|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.04.01/07 **Интеллектуальные системы анализа, обработки и интерпретации больших данных**

**Отчет**

|  |
| --- |
| **по лабораторной работе №8** |

**Название:**

Потоки

**Дисциплина:** Языка программирования для работы с большими данными

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-23М |  |  | Д.Д. Капитонов |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | П.В. Степанов |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2022

**Вариант 1, номера 2 и 3.**

2. Реализовать многопоточное приложение “Робот”. Надо написать робота, который умеет ходить. За движение каждой его ноги отвечает отдельный поток. Шаг выражается в выводе в консоль LEFT или RIGHT.

3. Реализовать многопоточное приложение “Магазин”. Вся цепочка: производитель-магазин-покупатель. Пока производитель не поставит на склад продукт, покупатель не может его забрать. Реализовать приход товара от производителя в магазин случайным числом. В том случае, если товара в магазине не хватает– вывести сообщение.

**Код программы:**

*package dan.lab\_8;*

*import java.util.Objects;*

*import java.util.concurrent.Exchanger;*

*public class Lab\_8 {*

*public static class Robot{*

*public class Leg implements Runnable {*

*private final String name;*

*private final boolean leg;*

*Leg(String name, boolean leg) {*

*this.name = name;*

*this.leg = leg;*

*}*

*@Override*

*public void run() {*

*while(true) {*

*if (leg == currentLeg) {*

*step();*

*currentLeg = !leg;*

*Thread.yield();*

*}*

*}*

*}*

*private void step() {*

*System.out.println(name);*

*}*

*}*

*boolean currentLeg = true;*

*Leg left = new Leg("LEFT", false);*

*Leg right = new Leg("RIGHT", true);*

*void run() {*

*new Thread(left).start();*

*new Thread(right).start();*

*}*

*public Robot() {*

*System.out.println("Robot:");*

*}*

*}*

*public static class Shop {*

*public static class Product implements Runnable {*

*private final String operation;*

*private final Integer request;*

*Product(String operation, Integer request) {*

*this.operation = operation;*

*this.request = request;*

*run();*

*}*

*@Override*

*public void run() {*

*if (Objects.equals(operation, "+")) {*

*step();*

*product += this.request;*

*}*

*else {*

*if (product < this.request){*

*System.out.println("No product");*

*}*

*else {*

*step();*

*product -= this.request;*

*}*

*}*

*}*

*private void step() {*

*System.out.println("Product - "+product+". Operation "+operation+". Change on "+request);*

*}*

*}*

*public static Integer product = 0;*

*Product people\_manuf = new Product("+", (int) (Math.random() \* 100));*

*Product people\_buyer = new Product("-", (int) (Math.random() \* 100));*

*public Shop() {*

*new Thread(people\_manuf).start();*

*new Thread(people\_buyer).start();*

*}*

*}*

*public static void main(String[] args) {*

*Robot robot1 = new Robot();*

*Shop shop1 = new Shop();*

*}*

*}*

**Вывод:** научились работать с потоками.