимо измерить две характеристики  $Lsum$  и  $Lavg$ , где  $Lsum$ - суммарная длина программы, а  $Lavg$ - средняя длина программы. Так как  $Lsum$ - суммарная длина программы, то она будет равна  $Tn$ . Чтобы посчитать  $Lavg$ , необходимо знать, сколько объектов различных классов выполняется на каком-то из этапов вычислений. Данная задача была решена с помощью подсчета количества вызовов операций и функций на различных ветвях выполнения программой. Зная, количества объектов, выполняющихся на ветвях программы, время выполнения функции или операции, можно подсчитать  $Lavg$ .

## Графики:

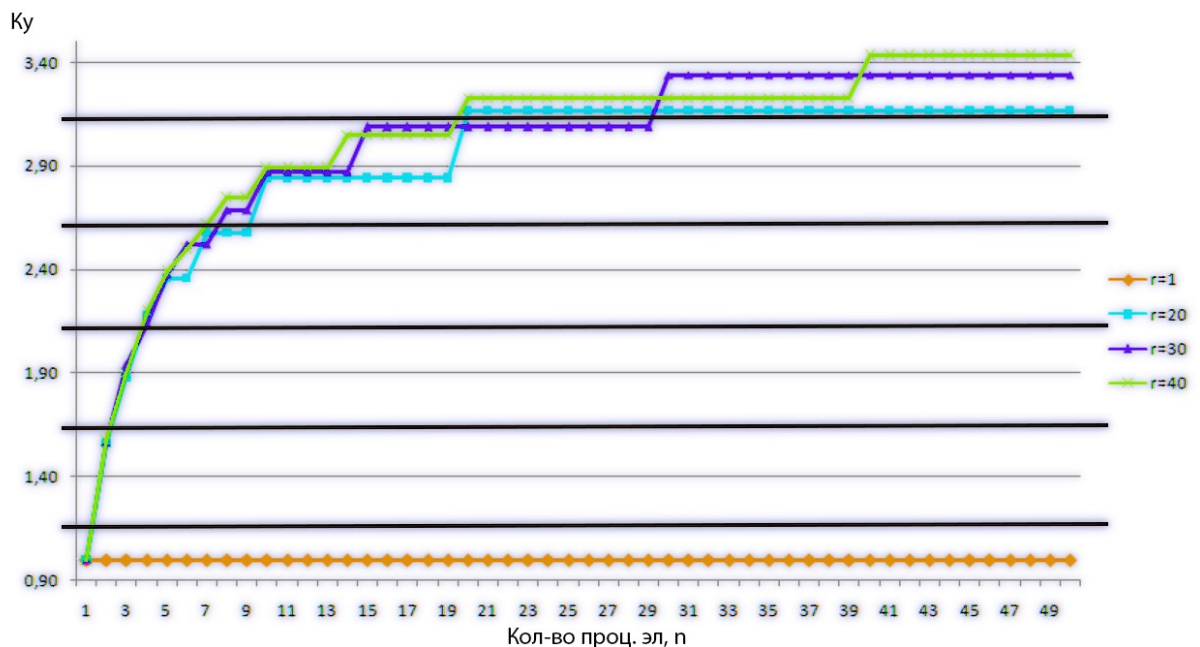


График 1-Ку(n)

К<sub>у</sub>

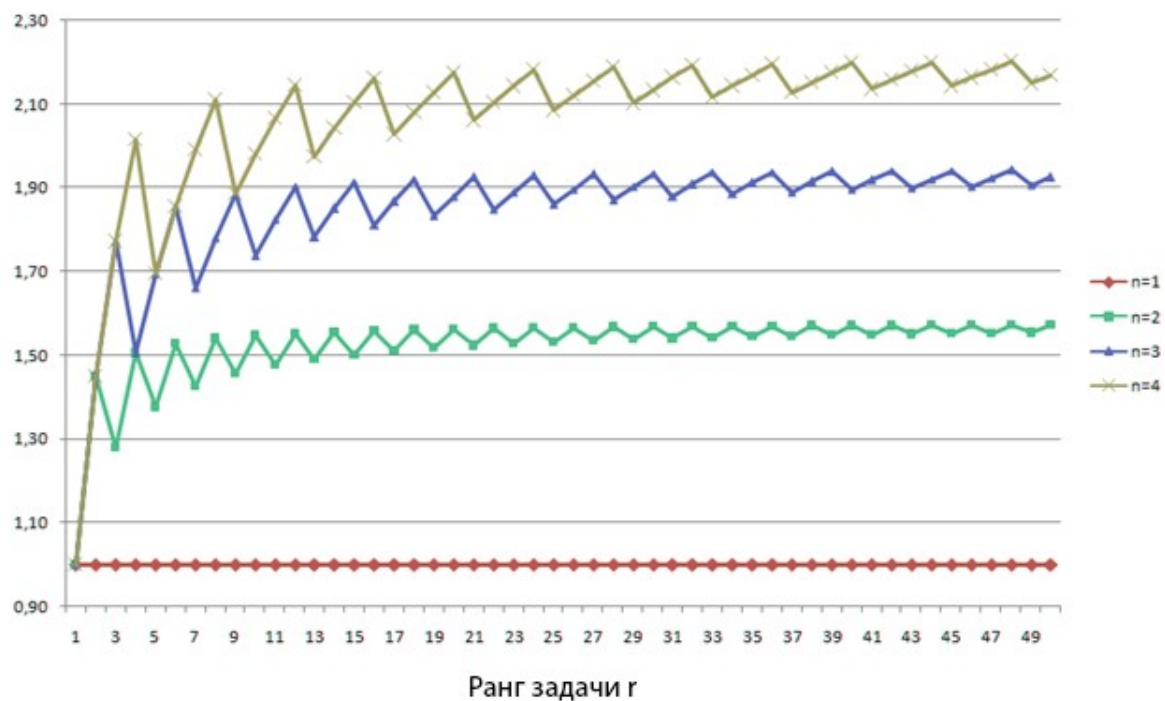


График 2 – К<sub>у</sub>(r)

е

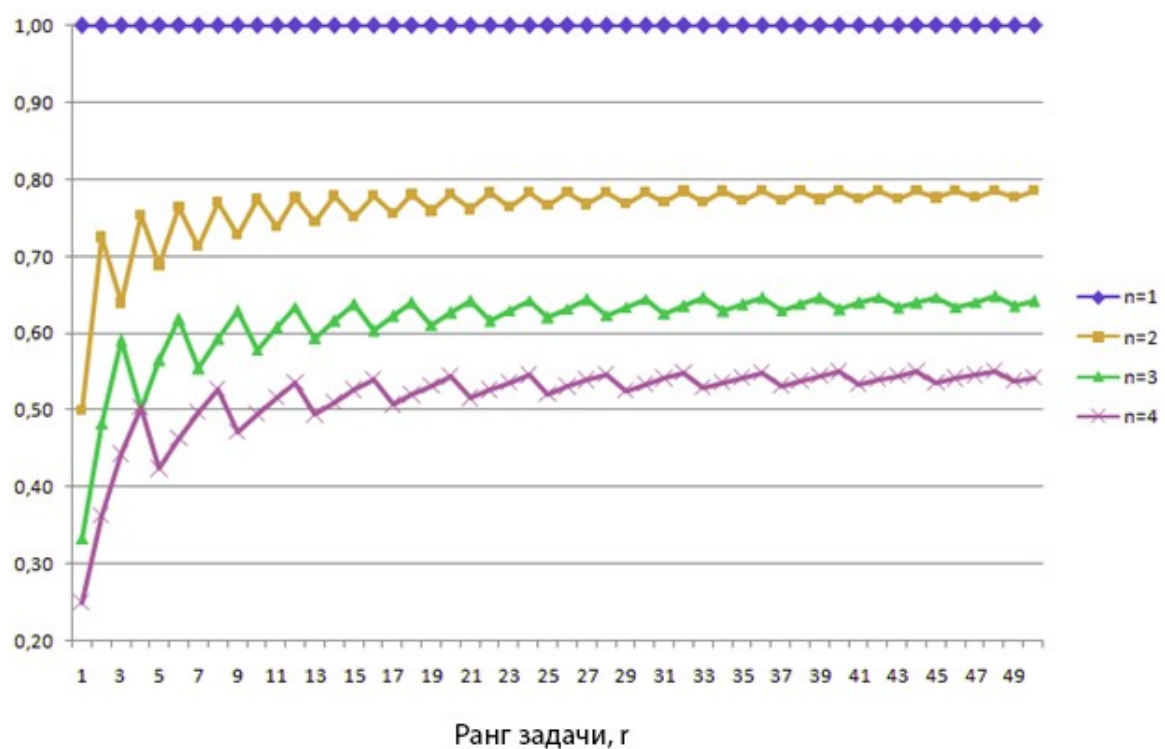


График 3–  $e(r)$

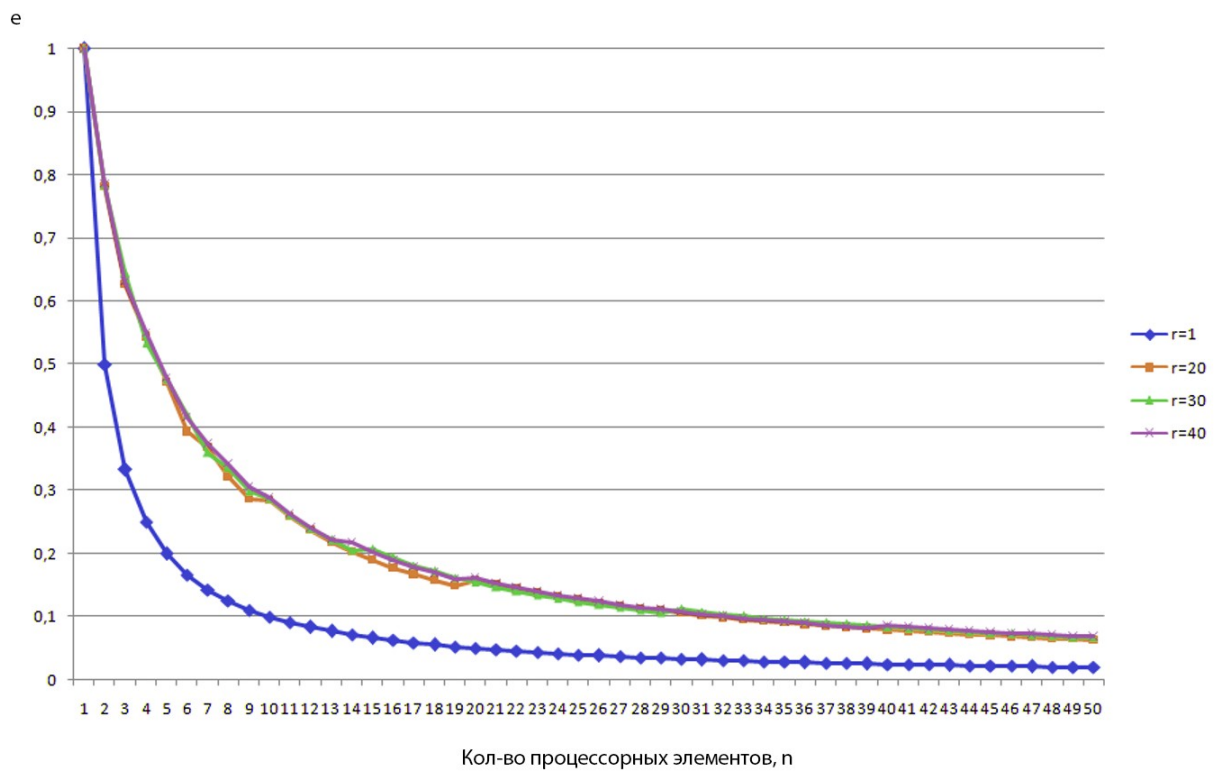


График 4–  $e(n)$

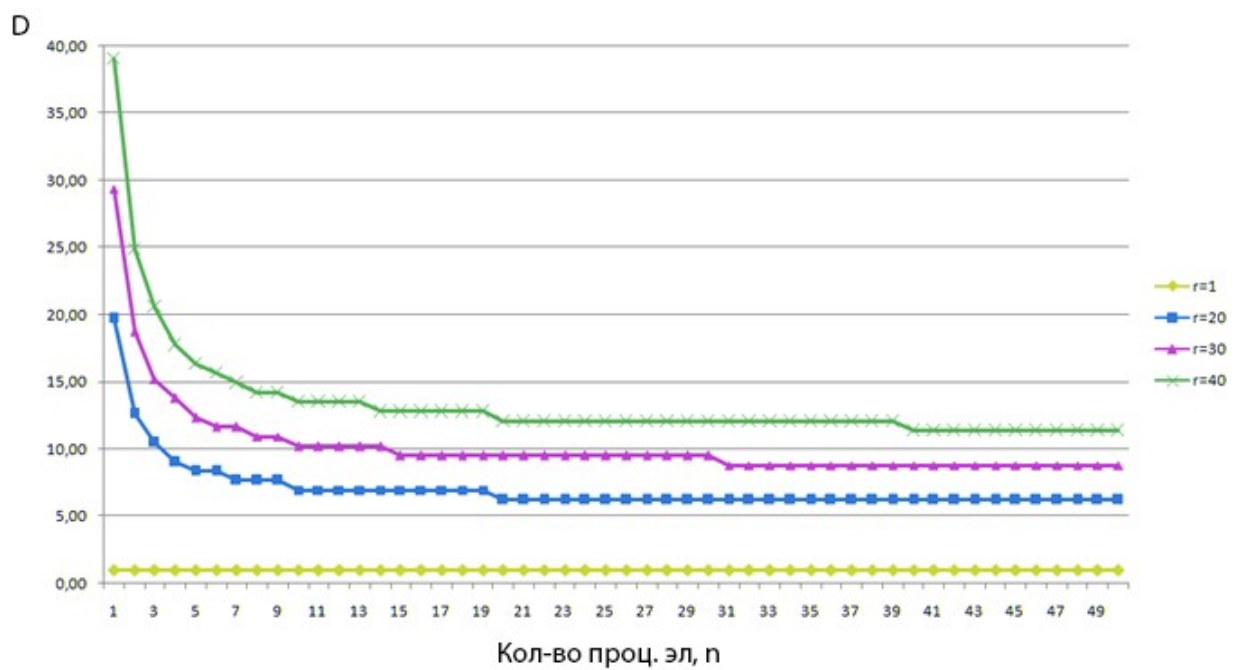
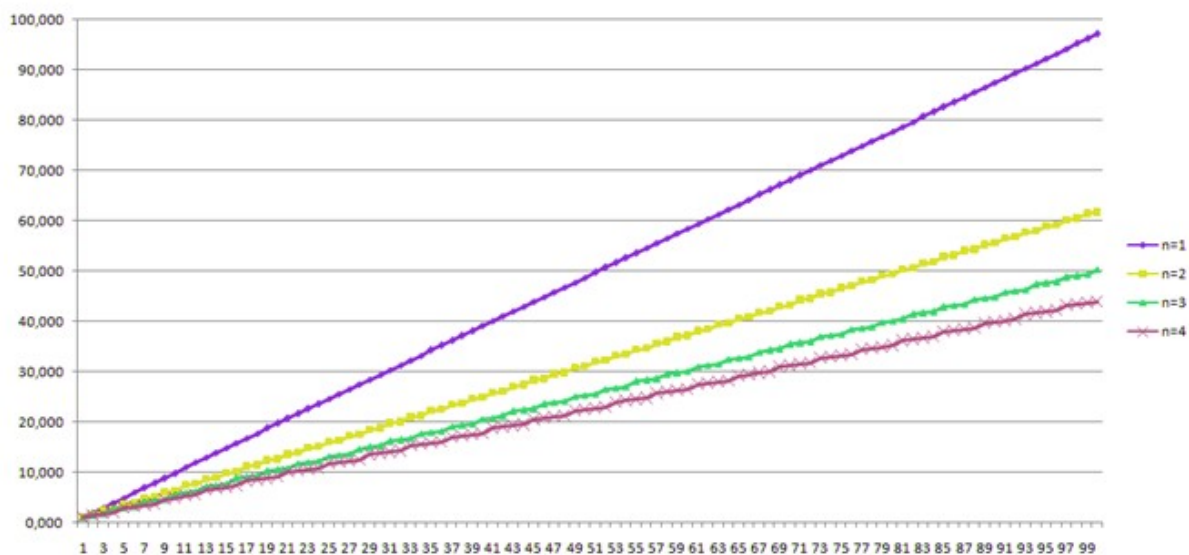


График 5 –  $D(n)$

D



Ранг задачи, r

График 6 –  $D(r)$

### Вопросы:

1. Проверить, что модель создана верно: программа работает правильно;

```
m = 2
p = 2
q = 2
n = 2

additionTime = 1
subtractionTime = 1
multiplicationTime = 1
divisionTime = 1
comparisonTime = 1
```

a  
0.4844648059359882 0.1870469880407546  
0.4375729190532336 -0.8157984271120551

b  
-0.7908016332352128 -0.4839999091682381  
-0.08276351009799598 0.28140885360899115

e  
-0.4938301984010287 -0.9945930568799579

g  
-0.7333640082521657 -0.9253236456432838  
0.5724789431794433 0.8255540161573796

c  
0.5116631971597323 0.16422286674645664  
0.795240749046283 0.4423141082447122

$C[0][0] = 0.511663197 * (3 * 1.1049233315 - 2) * 1.1049233315 + (0.03476914252 + (4 * 0.03476914252 - 3 * 0.03476914252) * 1.1049233315) * (1 - 1.1049233315);$

$C[0][1] = 0.1845210929 * (3 * 0.0000379155 - 2) * 0.0000379155 + (0.1642549091 + (4 * 0.1642549091 - 3 * 0.1642549091) * 0.0000379155) * (1 - 0.0000379155);$

$C[1][0] = 0.12121710761 * (3 * 2.5925759171 - 2) * 0.885244727217088 + (0.423044484 + (4 * 0.423044484 - 3 * 0.423044484) * 2.592575917) * (1 - 2.592575917);$

$C[1][1] = 1.4590469622 * (3 * 0.423044484 - 2) * 0.423044484 + (0.6644657911 + (4 * 0.6644657911 - 3 * 0.6644657911) * 0.0854107100867727) * (1 - 0.423044484).$

Модель создана верно.

## 2. Объяснить на графиках точки перегиба и асимптоты;

Были построены графики зависимостей  $Ky(n, r)$ ,  $e(n, r)$  и  $D(n, r)$ .

$Ky(n, r)$  – коэффициент ускорения;

$e(n, r)$  – эффективность;

$D(n, r)$  – коэффициент расхождения задачи.

### Асимптоты и точки перегиба графиков:

Асимптотой графика  $Ky(r)$  является прямая, параллельная оси абсцисс, а ордината всех точек этой прямой равна значению  $Ky(r)$ , при  $n=r$ . Точками перегиба являются точки, удовлетворяющие условию:  $r \% n = 0$ . Это объясняется тем, что при  $r \% n = 0$ , все процессорные элементы одновременно задействованы в вычислениях.

Асимптотой графика  $Ky(n)$  является прямая, параллельная оси абсцисс, а ордината всех точек этой прямой равна значению  $Ky(n)$ , при  $n=r$ . Это объясняется тем, что при  $n \geq r$ , будут задействованы только  $r$  процессорных элементов.

Асимптотой графика  $e(r)$  является прямая  $y=1$ . Точками перегиба являются точки, удовлетворяющие условию:  $r \% n = 0$ .

Асимптотой графика  $e(n)$  является прямая  $y=0$ , так как рост функции  $Ky(n)$  ограничен ( $Ky(n)=const$ , при  $n \geq r$ ), а количество процессорных элементов  $n$  продолжает увеличиваться.

Асимптотой графика  $D(n)$  является прямая, параллельная оси абсцисс, а ордината всех точек этой прямой равна значению  $D(n)$ , при  $n=r$ .

Асимптотой графика  $D(r)$  является функция  $y=D(r)$ .

3. Спрогнозировать как изменится вид графиков при изменении параметров модели; если модель позволяет, то проверить на ней правильность ответа;

При увеличении  $n$  значение  $Ky(n)$  растёт, пока  $n < r$ .

При увеличении  $r$  значение  $Ky(r)$  растёт скачкообразно.

При увеличении  $n$  значение  $e(n)$  уменьшается.

При увеличении  $r$  значение  $e(r)$  растёт скачкообразно.

При увеличении  $n$  значение  $D(n)$  снижается, пока  $n < r$ .

При увеличении  $r$  значение  $D(r)$  растёт.

### **Вывод:**

В результате выполнения лабораторной работы была реализована модель решения на ОКМД архитектуре задачи вычисления матрицы значений. Реализованная модель была проверена на работоспособность и правильность получаемых результатов. С помощью графиков были изучены зависимости коэффициента ускорения, эффективности и коэффициента расхождения программы от ранга задачи и количества процессорных элементов.