|  |  |
| --- | --- |
| 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**

**Отчет**

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

**Дисциплина: Языки программирования для работы с большими данными.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-21М |  |  | В.В. Ерискин |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | П.В. Степанов |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2020

**Задание**

Задача 1.

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.

**Решение**:

|  |
| --- |
|  |
| public class Robot { | |
|  | Object CommunicationBetweenFoots = new Object(); |
|  | RightFoot RightFoot = new RightFoot(CommunicationBetweenFoots); |
|  | LeftFoot LeftFoot = new LeftFoot(CommunicationBetweenFoots); |
|  |  |
|  | public static void main(String[] args) { |
|  | new Robot(); |
|  | } |
|  |  |
|  | Robot() { |
|  | RightFoot.start(); |
|  | LeftFoot.start(); |
|  | } |
|  |  |
|  | class RightFoot extends Thread { |
|  | Object CommunicationBetweenFoots = new Object(); |
|  |  |
|  | public RightFoot(Object communicationBetweenFoots) { |
|  | this.CommunicationBetweenFoots = communicationBetweenFoots; |
|  | } |
|  |  |
|  | @Override |
|  | public void run() { |
|  |  |
|  | while (true) { |
|  | try { |
|  | synchronized (CommunicationBetweenFoots) { |
|  | CommunicationBetweenFoots.wait(); |
|  | } |
|  | } catch (InterruptedException e) { |
|  | } |
|  | System.out.println("RIGHT"); |
|  | try { |
|  | sleep(1000); |
|  | } catch (InterruptedException e1) { |
|  | } |
|  | synchronized (CommunicationBetweenFoots) { |
|  | CommunicationBetweenFoots.notify(); |
|  | } |
|  | } |
|  | } |
|  |  |
|  | } |
|  |  |
|  | class LeftFoot extends Thread { |
|  | Object CommunicationBetweenFoots = new Object(); |
|  |  |
|  | public LeftFoot(Object communicationBetweenFoots) { |
|  | this.CommunicationBetweenFoots = communicationBetweenFoots; |
|  | } |
|  |  |
|  | @Override |
|  | public void run() { |
|  | while (true) { |
|  | System.out.println("LEFT"); |
|  | try { |
|  | sleep(1000); |
|  | } catch (InterruptedException e1) { |
|  | } |
|  | synchronized (CommunicationBetweenFoots) { |
|  | CommunicationBetweenFoots.notify(); |
|  | } |
|  | try { |
|  | synchronized (CommunicationBetweenFoots) { |
|  | CommunicationBetweenFoots.wait(); |
|  | } |
|  | } catch (InterruptedException e) { |
|  | } |
|  | } |
|  | } |
|  | } |
|  | } |

**Вывод:**

Задание выполнено, программа компилируются и успешно выполняются.