

**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»
(МТУСИ)

Информационные технологии и программирование

Отчет по лабораторной работе №1

Выполнил: студент группы
БПИ2305 Архипов А.А.

Москва

Цель работы

1. Ознакомиться с основами написания и выполнения программ на языке программирования Java.
2. Изучить процесс компиляции и запуска Java-программы через командную строку.
3. Закрепить навыки работы с методами и циклами.

Задание

Задание 1

Написать программу, которая выводит все простые числа меньше 100. Создать класс Primes, содержащий метод для проверки, является ли число простым.

Задание 2

Написать программу, которая проверяет, является ли введенная строка палиндромом. Создать класс Palindrome, реализующий методы для переворачивания строки и проверки её на палиндром.

Ход выполнения работы

Часть 1: Программа для вывода простых чисел

Код программы (Primes.java):

```
public class Primes {
    public static void main(String[] args) {
        int n = 100;
        for (int i = 0; i <= n; i++) {
            if (isPrime(i)) {
                System.out.println(i);
            }
        }
    }
    public static boolean isPrime(int n) {
        if (n < 2) {
            return false;
        }
        for (int i = 2; i < n; i++) {
            if (n % i == 0) { // n делится без остатка на i
                return false;
            }
        }
        return true;
    }
}
```

Описание работы программы:

1. Программа использует метод `isPrime(int n)`, который проверяет, является ли число `n` простым.
2. Метод перебирает все значения от 2 до `n-1` и проверяет делимость `n` на каждое значение. Если число делится на какое-либо значение без остатка, оно не является простым.
3. В методе `main` цикл перебирает все числа от 0 до 100 и выводит на экран те, которые являются простыми.

Результат работы программы:

На экране отображаются все простые числа от 2 до 100, а именно: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97.

Часть 2: Программа для проверки палиндромов

Код программы (Palindrome.java):

```
public class Palindrome {  
    public static void main(String[] args) {  
        for (int i = 0; i < args.length; i++) {  
            String s = args[i];  
            if (isPalindrome(s)) {  
                System.out.println(s + " is a palindrome");  
            } else {  
                System.out.println(s + " is not a palindrome");  
            }  
        }  
    }  
  
    public static String reverseString(String s) {  
        String reversed = "";  
        for (int i = s.length() - 1; i >= 0; i--) {  
            reversed += s.charAt(i);  
        }  
        return reversed;  
    }  
  
    public static boolean isPalindrome(String s) {  
        return s.equals(reverseString(s));  
    }  
}
```

Описание работы программы:

1. Метод `reverseString(String s)` переворачивает строку, используя цикл для добавления символов в обратном порядке в новую строку.
2. Метод `isPalindrome(String s)` проверяет, является ли строка палиндромом, сравнивая исходную строку с её перевёрнутой версией.

3. В методе `main` программа проходит по всем аргументам командной строки и проверяет, является ли каждый из них палиндромом, выводя результат на экран.

Результат работы программы:

Программа запускается с аргументами командной строки, например:

```
java Palindrome madam racecar apple kayak song noon
```

На выходе программа выдаёт:

```
madam is a palindrome  
racecar is a palindrome  
apple is not a palindrome  
kayak is a palindrome  
song is not a palindrome  
noon is a palindrome
```

Выводы

В ходе выполнения лабораторной работы я освоил основы программирования на языке Java, включая компиляцию и запуск программ через командную строку. Я научился создавать методы для выполнения специфических задач, таких как проверка чисел на простоту и проверка строк на палиндром. Выполненные задания позволили углубить понимание циклов и методов, что является основой для дальнейшего изучения программирования на Java.