**Міністерство освіти і науки України**

**Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського"**

**Факультет інформатики та обчислювальної техніки**

**Кафедра інформатики та програмної інженерії**

**Звіт**

з лабораторної роботи № 2 з дисципліни

«Технології паралельних обчислень»

**«Розробка паралельних алгоритмів множення матриць та дослідження їх ефективності»**

**Виконав(ла)**

(шифр, прізвище, ім'я, по батькові)

*ІП-02, Зусько Владислав Юрійович*

Київ 2023

**Завдання #1**

Реалізуйте програмний код, даний у лістингу, та протестуйте його при різних значеннях параметрів. Модифікуйте програму, використовуючи методи управління потоками, так, щоб її робота була завжди коректною. Запропонуйте три різних варіанти управління. **30 балів.**

Код 1.1

import *java.util.concurrent.atomic.AtomicIntegerArray*;

import *java.util.concurrent.locks.Lock*;

import *java.util.concurrent.locks.ReentrantLock*;

*class* Bank {

*public* *static* *final* int NTEST = 10000;

*private* *final* int[] accounts;

*private* long ntransacts = 0;

*private* *final* Lock locker = new ReentrantLock();

*public* Bank(int n, int initialBalance) {

    accounts = new int[n];

    int i;

    for (i = 0; i < accounts.length; i++)

      accounts[i] = initialBalance;

    ntransacts = 0;

  }

*public* *synchronized* void transferSync(int from, int to, int amount) {

    accounts[from] -= amount;

    accounts[to] += amount;

    ntransacts++;

    if (ntransacts % NTEST == 0)

      test();

  }

*public* void transferSyncBlock(int from, int to, int amount) {

*synchronized* (this) {

      accounts[from] -= amount;

      accounts[to] += amount;

      ntransacts++;

      if (ntransacts % NTEST == 0)

        test();

    }

  }

*public* void transferLocker(int from, int to, int amount) {

    locker.lock();

    try {

      accounts[from] -= amount;

      accounts[to] += amount;

      ntransacts++;

      if (ntransacts % NTEST == 0)

        test();

    } finally {

      locker.unlock();

    }

  }

*public* void test() {

    int sum = 0;

    for (int i = 0; i < accounts.length; i++)

      sum += accounts[i];

    System.out.println("Transactions:" + ntransacts

        + " Sum: " + sum);

  }

*public* int size() {

    return accounts.length;

  }

}

**Завдання #2**

Реалізуйте програмний код, даний у лістингу, та протестуйте його при різних значеннях параметрів. Модифікуйте програму, використовуючи методи управління потоками, так, щоб її робота була завжди коректною. Запропонуйте три різних варіанти управління. **30 балів.**