



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, искусственный интеллект и системы управления

КАФЕДРА Компьютерные системы и сети

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

ОТЧЕТ
по лабораторной работе №8

Название: Потоки

Дисциплина Языки программирования для работы с большими
данными

Студент ИУ6–22М
(Группа)

М.Э.Хабаров
(Подпись, дата) (И.О. Фамилия)

Преподаватель

П.В. Степанов
(Подпись, дата) (И.О. Фамилия)

2024 г

Цель: изучить работу с потоками на языке Java.

Задание №1

Формулировка задания и код программы представлены в листинге 1:

```
/*
1. Реализовать многопоточное приложение "Банк". Имеется банковский счет.
Сделать синхронным пополнение и снятие денежных средств на счет/со счет
случайной суммой. При каждой операции (пополнения или снятие) вывести
текущий
баланс счета. В том случае, если денежных средств недостаточно – вывести
сообщение.
*/
public class BankAccount {
    private int balance;

    public BankAccount(int initialBalance) {
        this.balance = initialBalance;
    }

    public synchronized void deposit(int amount) {
        balance += amount;
        System.out.println("Пополнение на " + amount + ". Текущий баланс: " +
balance);
        notify();
    }

    public synchronized void withdraw(int amount) {
        if (balance < amount) {
            System.out.println("Недостаточно средств для снятия " + amount);
            try {
                wait(); // Ждем пока не будет достаточно средств на счете
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
        balance -= amount;
        System.out.println("Снятие " + amount + ". Текущий баланс: " +
balance);
    }

    public static void main(String[] args) {
        BankAccount account = new BankAccount(1000);

        Thread depositThread = new Thread(() -> {
            while (true) {
                int amount = (int) (Math.random() * 500) + 1;
                account.deposit(amount);
                try {
                    Thread.sleep(1000); // Ждем 1 секунду перед следующим
пополнением
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
            }
        });

        Thread withdrawThread = new Thread(() -> {
            while (true) {
                int amount = (int) (Math.random() * 500) + 1;
                account.withdraw(amount);
                try {
```

```

        Thread.sleep(1500); // Ждем 1.5 секунды перед следующим
снятием
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
}
});

depositThread.start();
withdrawThread.start();
}
}

```

Задание №2

Формулировка задания и код программы представлены в листинге 2:

```

public class Robot {
    public static void main(String[] args) {
        Thread leftLeg = new Thread(() -> {
            while (true) {
                System.out.println("LEFT");
                try {
                    Thread.sleep(1000); // Задержка для имитации движения
ноги
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
            }
        });

        Thread rightLeg = new Thread(() -> {
            while (true) {
                System.out.println("RIGHT");
                try {
                    Thread.sleep(1000); // Задержка для имитации движения
ноги
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
            }
        });

        leftLeg.start();
        rightLeg.start();
    }
}

```

Вывод: в результате выполнения лабораторной работы были освоены основные принципы потоков на языке Java.