**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 №1 з дисципліни

«Технології паралельних обчислень»

**„Створення та запуск потоків: імітація руху більярдних кульок. Найпростіші засоби управління потоками.”**

**Виконав(ла)**

(шифр, прізвище, ім'я, по батькові)

*ІП-11 Головня Олександр Ростиславович*

**Перевірив**

(прізвище, ім'я, по батькові)

Київ 2024

**Завдання до комп’ютерного практикуму:**

1. Реалізуйте програму імітації руху більярдних кульок, в якій рух кожної кульки відтворюється в окремому потоці (див. презентацію «Створення та запуск потоків в java» та приклад). Спостерігайте роботу програми при збільшенні кількості кульок. Поясніть результати спостереження. Опишіть переваги потокової архітектури програм. 10 балів.

2. Модифікуйте програму так, щоб при потраплянні в «лузу» кульки зникали, а відповідний потік завершував свою роботу. Кількість кульок, яка потрапила в «лузу», має динамічно відображатись у текстовому полі інтерфейсу програми. 10 балів.

3. Виконайте дослідження параметру priority потоку. Для цього модифікуйте програму «Більярдна кулька» так, щоб кульки червоного кольору створювались з вищим пріоритетом потоку, в якому вони виконують рух, ніж кульки синього кольору. Спостерігайте рух червоних та синіх кульок при збільшенні загальної кількості кульок. Проведіть такий експеримент. Створіть багато кульок синього кольору (з низьким пріоритетом) і одну червоного кольору, які починають рух в одному й тому ж самому місці більярдного стола, в одному й тому ж самому напрямку та з однаковою швидкістю. Спостерігайте рух кульки з більшим пріоритетом. Повторіть експеримент кілька разів, значно збільшуючи кожного разу кількість кульок синього кольору. Зробіть висновки про вплив пріоритету потоку на його роботу в залежності від загальної кількості потоків. 20 балів.

4. Побудуйте ілюстрацію методу join() класу Thread через взаємодію потоків, що відтворють рух більярдних кульок різного кольору. Поясніть результат, який спостерігається. 10 балів.

5. Створіть два потоки, один з яких виводить на консоль символ ‘-‘, а інший – символ ‘|’. Запустіть потоки в основній програмі так, щоб вони виводили свої символи в рядок. Виведіть на консоль 100 таких рядків. Поясніть виведений результат. 10 балів. Використовуючи найпростіші методи управління потоками, добийтесь почергового виведення на консоль символів. 15 балів.

6. Створіть клас Counter з методами increment() та decrement(), які збільшують та зменшують значення лічильника відповідно. Створіть два потоки, один з яких збільшує 100000 разів значення лічильника, а інший – зменшує 100000 разів значення лічильника. Запустіть потоки на одночасне виконання. Спостерігайте останнє значення лічильника. Поясніть результат. 10 балів. Використовуючи синхронізований доступ, добийтесь правильної роботи лічильника при одночасній роботі з ним двох і більше потоків. Опрацюйте використання таких способів синхронізації: синхронізований метод, синхронізований блок, блокування об’єкта. Порівняйте способи синхронізації. 15 балів.

**Текст програми:**

[**https://github.com/YeaLowww/Parallel-Programming/tree/main/parallel\_lab\_1/src/main/java/org/example**](https://github.com/YeaLowww/Parallel-Programming/tree/main/parallel_lab_1/src/main/java/org/example)

**Результати виконання коду:**

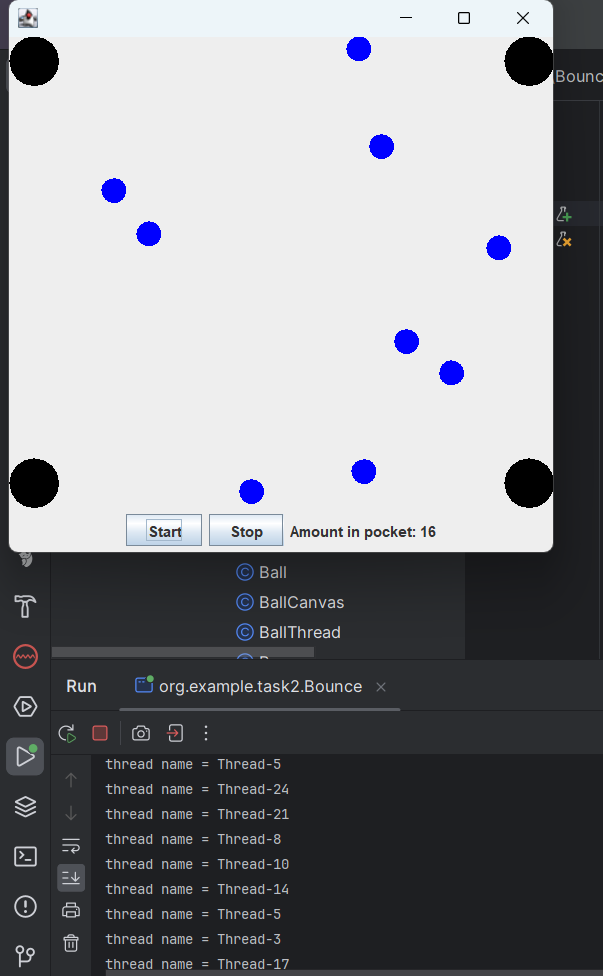
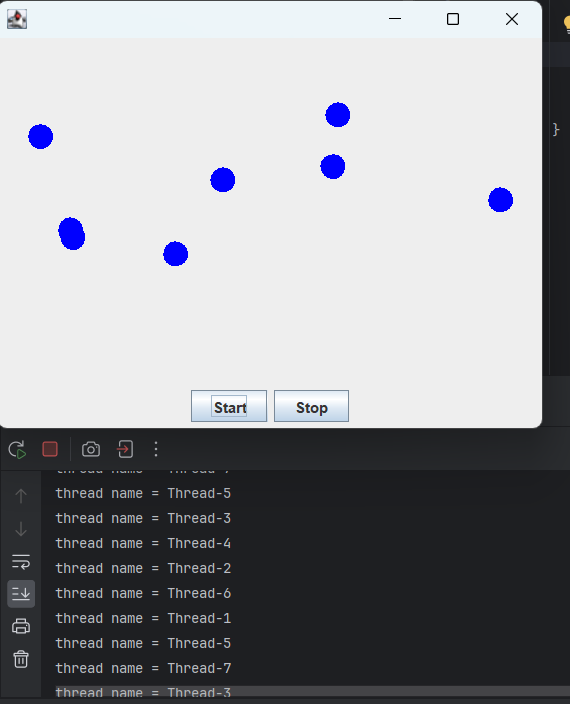


Рисунок 1. – Демонстрація виконання підзавдань 1 та 2: створення та припинення потоків

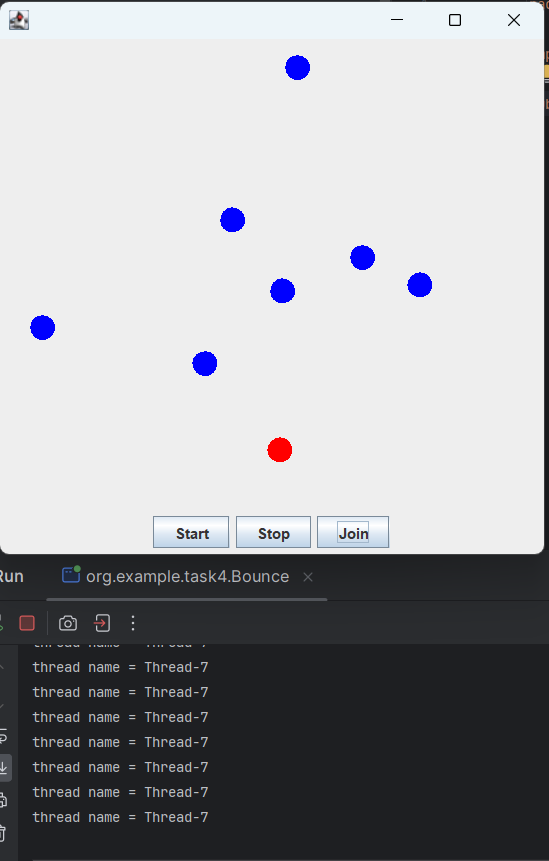
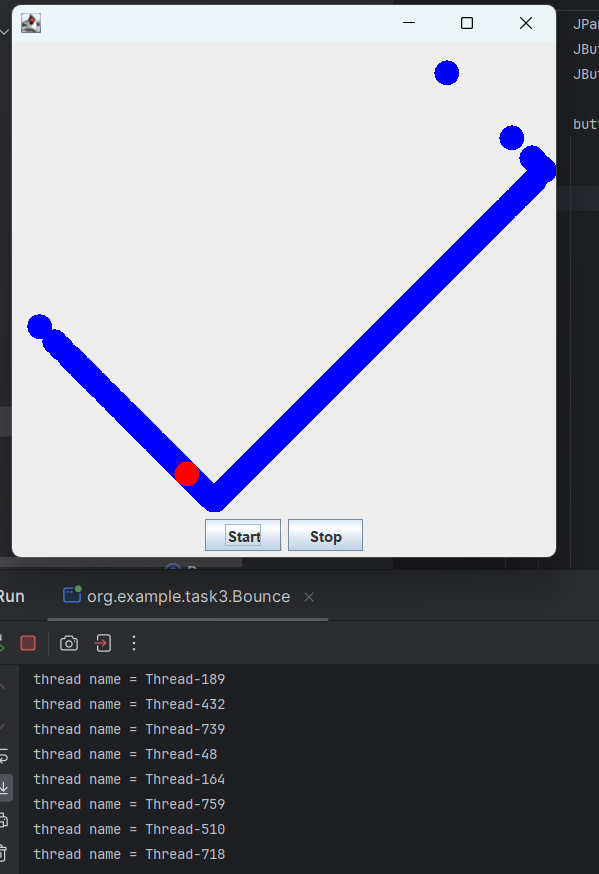


Рисунок 2. – Демонстрація виконання підзавдань 3 та 4: Приорітет та join()

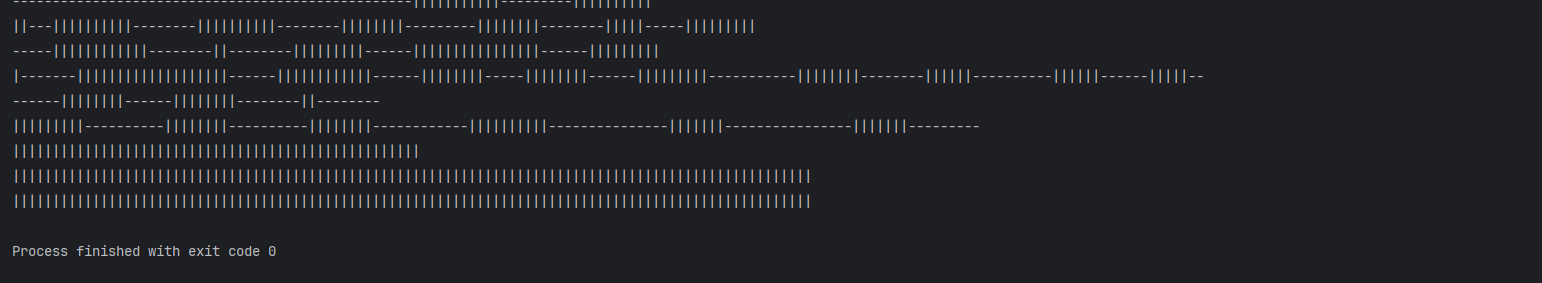
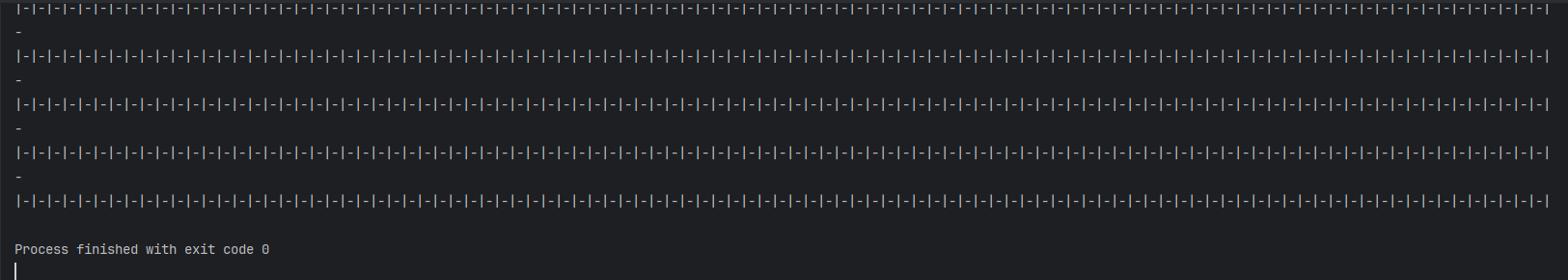


Рисунок 3. – Демонстрація виконання підзавдання 5: Вивід символів

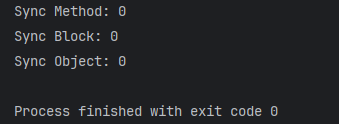
****

Рисунок 4. – Демонстрація виконання підзавдання 6: Результат лічильника при використанні різних способів синхронізації

**Висновок:** Під час розробки лабораторної роботи, я реалізував програму імітації руху більярдних кульок. Створив та запустив потоки в java, модифіковув програму для дослідження параметру priority потоку. Побудував ілюстрацію методу join() класу Thread. За допомогою двох потоків добився почергового виведення на консоль різних символів. Опрацював використання таких способів синхронізації: синхронізований метод, синхронізований блок, блокування об’єкта для класа Counter