Федеральное агентство связи

Ордена Трудового Красного знамени

Федеральное государственное бюджетное образовательное учреждение

Высшего образования

«Московский технический университет связи и информатики»

Кафедра Математической кибернетики и информационных технологий

Отчет по лабораторной работе №1

по дисциплине «Объектно ориентированное программирование»

на тему «Основы синтаксиса Java»

Выполнил

Студент группы БФИ1701

Неретин И.Г.

Проверила:

Мосеева М.С.

Москва 2020

1. **Цель работы**

Изучить синтаксис Java и освоить работу с компилятором Java и виртуальной машиной Java, выполнив простые задачи программирования.

1. **Задание**

* Написать программу, которая находит и выводит все простые числа меньше 100;
* Написать программу, которая принимает на вход слова «madam, racecar, apple, kayak, song, noon» через консоль и определяет, являются ли они палиндромами или нет;
* Обе программы скомпилировать и передать в них параметры через командную строку.

1. **Порядок выполнения работы**

**«Простые числа»**

Для выполнения первого задания был написан код, приведенный ниже (Листинг 1). Помимо реализации алгоритма нахождения простых чисел, была опробована возможность вывода русского текста в консоль и в документацию. На этапе компиляции возникала ошибка, которая находила некорректные символы в документации (Рисунок 1).

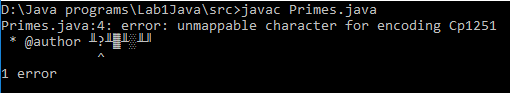


Рисунок – Ошибка с неверными символами в документации

Оказалось, что дело было в неверной кодировке, из-за которой русские символы выводились некорректно. Система Windows (на которой компилировалась программа) по умолчанию использовала кодировку Cp1251, в то время как консоль Windows использует кодировку Cp866. Кроме того, кодировка на этапе компиляции (Cp1251) также оказалась неподходящей. Решением этой проблемы оказалось следующее – следует скомпилировать программу с кодировкой UTF-8, а вывести в консоль с кодировкой Cp866 (чтобы символы выводились корректно). Для других операционных систем, возможно, потребуются другие кодировки.

Листинг – Primes.java

|  |
| --- |
| import java.io.PrintStream;  */\*\*  \* Программа "Простые числа" находит и выводит в консоль простые числа от 2 до 100 включительно.  \** ***@author*** *Иван  \** ***@version*** *1.0  \** ***@since*** *23.03.2020  \* \*\*/* public class Primes {  private static int *MAX\_COUNT\_OF\_FINDING\_PRIMES* = 50; //максимальный делитель  private static int *COUNT\_OF\_NUMBERS* = 100; //количество чисел на входе   //входная точка программы  public static void main(String[] args){   //ввод и проверка кодировки вывода в консоль  String consoleEncoding = System.*getProperty*("consoleEncoding");  if (consoleEncoding != null) {  try {  System.*setOut*(new PrintStream(System.*out*, true, consoleEncoding));  } catch (java.io.UnsupportedEncodingException ex) {  System.*err*.println("Unsupported encoding set for console: "+consoleEncoding);  }  }   //выполнение программы  System.*out*.println("Простые числа:");  for (int i = 2; i <= *COUNT\_OF\_NUMBERS*; i++){  //если число простое, выводим его  if (*IsPrime*(i)){  System.*out*.println(i);  }  }  System.*out*.println("Конец");  }   //функция, определяющая, является ли число простым или нет. Возвращает true, если число простое  public static boolean IsPrime(int x){  for (int i = 2; i < *MAX\_COUNT\_OF\_FINDING\_PRIMES*; i++){  //число не простое, если оно делится на какое-то другое число, но не само на себя  if (((x % i) == 0) && (x > i)){  return false;  }  }  //если делителей числа не нашлось, значит, число простое  return true;  } } |



Рисунок – Компиляция с кодировкой UTF-8

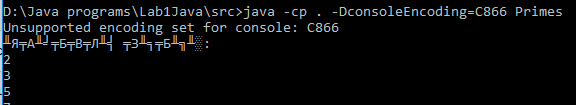


Рисунок – Проверка программы, попытка задать несуществующую кодировку вывода в консоль при запуске. Текст будет выведен некорректно, с кодировкой, установленной по умолчанию

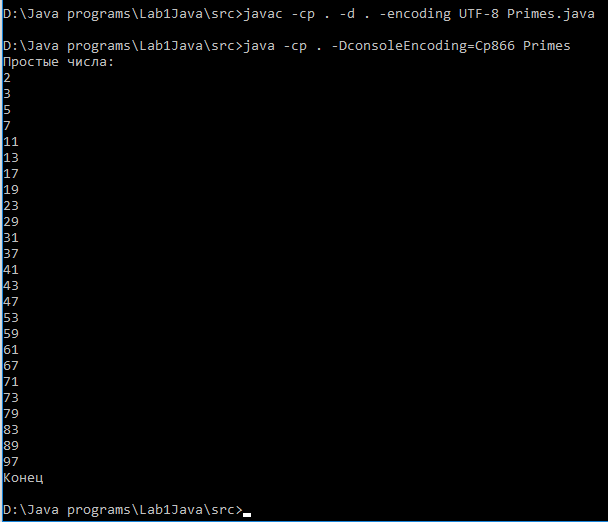


Рисунок – Компиляция с кодировкой UTF-8, вывод с кодировкой Cp866. Корректная работа программы

Выведенные числа соответствуют ряду простых чисел, соответственно, программа работает правильно.

**«Число-палиндром»**

Листинг – Palindrome.java

|  |
| --- |
| import java.io.PrintStream;  */\*\*  \* Программа "Палиндром" принимает на вход строковые значения через пробел и проверяет каждое из них -  \* является ли оно палиндромом или нет.  \** ***@author*** *Иван  \** ***@version*** *1.0  \** ***@since*** *24.03.2020  \* \*\*/* public class Palindrome {  //входная точка программы  public static void main(String[] args) {  *setConsoleEncoding*("Cp866"); //устанавливаем кодировку консоли, чтобы выводить русские символы  for (int i = 0; i < args.length; i++) {  String s = args[i];  if (*isPalindrome*(s)){  System.*out*.println("Слово " + s + " является палиндромом.");  }  else {  System.*out*.println("Слово " + s + " не является палиндромом.");  }  }  }   //метод принимает на вход строку и инвертирует ее  public static String reverseString(String s){  if (s == null) return "";  String result = "";  for (int i = s.length()-1; i >= 0; i--){  result += s.charAt(i);  }  return result;  }   //метод проверяет, является ли строка палиндромом или нет  public static boolean isPalindrome(String s){  String reversedStr = *reverseString*(s);  if (reversedStr.equals(s)){  return true;  }  return false;  }   //установка кодировки Cp866 для консоли. Необходима компиляция с кодировкой UTF-8  public static void setConsoleEncoding(String encoding){  try{  System.*setOut*(new PrintStream(System.*out*, true, encoding));  }  catch(java.io.UnsupportedEncodingException e){  System.*err*.println("Unsupported encoding set for console: " + encoding);  }  } } |

В данном случае был использован простой алгоритм, который проверяет, идентична ли инвертированная строка обычной строке. На этот раз нет необходимости вводить кодировку вывода в консоль, она по умолчанию равна «Cp866». Смена кодировки выведена в отдельную функцию «setConsoleEncoding». В случае необходимости вывода русского текста можно переносить эту функцию из проекта в проект (но важно не забывать о том, что компилировать необходимо в кодировке «UTF-8»).

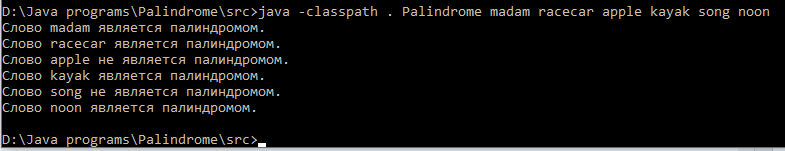


Рисунок – Результат работы программы проверки слов-палиндромов

Все слова были проверены правильно, соответственно, программа работает корректно.

1. **Заключение**

При использовании специальных символов в документации программы или выводимом ею тексте следует помнить о различиях кодировок в разных операционных системах и даже средах, чтобы избежать ошибок и неверного вывода информации, или же использовать только стандартные латинские символы.

В языке java есть возможность скомпилировать код программы прямо через консоль при помощи компилятора java, а затем там же запустить его. Также есть возможность сразу передавать в запускаемую программу параметры, что может быть полезно, если проект насчитывает несколько таких приложений и запускает их одно за другим, передавая им параметры.

1. **Литература**
2. Камаев В.А., Костерин В.В. Технологии программирования. М.: Высшая школа, 2006;
3. Жоголев Е.А.Технология программирования. – М.: Научный мир, 2004;
4. <https://www.google.ru/>
5. <http://www.skipy.ru/technics/encodings_console_comp.html>