Министерство образования и науки РФ

Федеральное государственное автономное

образовательное учреждение высшего образования

«Санкт-Петербургский национальный исследовательский университет

информационных технологий, механики и оптики»

**факультет программной инженерии и компьютерной техники**

**ЛАБОРАТОРНАЯ РАБОТА № 4**

по дисциплине

‘Методы и средства программной инженерии’

Вариант № 346712

*Выполнили:*

Студент

Р32111 Павлов Александр Сергеевич

*Преподаватель:*

Бострикова Дарья Константиновна



Санкт-Петербург, 2023

Задание

Изображение выглядит как текст, снимок экрана, Шрифт, документ

Автоматически созданное описание

Исходный код программы

<https://github.com/ExcaliBBur/mispi/tree/main/lab4>

Показания утилиты JConsole

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Демонстрация полей:

Изображение выглядит как текст, снимок экрана, число, линия

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, число, линия

Автоматически созданное описание

Наличие метода:

Изображение выглядит как текст, снимок экрана, число, линия

Автоматически созданное описание

Выполнение метода:

Изображение выглядит как текст, снимок экрана, дисплей, число

Автоматически созданное описание

Оповещение о 4 промахах подряд:

Изображение выглядит как текст, линия, Шрифт, снимок экрана

Автоматически созданное описание

**Время, прошедшее с момента запуска виртуальной машины:**

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

16 минут = 16 \* 60 \* 60 = 57600 мс.

Показания утилиты VisualVM

График изменения показаний первого бина (счётчика):

Изображение выглядит как текст, снимок экрана, диаграмма, Параллельный

Автоматически созданное описание

График изменения показаний второго бина (площадь):

Изображение выглядит как линия, График, диаграмма, снимок экрана

Автоматически созданное описание

**Имя класса, объекты которого занимают наибольший объем памяти JVM:**

****

Локализация и устранение проблем с производительностью

1. Проверка программы на утечку памяти

Изображение выглядит как текст, снимок экрана, диаграмма, График

Автоматически созданное описание

Исходя из полученных данных, можно сделать вывод что GC работает корректно и утечки памяти не происходит.

Изображение выглядит как текст, снимок экрана, График, линия

Автоматически созданное описание

Сверхъестественной нагрузки на процессор не наблюдается.

Изображение выглядит как текст, снимок экрана, линия, График

Автоматически созданное описание

Количество потоков не увеличивается.

**Таким образом, можно сделать вывод о том, что проблем с производительностью в данной программе нет.**

Вывод

Во время выполнения лабораторной работы мною были изучены такие утилиты, как JConsole и VisualVM для мониторинга и профилирования Java-приложений.