

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Шебзузов Татьяна Александровна

Должность: Директор Пятигорского института (филиал) Северо-Кавказского
федерального университета

Дата подписания: 12.09.2023 10:50:21

Уникальный программный ключ:

d74ce93cd40e39275c3ba2f58486412a1c8ef96f

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Пятигорский институт (филиал) СКФУ

Методические указания

по выполнению лабораторных работ

по дисциплине «**ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ**»

для студентов направления подготовки /специальности

09.03.02 Информационные системы и технологии

(ЭЛЕКТРОННЫЙ ДОКУМЕНТ)

СОДЕРЖАНИЕ

Цель и задачи освоения дисциплины	3
Место дисциплины в структуре основной образовательной программы	3
Связь с предшествующими дисциплинами	3
Связь с последующими дисциплинами	3
5. НАИМЕНОВАНИЕ ЛАБОРАТОРНЫХ РАБОТ	3
СОДЕРЖАНИЕ ЛАБОРАТОРНЫХ РАБОТ	4
7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ	9

1 Цель и задачи освоения дисциплины

Целью освоения дисциплины «Объектно-ориентированное программирование» является формирование у студентов представления о технологиях объектно-ориентированного и визуального программирования, получение устойчивых навыков самостоятельного программирования с применением современных программных средств разработки прикладных программ.

В соответствии с указанной целью при изучении дисциплины ставятся следующие задачи:

- создать теоретическую базу, ознакомив обучающихся с основными понятиями, возможностями, особенностями и преимуществами объектно-ориентированной технологии программирования;
- привить навыки работы в среде объектно-ориентированного программирования;
- дать сведения о принципах, технологиях и этапах разработки программных приложений.

2 Место дисциплины в структуре основной образовательной программы

Дисциплина «Объектно-ориентированное программирование» входит в вариативную часть блока Б1 ОП ВО подготовки бакалавра направления 09.03.02 Информационные системы и технологии. Ее освоение происходит в 3 семестре.

3 Связь с предшествующими дисциплинами

Дисциплины, необходимые для успешного изучения данной дисциплины: «Технологии программирования».

4 Связь с последующими дисциплинами

Знания, полученные при изучении данной дисциплины, необходимы для успешного освоения таких дисциплин, как «Разработка программных приложений», «Мультимедиа технологии», «Инструментальные средства мультимедиа технологий», «Технологии разработки web-приложений», «Проектный практикум», «Подготовка к сдаче и сдача государственного экзамена».

5. НАИМЕНОВАНИЕ ЛАБОРАТОРНЫХ РАБОТ

№ темы	Наименование тем дисциплины, их краткое содержание	Объем часов	Интерактивная форма проведения
3 семестр			
2-3	Лабораторная работа 1. Среда разработки Visual Studio.Net. Инструменты Visual Studio.NET. Реализация алгоритмов линейной и разветвляющейся структуры.	1,5	
4	Лабораторная работа 2. Реализация алгоритмов циклической структуры.	1,5	Решение проблемных задач
5	Лабораторная работа 3. Работа с одномерными и двумерными массивами.	1,5	
6	Лабораторная работа 4. Создание методов (подпрограмм).	1,5	Решение проблемных задач

1,6	Лабораторная работа 5. Создание классов.	1,5	Решение проблемных задач
6	Лабораторная работа 6. Наследование классов.	1,5	Решение проблемных задач
6	Лабораторная работа 7. Полиморфизм методов.	1,5	
7	Лабораторная работа 8. Работа с текстовыми файлами и обработка строк.	1,5	
8, 9	Лабораторная работа 9. Методы C# для работы с датами.	1,5	Решение проблемных задач
	Итого за 3 семестр	13,5	
	Итого	13,5	

6 СОДЕРЖАНИЕ ЛАБОРАТОРНЫХ РАБОТ

Лабораторная работа № 1. Среда разработки Visual Studio.Net.

Инструменты Visual Studio.NET. Элементы управления и их свойства.

Форма проведения: лабораторная работа в компьютерном классе.

Ход лабораторной работы:

1. В среде Visual Studio создать консольное приложение на языке C#, которое будет вычислять значение функции в соответствии с номером варианта.
2. Предусмотреть ввод исходных данных с клавиатуры.
3. Предусмотреть проверку допустимости исходных данных.

Вопросы для обсуждения:

1. Стили программирования.
2. Парадигмы программирования.
3. История ООП.
4. Преимущества ООП.
5. Основные понятия ООП.
6. Основные принципы ООП.
7. Основные инструменты (окна) Visual Studio и их назначение. Закрепление, перемещение и скрытие окон инструментов.
8. Какие файлы создаются при сохранении решения?
9. Как и для чего создается исполняемый файл приложения?
10. Основные типы данных Visual C#.
11. Операторы объявления локальных и глобальных переменных.
12. Работа с математическими методами среды .NET Framework.
13. Инструкция выбора if.

Работа с литературой:

Рекомендуемые источники информации (№ источника)			
Основная	Дополнительная	Методическая	Интернет-ресурсы
1	1-2	1	1-5

Оценочные средства: отчет по лабораторной работе (см.: Фонд оценочных средств)

Лабораторная работа № 2. Реализация алгоритмов циклической структуры.

Форма проведения: лабораторная работа в компьютерном классе.

Ход лабораторной работы:

1. Разработать программу на C# в соответствии с вариантом задания, используя цикл с заданным числом повторений (оператор for) и с неизвестным числом повторений (операторы while и do ... while).

Вопросы для обсуждения:

1. Назначение и синтаксис оператора цикла for.
2. Назначение и синтаксис оператора цикла while .
3. Назначение и синтаксис оператора цикла do ... while.
4. Инструкция выбора switch.
5. Использование управляющих символов и форматных строк.

Работа с литературой:

Рекомендуемые источники информации (№ источника)			
Основная	Дополнительная	Методическая	Интернет-ресурсы
1	1-2	1	1-5

Оценочные средства: отчет по лабораторной работе (см.: Фонд оценочных средств)

Лабораторная работа № 3. Работа с одномерными и двумерными массивами.

Форма проведения: лабораторная работа в компьютерном классе.

Ход лабораторной работы:

1. В среде Visual Studio создать консольное приложение на языке C# для решения задачи обработки одномерных массивов в соответствии с номером варианта.
2. Предусмотреть ввод элементов массива с клавиатуры.

Вопросы для обсуждения:

1. Способы объявления и инициализации одномерных и двумерных массивов.
2. Какие инструкции используются для перебора элементов массива.
3. Свойства класса Array.
4. Методы класса Array.

Работа с литературой:

Рекомендуемые источники информации (№ источника)			
Основная	Дополнительная	Методическая	Интернет-ресурсы
1	1-2	1	1-5

Оценочные средства: отчет по лабораторной работе (см.: Фонд оценочных средств)

Лабораторная работа № 4. Создание методов (подпрограмм).

Форма проведения: лабораторная работа в компьютерном классе.

Ход лабораторной работы:

1. Даны два одномерных и два двумерных массива целых чисел. Написать программу, содержащую процедуры (методы) для выполнения операций с массивами в соответствии с номером варианта.
2. Применить методы к каждому из двух массивов соответствующей размерности.

Вопросы для обсуждения:

1. Что такое подпрограмма, и какие преимущества дает использование подпрограмм?
2. Общая форма описания подпрограммы (метода). Структура заголовка и тела подпрограммы.
3. Какие типы данных может возвращать подпрограмма? Какой оператор используется для возвращения значения подпрограммы? В каких случаях используется служебное слово `void`?
4. В чем разница между фактическими и формальными параметрами подпрограммы?
5. Какие способы передачи параметров в подпрограммах предусматривает синтаксис языка C#? Какие ключевые слова при этом используются?
6. Как можно изменить значение фактического параметра подпрограммы в теле самой подпрограммы?
7. Можно ли использовать подпрограмму без параметров?
8. Обязательно ли инициализировать значения фактических параметров подпрограммы перед ее вызовом?

Работа с литературой:

Рекомендуемые источники информации (№ источника)			
Основная	Дополнительная	Методическая	Интернет-ресурсы
1	1-2	1	1-5

Оценочные средства: отчет по лабораторной работе (см.: Фонд оценочных средств)

Лабораторная работа № 5. Создание классов.

Форма проведения: лабораторная работа в компьютерном классе.

Ход лабораторной работы:

1. Разработать консольное приложение на C# в среде Visual Studio для решения задачи в соответствии номером варианта. Создать класс. Поля и методы класса перечислены в задании.
2. Доступ к значениям полей должен осуществляться с использованием общедоступных операций `set` и `get`.
3. Включить в класс методы, обеспечивающие:
 - заполнение полей данных;
 - отображение (вывод) полей данных;
 - конструктор, который инициализирует поля класса.
4. Главная программа должна демонстрировать выполнение всех методов класса.

Вопросы для обсуждения:

1. Принципы ООП. Инкапсуляция. Наследование. Полиморфизм.
2. Объявление класса в C#. Модификаторы доступа класса.
3. Члены класса и их уровни доступа.
4. Назначение и описание свойств объектов класса.
5. Назначение и описание методов класса.
6. Назначение и описание конструкторов класса. Перегрузка конструкторов.
7. Назначение и описание инициализаторов класса.
8. Деструкторы и «сборка мусора» в C#.

Работа с литературой:

Рекомендуемые источники информации (№ источника)			
Основная	Дополнительная	Методическая	Интернет-ресурсы
1	1-2	1	1-5

Оценочные средства: отчет по лабораторной работе (см.: Фонд оценочных средств)

Лабораторная работа № 6. Наследование классов.

Форма проведения: лабораторная работа в компьютерном классе.

Ход лабораторной работы:

1. Разработать два варианта проектов классов в соответствии с вариантом задания:
– проекты независимых друг от друга классов; – проекты зависимых классов с помощью наследования.
2. В каждом классе предусмотреть:
– поля и методы, указанные в задании;
– доступ к значениям полей с использованием свойств (set и get);
– как минимум, два конструктора для инициализации полей класса;
– метод для вывода значений полей класса.
3. Разработать два консольных приложения на C# в среде Visual Studio, реализующих каждый из вариантов проектов классов.
4. Главная программа (program) должна демонстрировать выполнение всех методов класса.

Вопросы для обсуждения:

1. Назначение наследования классов.
2. Понятие базового и производного классов. Могут ли наследоваться закрытые поля?
3. Синтаксис объявления производного класса в C#.
4. Свойства наследования.
5. Использование ключевого слова sealed.
6. Конструкторы и наследование. Форма расширенного объявления конструктора.
7. Принципы действия ключевого слова base.
8. Наследование и сокрытие имен.

Лабораторная работа № 7. Полиморфизм методов.

Форма проведения: лабораторная работа в компьютерном классе.

Ход лабораторной работы:

1. Для классов, созданных в лабораторной работе № 6, предусмотреть переопределение методов в производном классе.
2. Реализовать по два конструктора в базовом и производном классах.

Вопросы для обсуждения:

1. Принцип полиморфизма в ООП.
2. Переопределение методов.
3. Абстрактные классы.
4. Определение абстрактного метода.
5. Наследование абстрактного класса.
6. Перегрузка методов.
7. Чем перегрузка отличается от переопределения метода?

Лабораторная работа № 8. Работа с текстовыми файлами и обработка строк.

Форма проведения: лабораторная работа в компьютерном классе.

Ход лабораторной работы:

1. Написать программу, которая позволяет зашифровывать и расшифровывать с помощью «шифра Цезаря» текстовый файл или сообщение, написанное на русском языке. Этот шифр реализует следующее преобразование текста: каждая русская буква исходного текста заменяется третьей после нее буквой в алфавите, который считается написанным по кругу.
2. Знаки препинания и другие символы, отличные от строчных или прописных букв русского алфавита, не шифруются.

Вопросы для обсуждения:

1. Какие методы .NET Framework используются для загрузки в текстовое поле содержимого текстового файла?
2. Какие действия необходимы для создания нового текстового файла на диске?
3. Назначение и синтаксис методов **OpenRead** и **OpenWrite** класса **System.IO.File**.
4. Назначение и синтаксис методов **ReadAllText** и **ReadAllLines** класса **System.IO.File**.
5. Назначение и синтаксис методов **WriteAllText** и **WriteAllLines** класса **System.IO.File**.
6. Назначение и синтаксис методов **Copy**, **Move**, **Delete**, **Exists** класса **System.IO.File**.
7. Назначение метода .NET Framework **Length**.
8. Назначение и синтаксис метода .NET Framework **Substring**.
9. Назначение и синтаксис метода .NET Framework **IndexOf**.
10. Назначение и синтаксис метода .NET Framework **Remove**.
11. Назначение и синтаксис метода .NET Framework **Insert**.
12. Назовите основные кодировки символов и их особенности.
13. Как средствами C# получить юникод данного символа и символ по данному юникоду?
14. Какие действия необходимы для шифрования текстового файла?
15. Какие действия необходимы для расшифровки текстового файла?

Работа с литературой:

Рекомендуемые источники информации (№ источника)			
Основная	Дополнительная	Методическая	Интернет-ресурсы
1	1-2	1	1-5

Оценочные средства: отчет по лабораторной работе (см.: Фонд оценочных средств)

Лабораторная работа № 9. Методы C# для работы с датами.

Форма проведения: лабораторная работа в компьютерном классе.

Ход лабораторной работы:

1. Написать программу «Расчет заработной платы», которая для заданного количества сотрудников вычисляет сумму к выдаче.
2. Программа должна вычислять для каждого сотрудника: начисленную сумму; сумму удержаний; сумму к выдаче.
3. Программа должна формировать итоговую таблицу расчета заработной платы по всем сотрудникам, содержащую следующие поля: ФИО; Оклад; Стаж; Возраст; Доплата; Начислено; Удержано; К выдаче.

4. Считать, что заработная плата рассчитывается за текущий месяц.
5. Доплата за стаж работы на данном предприятии вычисляется следующим образом: если стаж составляет от 0 до 5 лет, то доплата равна 10% от оклада; если стаж составляет от 5 до 10 лет, то доплата равна 30% от оклада; если стаж равен или больше 10 лет, то доплата равна 50% от оклада.
6. Удерживается со всех сотрудников 13% от начисленной суммы (НДФЛ).

Вопросы для обсуждения:

1. Перечислите структуры C# для работы с датами и временем.
2. Перечислите способы создания экземпляра TimeSpan.
3. Какие операции допустимы над экземплярами TimeSpan?
4. Свойства TimeSpan.
5. В чем отличие структур DateTime и DateTimeOffset?
6. Создание экземпляров DateTime и DateTimeOffset.
7. Текущая дата и время.
8. Свойства DateTime и DateTimeOffset.
9. Методы DateTime и DateTimeOffset.
10. Операции над DateTime и DateTimeOffset.
11. Преобразование строки в дату и время.
12. Преобразование даты в строку.
13. Стандартные форматные строки для даты, чувствительные к культуре.
14. Специальные форматные строки для даты и времени.

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

7.1. Рекомендуемая литература

7.1.1. Основная литература

1. Мейер Б. Объектно-ориентированное программирование и программная инженерия [Электронный ресурс]/ Мейер Б.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 285 с. — Режим доступа: <http://www.iprbookshop.ru/39552>.— ЭБС «IPRbooks».

7.1.2. Дополнительная литература

1. Зыков, С.В. Введение в теорию программирования. Объектно-ориентированный подход / С.В. Зыков. - 2-е изд., испр. - М.: Национальный Открытый Университет «ИНТУИТ», 2016. - 189 с. [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429073>.
2. Битюцкая Н.И. Разработка программных приложений: лабораторный практикум. – Ставрополь: Изд-во СКФУ, 2015. – 140 с.

7.1.3. Методическая литература

1. Методические рекомендации для студентов по организации самостоятельной работы по дисциплине «Объектно-ориентированное программирование»

7.1.4. Интернет-ресурсы

1. <http://www.intuit.ru> – сайт дистанционного образования в области информационных технологий
2. <http://www.iprbookshop.ru> – ЭБС «IPRbooks».
3. <http://www.biblioclub.ru> – университетская библиотека онлайн
4. <http://window.edu.ru> – система федеральных образовательных порталов. Каталоги, библиотеки, форумы, законы, документы, стандарты

5. <http://www.iqlib.ru> - интернет библиотека образовательных изданий, в которой собраны электронные учебники, справочные и учебные пособия.