



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

О Т Ч Е Т

по лабораторной работе № 8

Название: Потоки

Дисциплина: Языки программирования для работы с большими
данными

Студент

ИУ6-23М

(Группа)

(Подпись, дата)

Д.В. Авдонин

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

(И.О. Фамилия)

Москва, 2022

Лабораторная работа № 8

Задание:

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 снятия) вывести текущий баланс счета. В том случае, если денежных средств недостаточно – вывести сообщение.

Ход работы:

Код программы:

```
public class Bank {
    public int balance;
    Random random = new Random();

    public static void main(String[] args){

        System.out.println("Start");
        new Bank().doStart();
    }
    void doStart() {
        balance = 0;
        new Thread(minus).start();
        new Thread(plus).start();
    }

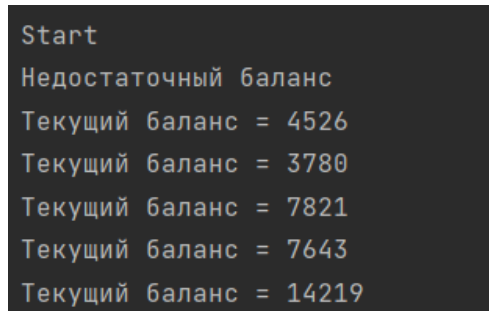
    Runnable minus = new Runnable() {
        @Override
        public void run() {
            while (true) {
                int sum = random.nextInt(1000);
                if (sum > balance) {
                    System.out.println("Недостаточный баланс");
                } else {
                    balance -= sum;
                    System.out.println("Текущий баланс = " + balance);
                }
                try {
                    Thread.sleep(1000);
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
            }
        }
    };

    Runnable plus = new Runnable() {
        @Override
        public void run() {
            while (true) {
                int sum = random.nextInt(10000);
                balance += sum;
                System.out.println("Текущий баланс = " + balance);
            }
        }
    };
}
```

```

        try {
            Thread.sleep(1000);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}
};
}

```



```

Start
Недостаточный баланс
Текущий баланс = 4526
Текущий баланс = 3780
Текущий баланс = 7821
Текущий баланс = 7643
Текущий баланс = 14219

```

Рисунок 1 – Результат работы программы

Задание:

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.

Ход работы:

Код программы:

```

public volatile boolean leg_flag = true;
public static void main(String[] args){
    System.out.println("Start");
    new Lab8().doStart();
}
void doStart() {
    new Thread(left).start();
    new Thread(right).start();
}

Runnable left = new Runnable() {
    @Override
    public void run() {
        while (true) {
            if (leg_flag) {
                System.out.println("LEFT");
                leg_flag = false;
            }
            Thread.yield();
        }
    }
};

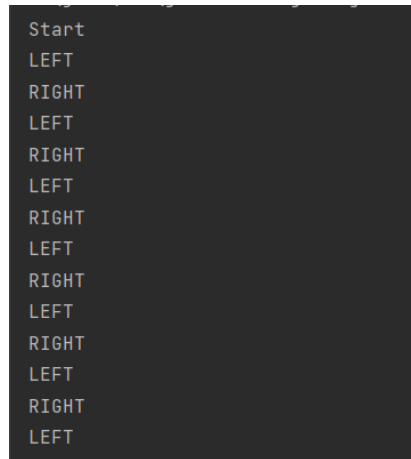
Runnable right = new Runnable() {
    @Override

```

```

public void run() {
    while (true){
        if (!leg_flag) {
            System.out.println("RIGHT");
            leg_flag = true;
        }
        Thread.yield();
    }
}
};

```



```

Start
LEFT
RIGHT
LEFT
RIGHT
LEFT
RIGHT
LEFT
RIGHT
LEFT
RIGHT
LEFT
RIGHT
LEFT
RIGHT
LEFT

```

Рисунок 2 – Результат работы программы

Задание:

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.

Ход работы:

Код программы:

```

public class Magazin {
    public int amount;
    Random random = new Random();

    public static void main(String[] args){
        System.out.println("Start");
        new Magazin().doStart();
    }
    void doStart() {
        amount = 0;
        new Thread(delivery).start();
        new Thread(purchase).start();
    }

    Runnable purchase = new Runnable() {
        @Override
        public void run() {

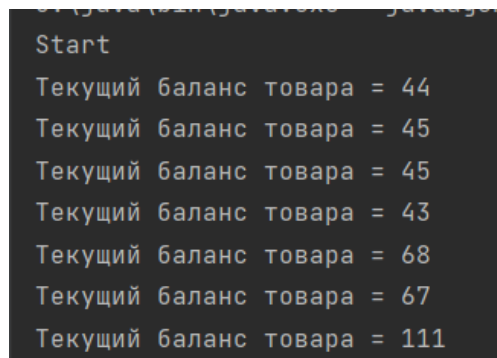
```

```

while (true) {
    if (amount == 0) {
        System.out.println("Недостаточно товара");
    } else {
        amount--;
        System.out.println("Текущий баланс товара = " + amount);
    }
    Thread.yield();
    try {
        Thread.sleep(4000);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
}
}
};

Runnable delivery = new Runnable() {
    @Override
    public void run() {
        while (true) {
            int sum = random.nextInt(100);
            amount += sum;
            System.out.println("Текущий баланс товара = " + amount);
            Thread.yield();
            try {
                Thread.sleep(4000);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}
};

```



```

Start
Текущий баланс товара = 44
Текущий баланс товара = 45
Текущий баланс товара = 45
Текущий баланс товара = 43
Текущий баланс товара = 68
Текущий баланс товара = 67
Текущий баланс товара = 111

```

Рисунок 3 – Результат работы программы

Вывод: лабораторная работа выполнена в соответствии с заданием и вариантом.