**РОЗДІЛ 2. РОЗРОБКА ПРОГРАМИ ПРГ1 ДЛЯ ПКС СП**

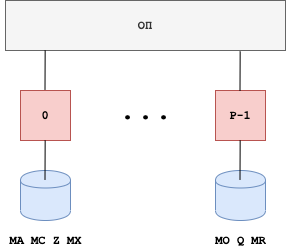
У даному розділі розроблюється програма ПРГ1 для системи зі спільною пам’яттю, що відповідає технічному завданню, представленому на рис. 2.1.

Рис 2.1 Структура ПКС СП

* Мова програмування: Java;
* Математичний вираз: MA = max(Z) ∙ MO + (min(Q) ∙ MC) ∙ (MR ∙ MX).

**2.1 Розробка паралельного математичного алгоритму**

Паралельний математичний алгоритм відповідно до рекомендованої методичної літератури можна подати у вигляді наступних етапів:

1. ai = max(ZH), i = (0, P-1);
2. a = max(a, ai), i = (0, P-1);
3. bi = min(QH), i = (0, P-1);
4. b = max(b, bi), i = (0, P-1);
5. MAH = a ∙ MOH + b∙MCH ∙ (MR∙MXH).

де:

H = N/P;

MAH – H рядків матриці MА;

MOH – H рядків матриці MO;

MCH – H рядків матриці MC;

MXH – H рядків матриці MX;

Спільні ресурси: a, b, MR

**2.2 Розробка алгоритмів процесів**

Оскільки розроблюване програмне забезпечення є масштабованим і працює на системі із кількістю процесорів P ≥ 2, то для реалізації необхідно скласти єдиний алгоритм для всіх задач.

**2.3 Розробка схеми взаємодії процесів**

На основі алгоритму для всіх задач, наведеному в попередньому розділі, були розроблені структурні схеми моніторів для взаємодії задач (рис. 2.2). Вони дозволяють наочно контролювати бар’єри та критичні ділянки.

Для демонстрації взаємодії між задачами обрано три задачі: T(0);

T(i), i = (1, P-2); T(P-1) . Задачі T(0) і T(P-1) вводять дані, тому з ними взаємодіють всі інші (синхронізація по вводу). Задачі T(i), i = (1, P-2) виконують лише обчислення і синхронізуються зі всіма іншими задачами по обчисленню. Задача T(0) виводить результат обчислень.

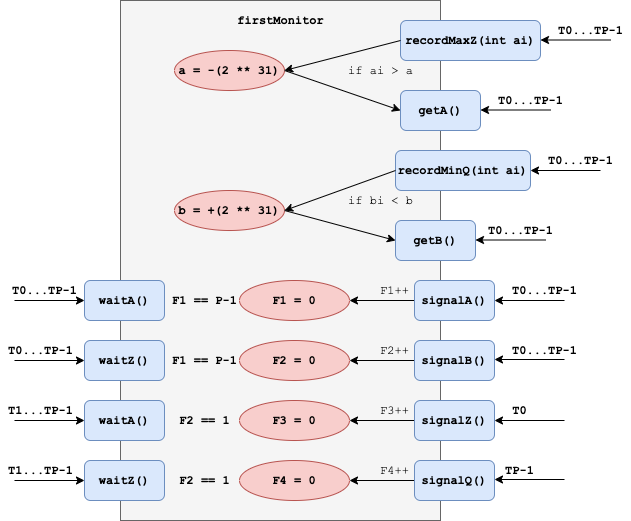
Рис 2.2 Структурна схема першого монітору

Рис 2.3 Структурна схема другого монітору

**2.4 Розробка програми ПРГ2**

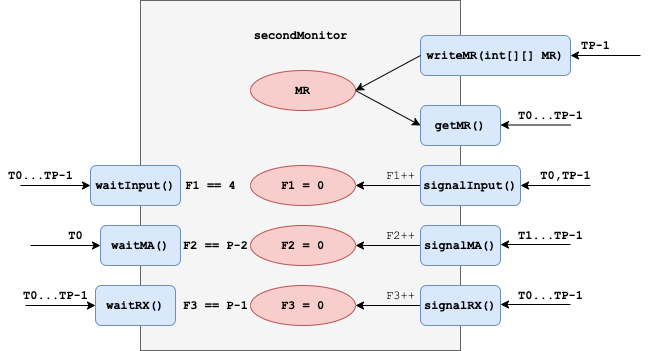
Програма написана на мові Java з використанням допоміжного модулю тестування написаного на мові Python.

В директорії src знаходиться сирцевий код програми.

В директорії operations знаходяться допоміжні інструменти для роботи.

В директорії bin знакодиться скомпільований код програми.

В директорії data знаходяться дані отримані з тестування программи.

Основний клас Main містить декілька функцій.

main – точка входу в програму. Формує ідентифікатори задач, запускає задачі.

calclateFunction - обчислює задану за варіантом ПКС СП.

singleThreadedImplementation - спеціальна реалізація для однопоточної системи.

Класи FirstMonitor та SecondMonitor містять в собі реалізацію моніторів. Класи FirstTask, IntermediateTask, LastTask реалізують логіку взаємодії з монітором.

Лістинг програми ПРГ1 наведено у додатку А.

**2.5 Тестування ПРГ1**

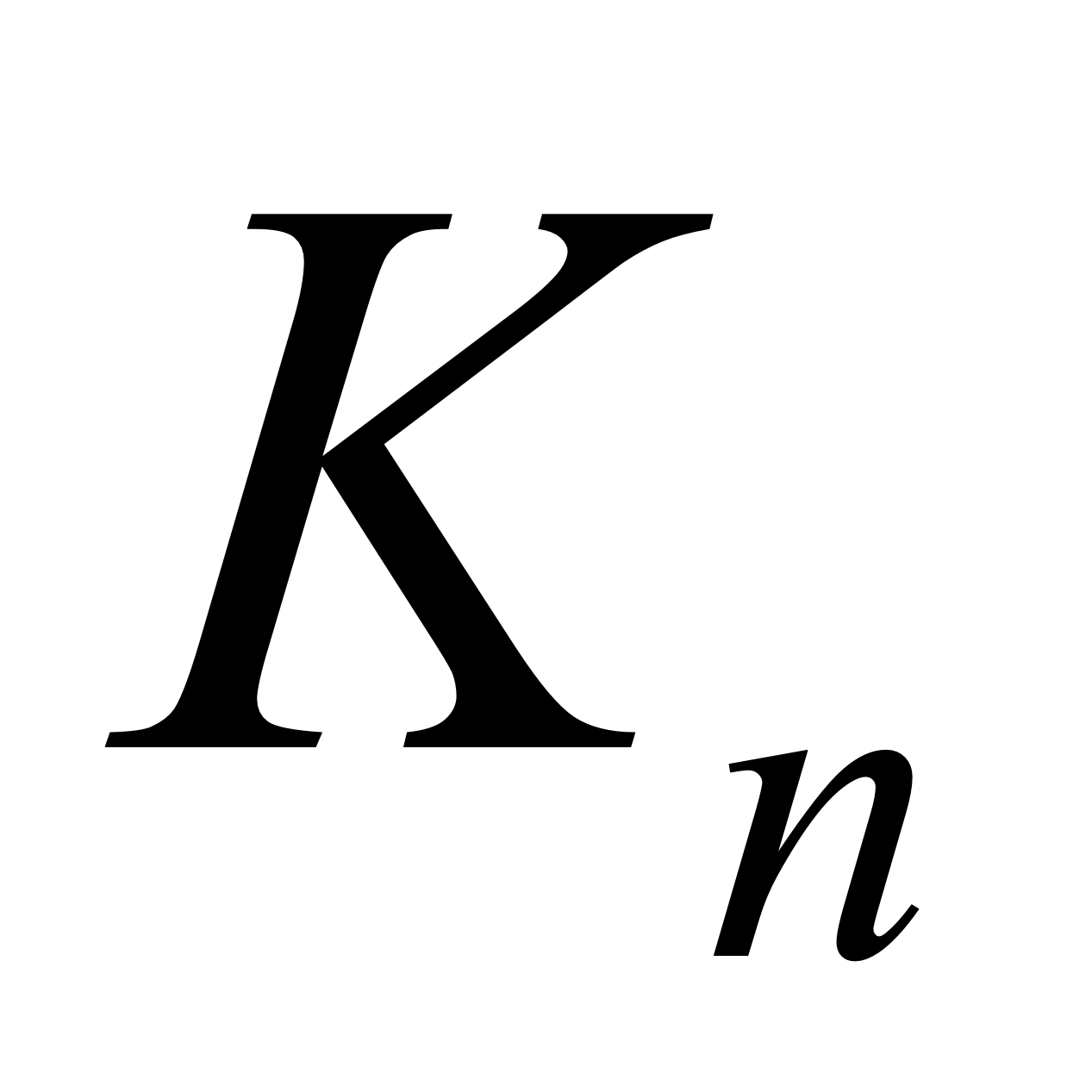
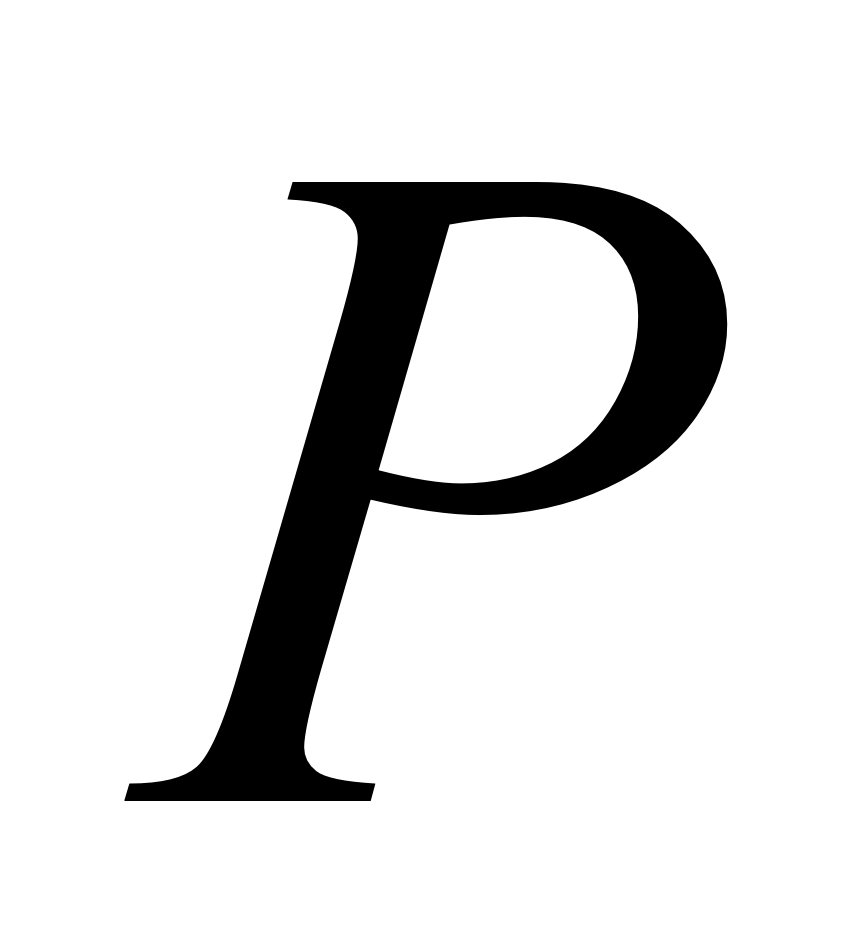
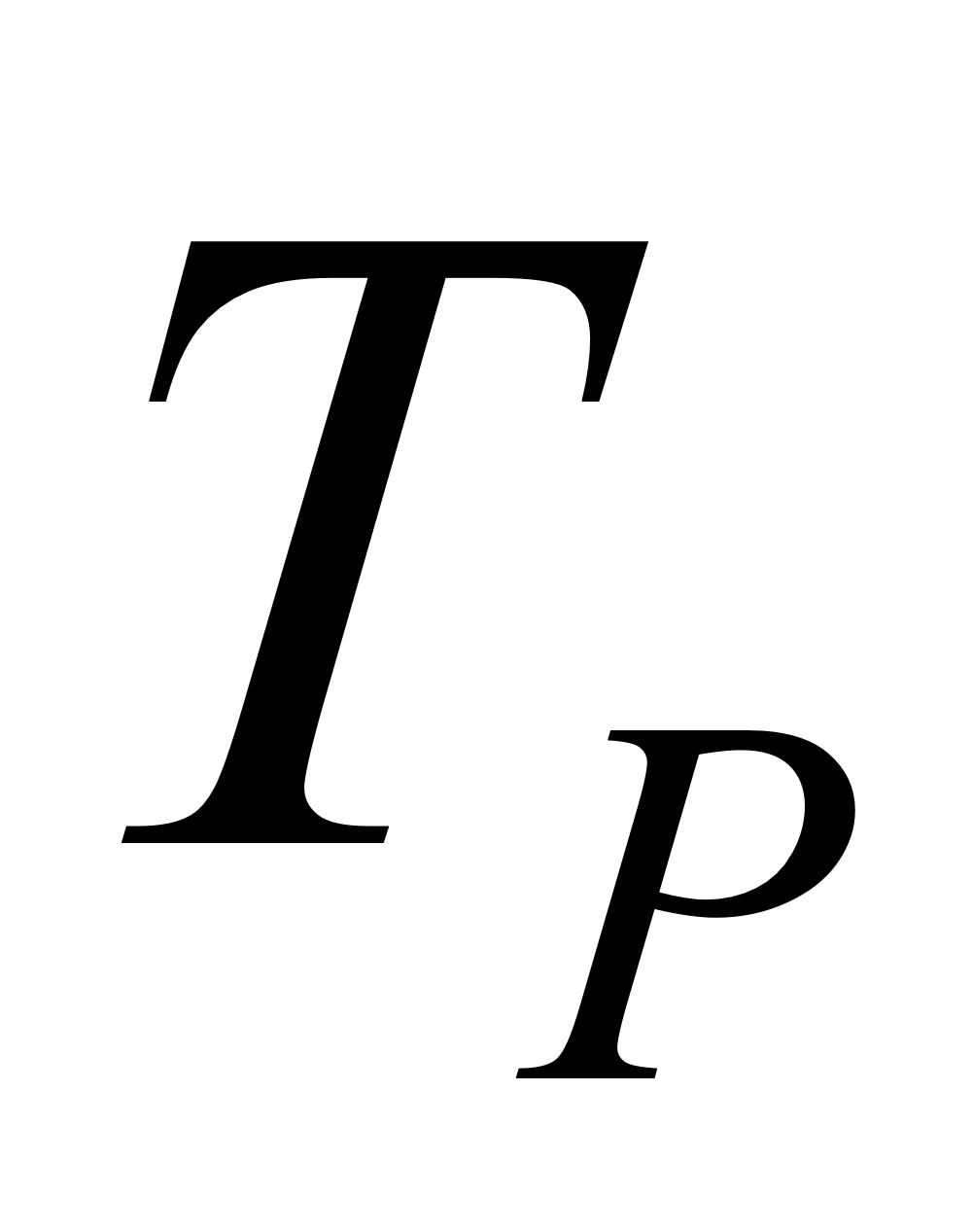
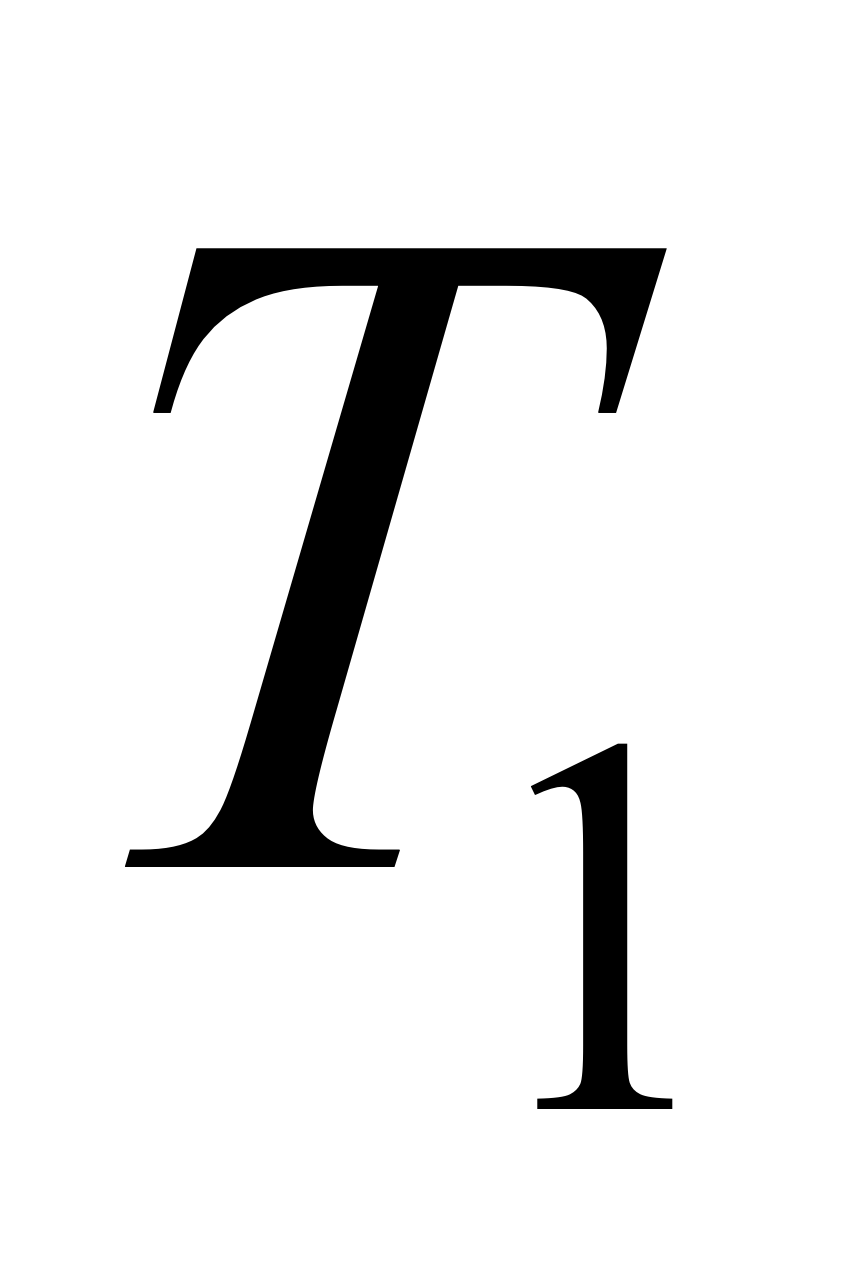
Для тестування використовувалась паралельна обчислювальна система з наступними апаратними характеристиками:

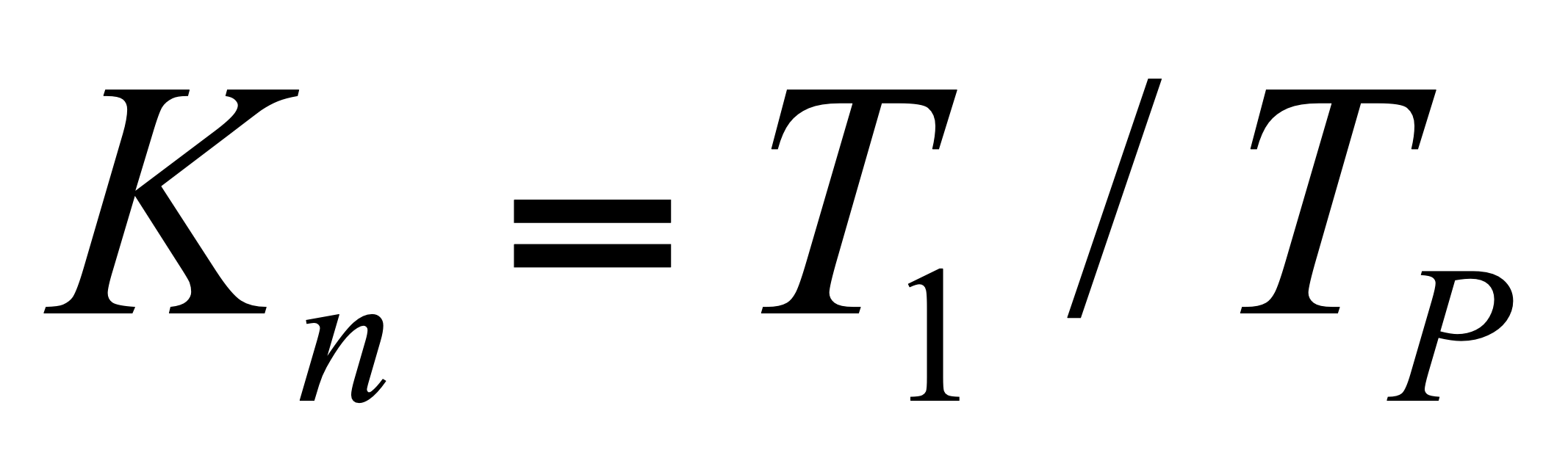
* 2,4 GHz 4‑ядерний процесор Intel Core i5
* 8 ГБ 2133 MHz LPDDR3

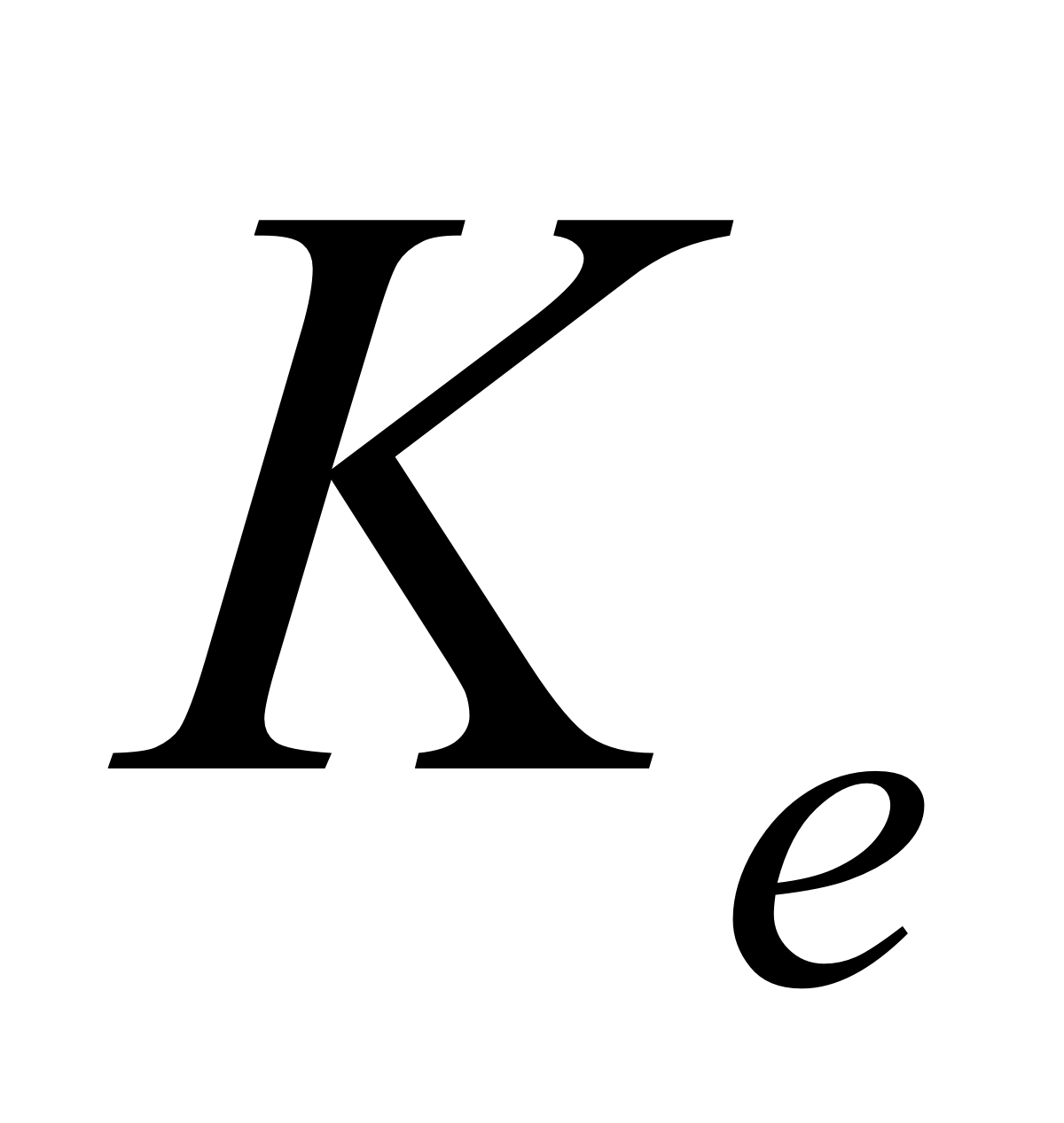
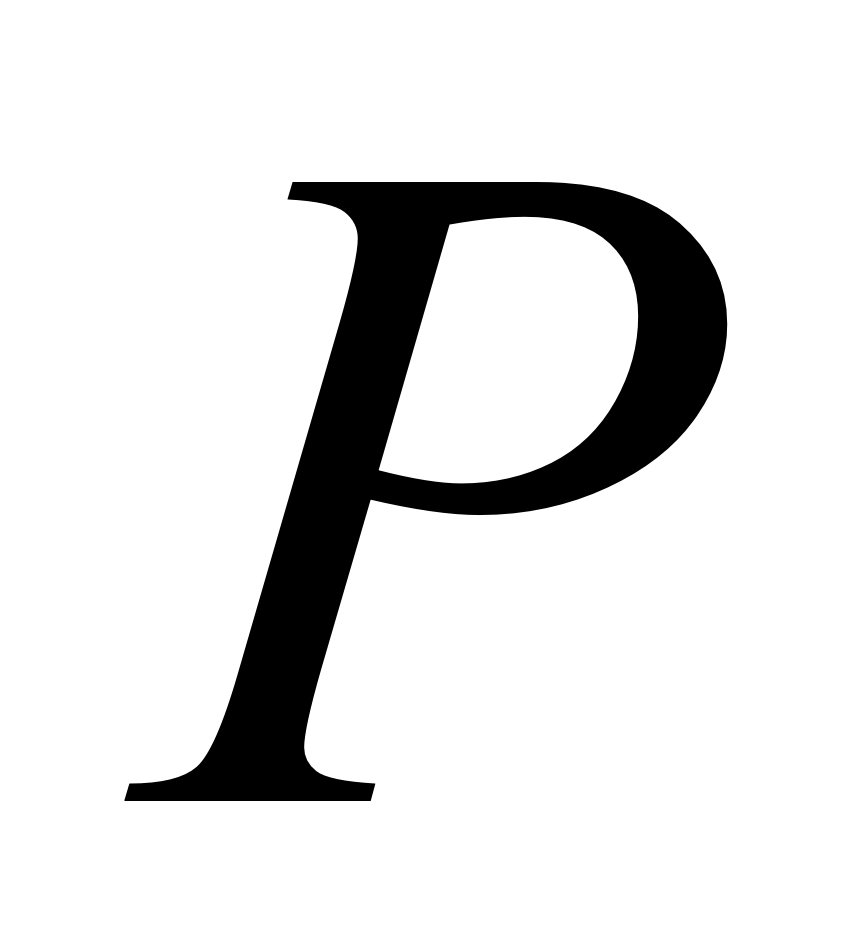
В якості програмного забезпечення виступали:

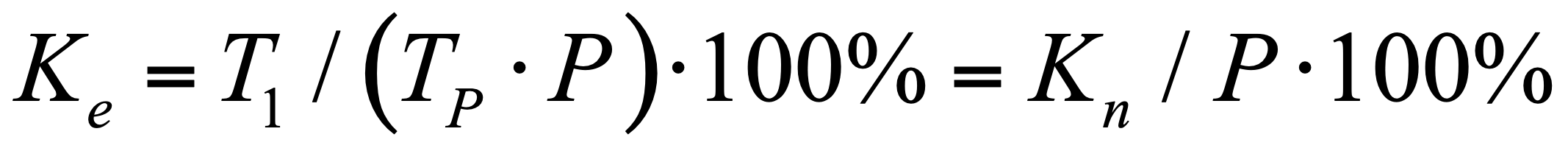
* операційна система:  macOS Catalina 10.15.4
* середовище розробки програм SublimeText3.

Щоб виміряти час виконання окремих потоків використовувалася системна функція currentTimeMillis.Задля автоматичного тестування був розроблений власний модуль, котрий взаємодіє з модулями: os, numpy, matplotlib.

Коефіцієнт прискорення  показує скорочення часу виконання паралельної програми в паралельній системі з  процесорами  в порівнянні з часом виконання послідовної програми в однопроцесорній системі :



Коефіцієнт ефективності  застосування комп’ютерної системи показує ступінь використання процесорів системи:



Результати тестування і проведених досліджень ефективності розробленої програми наведено в таблицях 2.1 – 2.3.

Таблиця 2.1 Час виконання програми ПРГ1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| N | Т1, мс | Т2, мс | Т3, мс | Т4, мс | Т5, мс | Т6, мс |
| 900 | 3194 | 1531 | 1236 | 956 | 1057 | 1117 |
| 1500 | 38152 | 18515 | 13032 | 9917 | 9987 | 10092 |
| 1800 | 83187 | 42517 | 29258 | 22303 | 21606 | 20889 |

На основі даних із таблиці 2.1 виконано розрахунок значень коефіцієнтів прискорення, які наведені в таблиці 2.2.

Таблиця 2.2 Коефіцієнти прискорення для програми ПРГ1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| N | Кількість процесорів (P) | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 |
| 900 | 1,00 | 2,09 | 2,58 | 3,34 | 3,02 | 2,86 |
| 1500 | 1,00 | 2,06 | 2,93 | 3,85 | 3,82 | 3,78 |
| 1800 | 1,00 | 1,96 | 2,84 | 3,73 | 3,85 | 3,98 |

Коефіцієнти ефективності (таблиця 2.3) обчислені за даними таблиці 2.2.

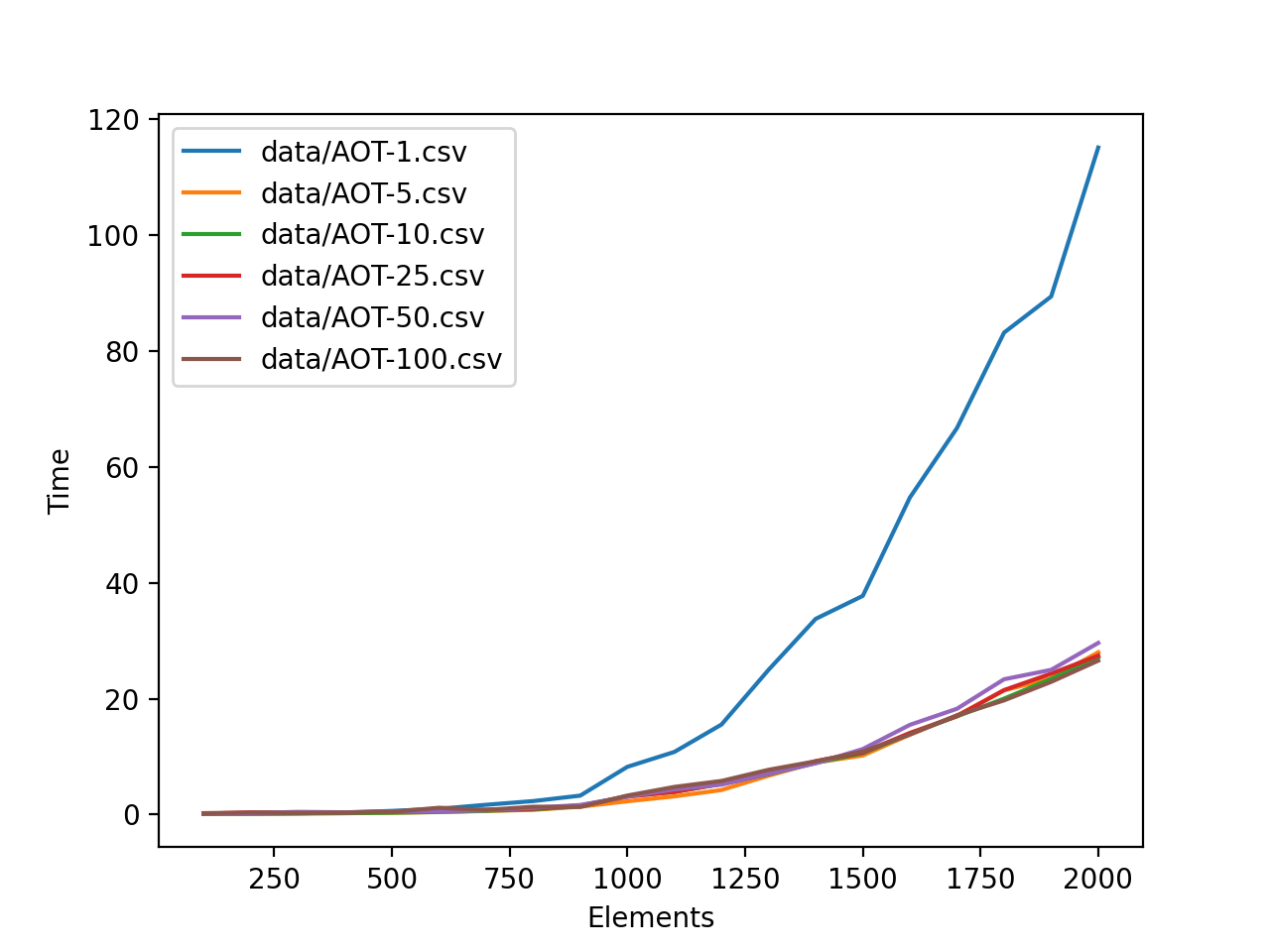
Таблиця 2.3 Коефіцієнти ефективності для програми ПРГ1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| N | Кількість процесорів (P) | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 |
| 900 | 100,00% | 104,5% | 86% | 83,5% | 60,4% | 57,2% |
| 1500 | 100,00% | 103% | 97,67% | 96,26% | 76,4% | 75,6% |
| 1800 | 100,00% | 98% | 94,67% | 93,25% | 77% | 79,6% |

Використовуючи таблиці 2.2-2.3 побудовано графіки(pис. 2.4, рис 2.5) зміни коефіцієнтів прискорення і ефективності в залежності від *N* і *P*.

Рис. 2.4 – Графік зміни коефіцієнту прискорення програми ПРГ1 в залежності від кількості потоків

Рис. 2.5 – Графік зміни коефіцієнту ефективності програми ПРГ1 в залежності від кількості потоків

Рис. 2.6 – Графік зміни часу виконання програми в залежності від кількості потоків та елементів

**2.6 Висновки до розділу 2**

Виконано розробку програми ПРГ1 для ПКС СП з використанням мови Java. Тестування програми ПРГ1 показало наступне:

* Використання багатоядерної ПКС та програми ПРГ1 забезпечує скорочення часу обчислення заданої математичної задачі при збільшенні P для всіх N, як показано у табл. 2.1.
* Значення Kn лежить в межах від 1,96 до 3,98, як показано у табл. 2.2.
* Максимальне значення Kn = 3,98 забезпечує ПКС з P = 6 та N = 1800.
* Мінімальне значення Kn = 1,96 виявлено у ПКС з P = 2 та N = 1800.
* З ростом N для усіх значень P Kn лінійно зростає поки кількість потоків не стає більше 4, надалі збільшеня потужності сильно спадає, як показано на рис.2.6.Це обумовлено кількістью ядер у системі, а саме в моїй їх 4
* Значення Ke змінюються від 57,2% до 104,5%, як показано у табл.2.3.
* Найефективніше програма ПРГ1 використовує ПКС з P = 2 та N = 900 при цьому Ke = 104,5%.
* Найнижча ефективність використання ПКС програмою ПРГ1 виявлена при P = 6, N = 900 при цьому Ke = 57,2%.