МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ

КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

**«Московский технический университет связи и информатики»**

Кафедра «Математическая кибернетика и информационные технологии»

Лабораторная работа №7

**по дисциплине**

**«Информационные технологии и программирование»**

Выполнил: студент гр. БВТ2201

Аблязов И.В.

Проверил:

Москва, 2023 г.

# ОСНОВНАЯ ЧАСТЬ

## Задание 1

В первом задании необходимо реализовать многопоточную программу для вычисления суммы элементов массива при помощи двух потоков (1 вариант).

Для начала создадим свой класс-поток MyThread. В нем реализуем такие поля, как: сумма, начальный индекс массива, конечный индекс массива, исходный массив. В методе run пропишем механизм сложения всех значений массива с индексов от старта, до финиша.

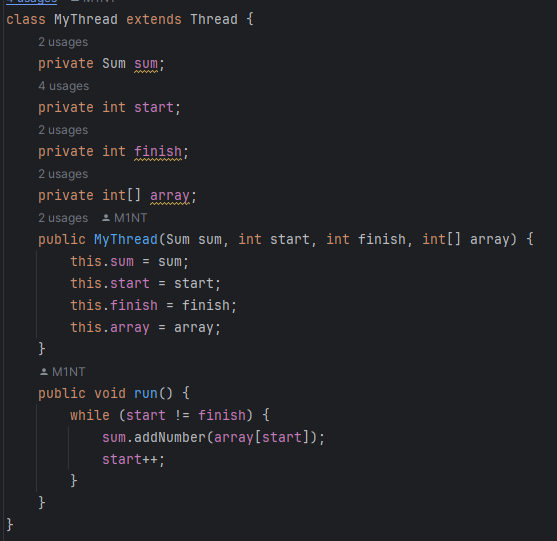


Рисунок 1 Класс MyThread

Также для данного класса понадобится другой класс, в котором будет храниться текущая сумма.

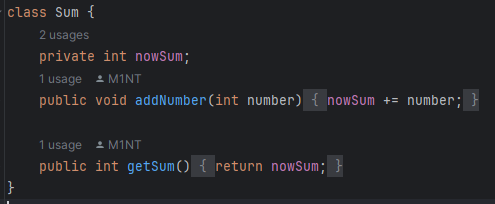


Рисунок 2 Класс Sum

После необходимо создать два потока, раздели массив пополам с помощью индексов.



Рисунок 3 Метод ArraySum

Проведем тестирование этого метода.



Рисунок 4 Тест 1 задания

Получим следующие ответы:



Рисунок 5 Задание 1

## Задание 2

Во втором задании необходимо реализовать многопоточную программу для поиска наибольшего элемента в матрице с помощью пула потоков (2 вариант, класс ExecutorService)

Для данного задания создаем пул потоков с помощью Executors.newFixedThreadPool, где количество потоков совпадает с количеством строк. Далее в каждом потоке необходимо найти максимальный элемент строки и добавить его в новый список. После выполнения всех потоков находим максимальное число из всех полученных результатов.



Рисунок 6 2 задание

Проведем тестирование данной программы

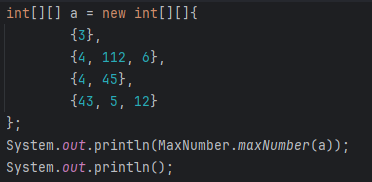


Рисунок 7 Тест 2 задания

Получим необходимый ответ.



Рисунок 8 Ответ на тест (2 задание)

## 3 задание

В третьем задании необходимо создать программу по перевозке груза из одного склада на другой с учетом ограничения на 150 кг веса и 3-х грузчиков, используя класс Thread, классы для товаров, складов и грузчиков (1 вариант).

В данном задании создадим 4 класса – продукты, склады, грузчики и машина.

В классе продуктов есть только 1 поле – вес продукта.

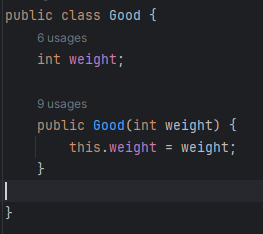


Рисунок 9 Класс Good

В классе склада находится поле - список из товаров и методы для взятия товара со склада, доставки товара на склад и проверки на пустоту склада.



Рисунок 10 Класс Storage

Класс Car похож на класс склада, но для него также необходимо учитывать вес хранимого товара.



Рисунок 11 Класс Car

Класс грузчиков мы наследуем от Thread и реализуем логику перетаскивания товара.

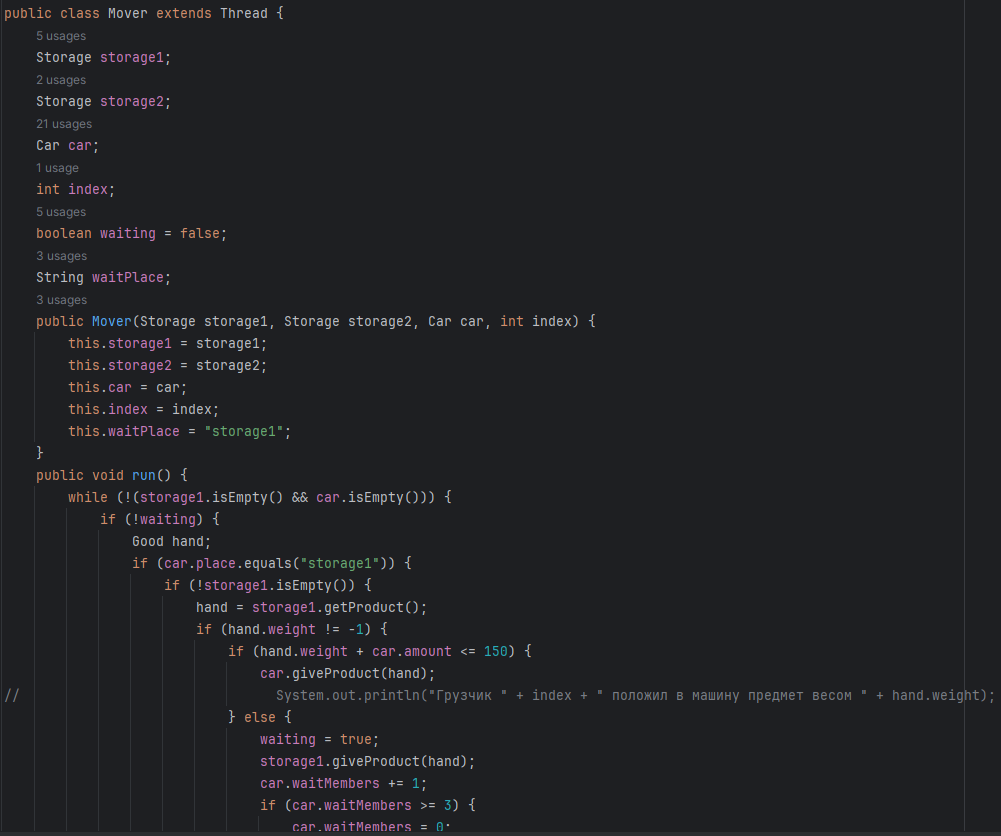


Рисунок 12 Класс Mover (1)

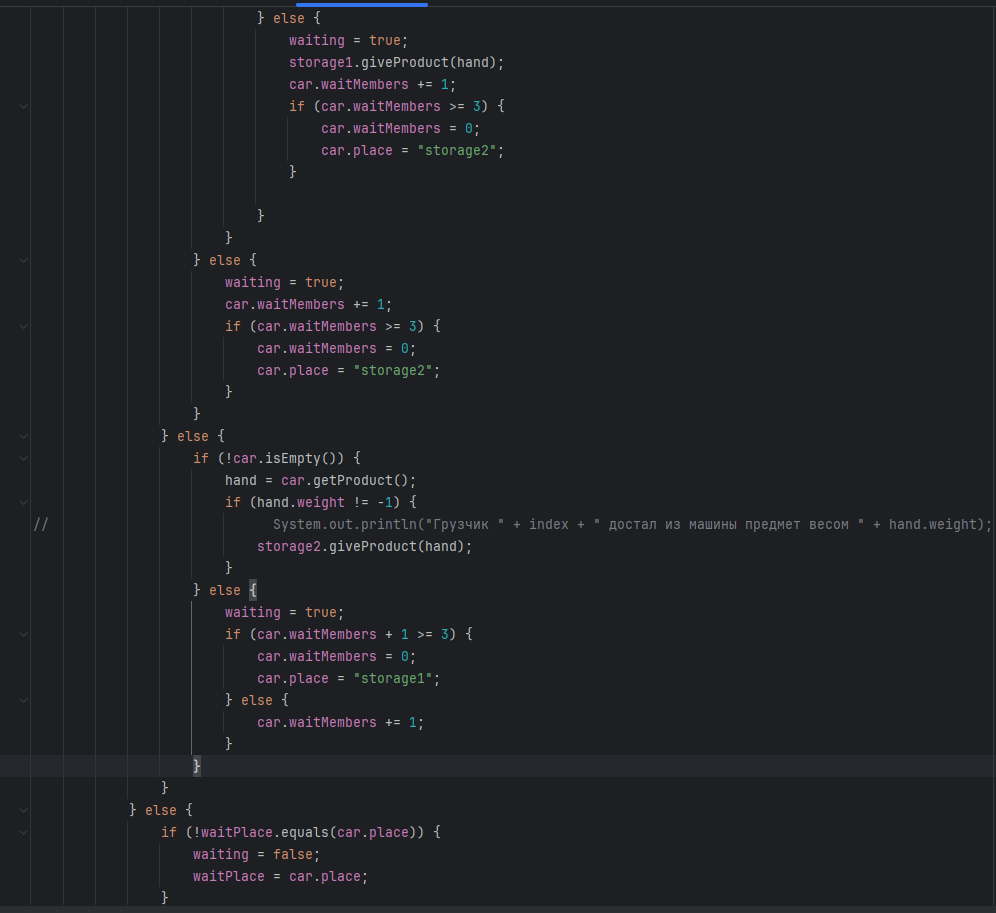


Рисунок 13 Класс Mover (2)

Протестируем нашу программу.

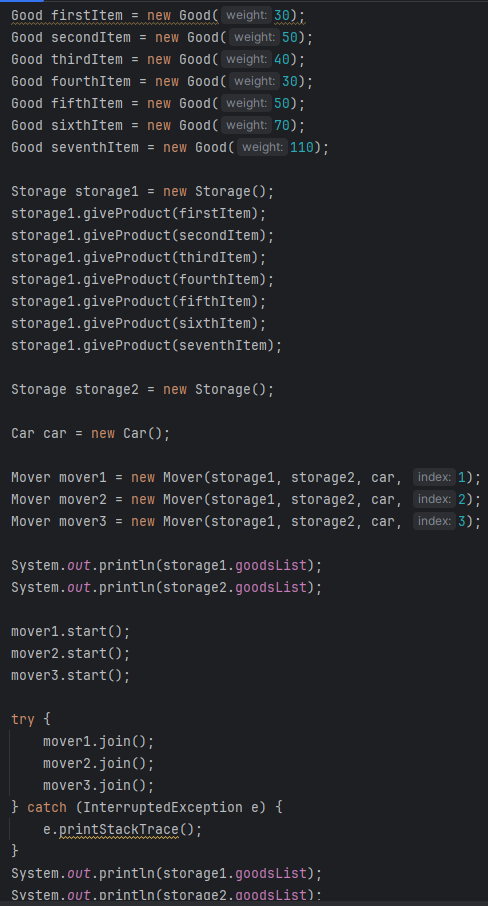


Рисунок 14 Тест третьего задания

Получим следующий результат (рисунок 15).

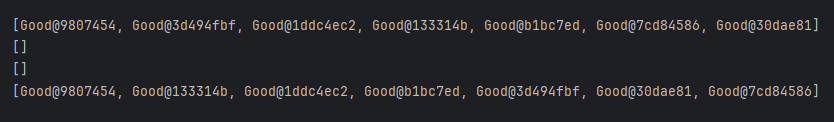


Рисунок 15 Задание 3

# ВЫВОД

Данная лабораторная работа помогла лучше освоить язык Java. Была проделана работа с многопоточностью. Также были изучены различные классы для работы с потоками. Все знания были закреплены на практике и сохранены на GitHub: <https://github.com/HungryM1NT/ITIP>