


МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Тульский государственный университет»

Институт прикладной математики и компьютерных наук
Кафедра «Прикладная математика и информатика»

Утверждено на заседании кафедры
«Прикладная математика и информатика»
24 января 2022 г., протокол № 5

Заведующий кафедрой

 М.В. Грязев

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
по выполнению лабораторных работ
по дисциплине (модулю)
«Практикум на электронных вычислительных машинах»
Часть 3

основной профессиональной образовательной программы
высшего образования – программы бакалавриата

по направлению подготовки
01.03.02 Прикладная математика и информатика

с направленностью (профилем)
Прикладная математика и информатика

Форма обучения: очная


Идентификационный номер образовательной программы: 010302-01-22

Тула 2022 год

Разработчик методических указаний

Смирнов О.И., доцент каф. ПМиИ, к.ф.-м.н., доцент

(ФИО, должность, ученая степень, ученое звание)



(подпись)

ЛАБОРАТОРНАЯ РАБОТА № 1.

Разработка простейшего консольного приложения на Java с обработкой чисел.

1. Цель и задачи работы

Приобретение навыков подготовки программ, написанных на языке Java.

2. Объекты исследования, оборудование, материалы и наглядные пособия.

При выполнении работ используются ПЭВМ.

3. Задание на работу.

- 1) Создайте первое приложение HelloWorldApp,
- 2) создайте и запустите программу Calculator,
- 3) внесите в программу Calculator необходимые исправления.

4. Порядок выполнения работы.

1. Создайте текстовый файл с именем HelloWorldApp.java и содержащий следующий фрагмент кода:

```
/**
 * The HelloWorldApp class implements an application that
 * simply prints "Hello World!" to standard output.
 */
class HelloWorldApp {
    public static void main(String[] args) {
        System.out.println("Hello World!"); // Display the string.
    }
}
```

2. Скомпилируйте приложение.

Для этого запустите обработчик команд:

Пуск>Выполнить, команда cmd.

Перейдите в директорию с вашим приложением. Для компиляции приложения необходимо выполнить следующую команду

```
javac HelloWorldApp.java
```

Если все выполнено правильно компилятор сгенерирует файл HelloWorldApp.class, содержащий байткод, готовый к запуску на виртуальной машине, в противном случае компилятор выдаст сообщение об ошибке.

3. Запустите приложение.

Для этого в этой же директории выполните команду java HelloWorldApp.

4. Создать и запустить программу Calculator:

```
public class Calculator {

    public static final char ADD = 43;
    public static final char SUBTRACT = 45;
    public static final char MULTIPLY = 42;
    public static final char DIVIDE = 47;

    public static void main(String[] args) {
```

```

        new Calculator();
    }

    public Calculator() {
        Scanner console = new Scanner(System.in);

        while (true) {
            System.out.print("First Number: ");
            double num1 = console.nextDouble();
            System.out.print("Second Number: ");
            double num2 = console.nextDouble();
            System.out.print("Operator (+-/*) : ");
            String input = console.next("[+-/*]");
            char operator = input.charAt(0);
            double result = Double.NaN;

            switch (operator) {
                case ADD:
                    result = num1 + num2;
                    break;
                case SUBTRACT:
                    result = num1 - num2;
                    break;
                case MULTIPLY:
                    result = num1 * num2;
                    break;
                case DIVIDE:
                    result = num1 / num2;
                    break;
            }

            System.out.println("" + num1 + " " + operator + " " +
num2 + " = " + result + "\n");
        }
    }
}

```

5. Исправить программу так, чтобы после каждой итерации выводился запрос на повторное вычисление.

6. Исправить программу так, чтобы операнды и операция задавались как параметры командной строки.

7. Внести другие изменения по требованию преподавателя.

5. Содержание отчета.

Отчет должен содержать:

- 1) титульный лист,
- 2) цель работы,
- 3) задание,
- 4) исходные тексты программ,
- 5) результаты работы программ.

ЛАБОРАТОРНАЯ РАБОТА № 2.

Использование массивов и файлового ввода-вывода в программах на Java.

1. Цель и задачи работы

Получение практических навыков по работе с массивами, работе с файлами, а также по использованию механизмов ввода-вывода.

2. Объекты исследования, оборудование, материалы и наглядные пособия.

При выполнении работ используются ПЭВМ.

3. Задание на работу.

Создать текстовый файл со значениями переменной x и заданной функции $f(x)$, считать значения из файла в массив (массивы), вывести их на экран. Изучить механизм сериализации.

Варианты заданий.

1. $\sin(x)$
2. $\cos(x)$
3. $\operatorname{tg}(x)$
4. $\operatorname{ctg}(x)$
5. $\arcsin(x)$
6. $\arccos(x)$
7. $\operatorname{arctg}(x)$
8. $\operatorname{arcctg}(x)$
9. $\sec(x)$
10. $\operatorname{cosec}(x)$
11. $\operatorname{arcsec}(x)$
12. $\operatorname{arccosec}(x)$
13. $\exp(x)$
14. $\ln(x)$
15. $\log[2](x)$
16. 2^x
17. 3^{-x}
18. $\log[1/3](x)$
19. \sqrt{x}
20. $\sqrt[3]{x}$

4. Порядок выполнения работы.

1. Создать с помощью программы на Java текстовый файл, содержащий несколько значений переменной x и функции $f(x)$.

2. Создать двухмерный массив типа `double` (или два массива) и прочитать в него значения из файла.

3. Вывести значения массива(-ов) на экран.

4. При помощи механизма сериализации сохранить созданный массив целиком в новый файл.

5. Создать новый массив чисел типа `double` и при помощи механизма сериализации прочитать массив из второго файла.

6. Прodelать те же операции, но выполняя сериализацию для каждого элемента массива.

7. Вывести все значения элементов созданных массивов на экран для сравнения.

5. Содержание отчета.

Отчет должен содержать:

- 1) титульный лист,
- 2) цель работы,
- 3) задание,
- 4) исходный текст программы,
- 5) результаты работы программы.

ЛАБОРАТОРНАЯ РАБОТА № 3.

Разработка и демонстрация использования специального класса пользователя.

1. Цель и задачи работы

Получить практические навыки по работе с классами и объектами на платформе java. Научиться проводить декомпозицию приложений с использованием пакетов.

2. Объекты исследования, оборудование, материалы и наглядные пособия.

При выполнении работ используются ПЭВМ.

3. Задание на работу.

Разработать указанные классы, соблюдая принципы ООП.

4. Порядок выполнения работы.

1. Реализовать класс, представляющий игральную карту. Игральная карта характеризуется двумя параметрами: масть и ранг. **Требования:** все масти и ранги карт заданы числовыми константами, все атрибуты объекта имеют модификатор доступа private, реализовать методы проверки правильности ранга и масти, преобразования ранга в строку, преобразование масти в строку, класс должен находиться в пакете deck.

2. Реализовать класс, представляющий колоду карт. **Требования:** реализовать методы инициализации колоды карт и выборки карты из колоды, класс должен находиться в пакете playingcards.

3. Реализовать класс, демонстрирующий работоспособность первых двух классов – создание колоды и отображение всех карт. Кроме этого, этот класс должен демонстрировать различные виды импорта, включая статический.

5. Содержание отчета..

Отчет должен содержать:

- 1) титульный лист,
- 2) цель работы,
- 3) задание,
- 4) исходный текст программы,
- 5) результаты работы программы.

ЛАБОРАТОРНАЯ РАБОТА № 4.

Использование системы обработки исключений для контроля конфликтных ситуаций.

1. Цель и задачи работы

Научиться обрабатывать ошибки с помощью исключений.

2. Объекты исследования, оборудование, материалы и наглядные пособия.

При выполнении работ используются ПЭВМ.

3. Задание на работу.

Основываясь на приложении, разработанном на лабораторной работе № 3, продемонстрировать обработку ошибок с помощью исключений.

4. Порядок выполнения работы.

1. Продумать ситуации, когда возможно возникновение исключительных ситуаций.

2. Разработать код обработки ошибок с использованием собственных исключений.

3. Разработать код обработки ошибок с использованием блоков `try`, `catch`, `finally`.

4. Реализовать самостоятельное выбрасывание исключения и выбрасывание исключения системой времени выполнения `java`.

5. Содержание отчета..

Отчет должен содержать:

- 1) титульный лист,
- 2) цель работы,
- 3) задание,
- 4) исходный текст программы,
- 5) результаты работы программы.

ЛАБОРАТОРНАЯ РАБОТА № 5.
Разработка простейшего графического приложения
с использованием пакета Java.awt.

1. Цель и задачи работы

Получение практических навыков по использованию графических возможностей языка Java (библиотека AWT).

2. Объекты исследования, оборудование, материалы и наглядные пособия.

При выполнении работ используются ПЭВМ.

3. Задание на работу.

Разработать программу, содержащую диалоговые элементы для ввода параметров фигуры, вывод произвольных изображения.

ВАРИАНТЫ ЗАДАНИЙ.

Изобразить следующую фигуру:

- 1) окружность,
- 2) Эллипс,
- 3) Гипербола,
- 4) Парабола,
- 5) Парабола Нейля,
- 6) Локон Аньези,
- 7) Конхоида Никомеда,
- 8) Циссоида Диоклеса,
- 9) Лемниската Бернулли,
- 10) Овал Кассини для $a > c$,
- 11) Овал Кассини для $a < c$,
- 12) Строфоида,
- 13) "Крест"
- 14) Кардиоида,
- 15) Трисектриса,
- 16) Астроида,
- 17) Декартов лист,
- 18) Улитка Паскаля для $a > b$,
- 19) Улитка Паскаля для $a < b$,
- 20) Цепная линия,
- 21) Спираль Архимеда,
- 22) Параболическая спираль,
- 23) Логарифмическая спираль,
- 24) Укороченная циклоида,
- 25) Удлиненная циклоида,
- 26) Эпициклоида,
- 27) Гипоциклоида,
- 28) Трактриса.

4. Порядок выполнения работы.

1. Создайте приложение, отображающее изображение из файла.
2. Создайте приложение, отображающее диалоговое окно. В окне должны присутствовать поля ввода текста и одна кнопка. Для полей ввода должны быть предусмотрены подписи.
3. При нажатии на кнопку в диалоговом окне должно открыться новое окно, отображающее заданную фигуру с заданными параметрами.

5. Содержание отчета.

Отчет должен содержать:

- 1) титульный лист,
- 2) цель работы,
- 3) задание,
- 4) алгоритм решения задачи,
- 5) исходный текст программы,
- 6) результаты работы программы.

ЛАБОРАТОРНАЯ РАБОТА № 6.

Разработка многопоточного приложения на Java.

1. Цель и задачи работы

Получение практических навыков по использованию сетевых возможностей языка Java, созданию многопоточных приложений на языке Java.

2. Объекты исследования, оборудование, материалы и наглядные пособия.

При выполнении работ используются ПЭВМ.

3. Задание на работу.

В ходе выполнения лабораторной работы необходимо создать два приложения - прокси-сервер и клиент.

4. Порядок выполнения работы.

1. Создать приложение прокси-сервер.

Прокси-сервер получает запрос по протоколу HTTP/1.0 и возвращает клиенту содержимое страницы. Поддержка протокола должна быть достаточной для того, чтобы данный прокси-сервер можно было использовать в любом браузере (минимум в Internet Explorer, без авторизации). Пример работы с прокси-сервером (на PHP) можно посмотреть здесь:

<http://php.novgorod.ru/manual/ru/function.fopen.php>

2. Создать приложение клиент.

Клиент получает от пользователя адрес страницы, адрес прокси-сервера и его порт в качестве параметров командной строки. Клиент отправляет запрос прокси-серверу, получает содержимое страницы и выводит его на экран. В случае возникновения ошибки (невозможность подключиться к прокси-серверу, невозможность загрузки страницы и т.д.) отображается соответствующее сообщение.

3. Обеспечить работу прокси-сервера в многопоточном режиме, т.е. подключение к серверу более одного клиента одновременно. Продемонстрировать данную возможность.

5. Содержание отчета.

Отчет должен содержать:

- 1) титульный лист,
- 2) цель работы,
- 3) задание,
- 4) тексты приложений,
- 5) результаты работы приложений.

Список библиографических источников

1. Шилдт, Schildt G. Искусство программирования на Java / Г.Шилдт, Д.Холмс;пер.с англ.и ред. Г.В. Галисеева .– М.и др. : Вильямс, 2005 .– 331с.
2. Вязовик, Н.А. Программирование на Java : Курс лекций для вузов / Н.А.Вязовик .– М., 2003 .– 592с.
3. Вязовик, Н.А. Программирование на Java. Дистанционный учебный курс. Интернет-университет информационных технологий – INTUIT.ru, 2009.