

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Шебзухова Татьяна Александровна
Должность: Директор Пятигорского института (филиал) Северо-Кавказского
федерального университета
Дата подписания: 19.07.2023 11:14:41
Уникальный программный ключ:
d74ce93cd40e39275c3ba2f58486412a1c8ef96f

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
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 направления подготовки **09.03.02 Информационные системы и
технологии**
направленность (профиль) **Информационные системы и технологии
обработки цифрового контента**

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат: 2C0000043E9AB8B952205E7BA500060000043E
Владелец: Шебзухова Татьяна Александровна
Действителен: с 19.08.2022 по 19.08.2023

**Пятигорск
2023**

ВВЕДЕНИЕ

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. Наименование лабораторных работ

№ Темы дисциплины	Наименование тем дисциплины, их краткое содержание	Объем часов	Из них практическая подготовка, часов
5 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	
5	Лабораторная работа 3. Работа с одномерными и двумерными массивами.	1,5	
6	Лабораторная работа 4. Создание методов (подпрограмм).	1,5	
1,6	Лабораторная работа 5. Создание классов.	1,5	
6	Лабораторная работа 6. Наследование классов.	1,5	
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	
	Итого	13,5	

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 2C0000043E9AB8B952205E7BA500060000043E
Владелец: Шебзухова Татьяна Александровна

Действителен: с 19.08.2022 по 19.08.2023

СТРУКТУРА И СОДЕРЖАНИЕ ЛАБОРАТОРНЫХ ЗАНЯТИЙ

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.

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат: 88296318-88296318-06000000
Владелец: Шебзухова Татьяна Александровна
Действителен: с 19.08.2022 по 19.08.2023

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).

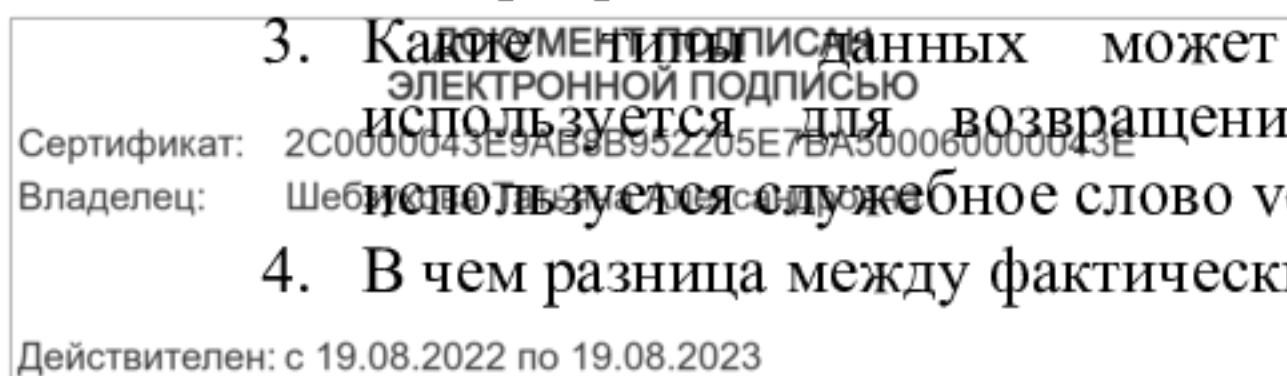
Форма проведения: лабораторная работа в компьютерном классе.

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

Сертификат: 2C0000043E9AB8B952205E7BA500060000043E
 Владелец: Шебзухова Татьяна Александровна
 Действителен: с 19.08.2022 по 19.08.2023

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:

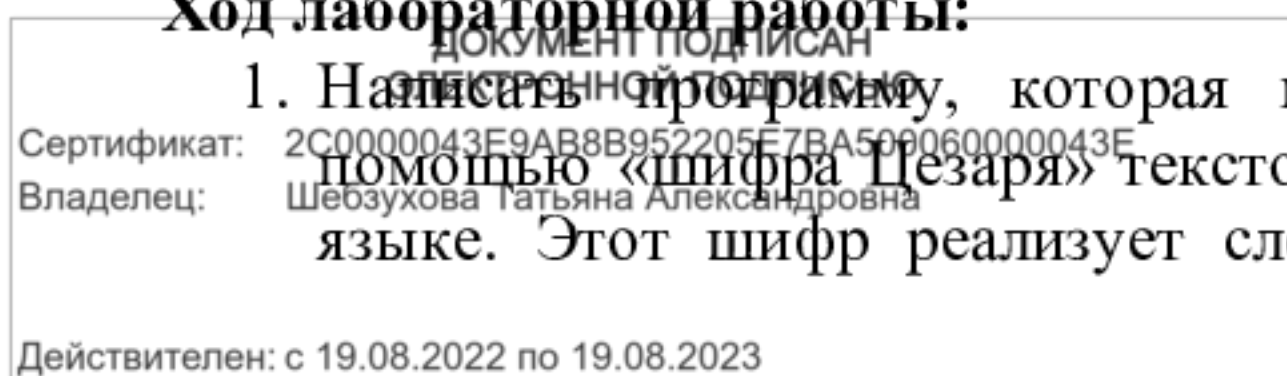
1. Принцип полиморфизма в ООП.
2. Переопределение методов.
3. Абстрактные классы.
4. Определение абстрактного метода.
5. Наследование абстрактного класса.
6. Перегрузка методов.
7. Чем перегрузка отличается от переопределения метода?

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).

Документ подписан
электронной подписью
Сертификат: 2C0000043E9AB8B952205E7BA500060000043E
Владелец: Шебухова Татьяна Александровна

Действителен: с 19.08.2022 по 19.08.2023

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.

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 2C0000043E9AB8B952205E7BA500060000043E
Владелец: Шебзухова Татьяна Александровна

Действителен: с 19.08.2022 по 19.08.2023

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Пятигорский институт (филиал) СКФУ

Методические указания

для обучающихся по организации и проведению самостоятельной работы
по дисциплине «**ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ**»
для студентов направления подготовки **09.03.02 Информационные системы
и технологии**
направленность (профиль) **Информационные системы и технологии
обработки цифрового контента**

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 2C0000043E9AB8B952205E7BA500060000043E
Владелец: Шебзухова Татьяна Александровна

Действителен: с 19.08.2022 по 19.08.2023

Пятигорск, 2023

СОДЕРЖАНИЕ

1. Общие положения	3
2. Цель и задачи самостоятельной работы	4
3. Технологическая карта самостоятельной работы студента	4
4. Порядок выполнения самостоятельной работы студентом	5
4.1. Методические рекомендации по работе с учебной литературой	5
4.2. Методические рекомендации по подготовке к практическим занятиям	6
4.3. Методические рекомендации по самопроверке знаний	7
4.4. Методические рекомендации по написанию научных текстов (докладов, рефератов, эссе, научных статей и т.д.)	8
4.5. Методические рекомендации по подготовке к зачетам	10
Список литературы для выполнения СРС	10

1.

1. Общие положения

Самостоятельная работа – планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Самостоятельная работа студентов (СРС) в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К основным видам самостоятельной работы студентов относятся:

- формирование и усвоение содержания конспекта лекций на базе рекомендованной лектором учебной литературы, включая информационные образовательные ресурсы (электронные учебники, электронные библиотеки и др.);
- написание докладов;
- подготовка к семинарам, практическим и лабораторным работам, их оформление;
- составление аннотированного списка статей из соответствующих журналов по отраслям знаний (педагогических, психологических, методических и др.);
- выполнение учебно-исследовательских работ, проектная деятельность;
- подготовка практических разработок и рекомендаций по решению проблемной ситуации;
- выполнение домашних заданий в виде решения отдельных задач, проведения типовых расчетов, расчетно-компьютерных и индивидуальных работ по отдельным разделам содержания дисциплин и т.д.;
- компьютерный текущий самоконтроль и контроль успеваемости на базе электронных обучающих и аттестующих тестов;
- выполнение курсовых работ (проектов) в рамках дисциплин;
- выполнение выпускной квалификационной работы и др.

Методика организации самостоятельной работы студентов зависит от структуры, характера и особенностей изучаемой дисциплины, объема часов на ее изучение, вида заданий для самостоятельной работы студентов, индивидуальных качеств студентов и условий учебной деятельности.

Процесс организации самостоятельной работы студентов включает в себя следующие этапы:

- подготовительный (определение целей, составление программы, подготовка методического обеспечения, подготовка оборудования);
- основной (реализация программы, использование приемов поиска информации, усвоения, переработки, применения, передачи знаний, фиксирование результатов, самоорганизация процесса работы);
- заключительный (оценка значимости и анализ результатов, их систематизация, оценка эффективности программы и приемов работы, выводы о направлениях оптимизации труда).

2. Цель и задачи самостоятельной работы

Ведущая цель организации и осуществления СРС совпадает с целью обучения студента – формирование универсальных компетенций.

При организации СРС важным и необходимым условием становятся формирование умения самостоятельной работы для приобретения знаний, навыков и возможности организации учебной и научной деятельности. Целью самостоятельной работы студентов является овладение фундаментальными знаниями, профессиональными умениями и навыками деятельности по профилю, опытом творческой, исследовательской деятельности. Самостоятельная работа студентов способствует развитию самостоятельности, ответственности и организованности, творческого подхода к решению проблем учебного и профессионального уровня.

Задачами СРС являются:

- систематизация и закрепление полученных теоретических знаний и практических умений студентов;
- углубление и расширение теоретических знаний;
- формирование умений использовать нормативную, правовую, справочную документацию и специальную литературу;
- развитие познавательных способностей и активности студентов: творческой инициативы, самостоятельности, ответственности и организованности;
- формирование самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;
- развитие исследовательских умений;
- использование материала, собранного и полученного в ходе самостоятельной работы и лабораторных занятий.

3. Технологическая карта самостоятельной работы студента

Коды реализуемых компетенций, индикатора(ов)	Вид деятельности студентов	Средства и технологии оценки	Объем часов, в том числе		
			СРС	Контактная работа с преподавателем	Всего
5 семестр					
ИД-1 _{ПК-5} , ИД-2 _{ПК5} ИД-1 _{ПК-9} , ИД-2 _{ПК9} ИД-1 _{ПК-10} , ИД-2 _{ПК-10} ИД-1 _{ПК-13} , ИД-2 _{ПК13}	Самостоятельное изучение литературы	Собеседование	48,1	5,35	53,45
ИД-1 _{ПК-5} , ИД-2 _{ПК5} ИД-1 _{ПК-9} , ИД-2 _{ПК9} ИД-1 _{ПК-10} , ИД-2 _{ПК-10} ИД-1 _{ПК-13} , ИД-2 _{ПК13}	Подготовка к лабораторным занятиям	Собеседование	3,64	0,41	4,05
ИД-1 _{ПК-5} , ИД-2 _{ПК5} ИД-1 _{ПК-9} , ИД-2 _{ПК9} ИД-1 _{ПК-10} , ИД-2 _{ПК-10} ИД-1 _{ПК-13} , ИД-2 _{ПК13}	Подготовка доклада	Доклад	9	1	10
Итого за 5 семестр			60,75	6,75	67,5

Сертификат
Владелец:

20000043E9A9D92205E7BA500060000043E
Шебзухова Татьяна Александровна

Действителен: с 19.08.2022 по 19.08.2023

	Итого	60,75	6,75	67,5
--	-------	-------	------	------

4. Порядок выполнения самостоятельной работы студентом

4.1. Методические рекомендации по работе с учебной литературой

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги.

Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил.

Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу.

Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода).

При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа.

Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем.

Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались.

Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того на сколько осознанно читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Выделяют **четыре основные установки в чтении научного текста**:

информационно-поисковый (задача – найти, выделить искомую информацию)

усваивающая (усилия читателя направлены на то, чтобы как можно полнее осознать и запомнить как сами сведения излагаемые автором, так и всю логику его рассуждений)

аналитико-критическая (читатель стремится критически осмыслить материал, проанализировав его, определив свое отношение к нему)

творческая (создает у читателя готовность в том или ином виде – как отправной пункт для своих рассуждений, как образ для действия по аналогии и т.п. – использовать суждения автора, ход его мыслей, результат наблюдения, разработанную методику, дополнить их, подвергнуть новой проверке).

Основные виды систематизированной записи прочитанного:

Аннотирование – предельно краткое связное описание просмотренной или прочитанной книги (статьи), ее содержания, источников, характера и назначения;

Планирование – краткая логическая организация текста, раскрывающая содержание и структуру изучаемого материала;

Тезирование – лаконичное воспроизведение основных утверждений автора без привлечения фактического материала;

Цитирование – дословное выписывание из текста выдержек, извлечений, наиболее существенно отражающих ту или иную мысль автора;

Конспектирование – краткое и последовательное изложение содержания прочитанного.

Конспект – сложный способ изложения содержания книги или статьи в логической последовательности. Конспект аккумулирует в себе предыдущие виды записи, позволяет всесторонне охватить содержание книги, статьи. Поэтому умение составлять план, тезисы, делать выписки и другие записи определяет и технологию составления конспекта.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта.

2. Выделите главное, составьте план.

3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора.

4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли.

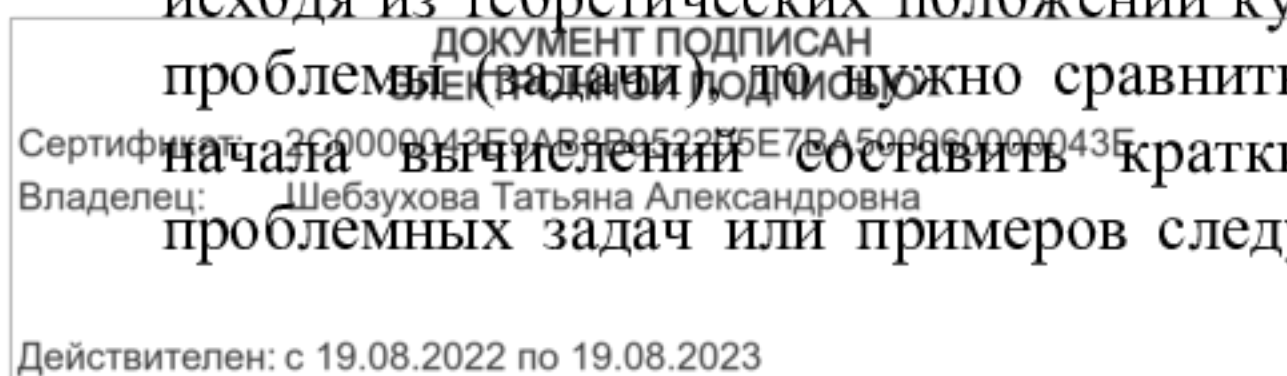
В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля.

Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

4.2. Методические рекомендации по подготовке к практическим занятиям

Для того чтобы практические занятия приносили максимальную пользу, необходимо помнить, что упражнение и решение задач проводятся по вычитанному на лекциях материалу и связаны, как правило, с детальным разбором отдельных вопросов лекционного курса. Следует подчеркнуть, что только после усвоения лекционного материала с определенной точки зрения (а именно с той, с которой он излагается на лекциях) он будет закрепляться на лабораторных занятиях как в результате обсуждения и анализа лекционного материала, так и с помощью решения проблемных ситуаций, задач. При этих условиях студент не только хорошо усвоит материал, но и научится применять его на практике, а также получит дополнительный стимул (и это очень важно) для активной проработки лекции.

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в



строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками.

Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

4.3. Методические рекомендации по самопроверке знаний

После изучения определенной темы по записям в конспекте и учебнику, а также решения достаточного количества соответствующих задач на практических занятиях и самостоятельно студенту рекомендуется провести самопроверку усвоенных знаний, ответив на контрольные вопросы по изученной теме.

В случае необходимости нужно еще раз внимательно разобраться в материале.

Иногда недостаточность усвоения того или иного вопроса выясняется только при изучении дальнейшего материала. В этом случае надо вернуться назад и повторить плохо усвоенный материал. Важный критерий усвоения теоретического материала – умение отвечать на вопросы для собеседования.

Вопросы для собеседования

Знать

1. Основные стили и парадигмы программирования. Парадигма ООП. Понятия объекта и класса.
2. Инкапсуляция. Наследование. Полиморфизм.
3. Microsoft Visual Studio.NET: ее компоненты, версии, редакции и возможности.
4. Идентификаторы и литералы C#. Типы данных. Преобразование типов.
5. Арифметические и логические операторы C#.
6. Декларативные инструкции C#.
7. Инструкции выбора (if, switch).
8. Инструкции повторения (for, while, do-while, foreach).
9. Инструкции перехода (break, continue, goto, return).
10. Способы объявления и инициализации одномерных и двумерных массивов.
11. Свойства и методы класса Array.
12. Объявление класса. Модификаторы доступа.
13. Создание методов классов. Типы возвращаемых методом значений.
14. Конструкторы классов.
15. Задание свойств класса.
16. Наследование классов.
17. Полиморфизм методов. Переопределение и перегрузка методов.
18. Назначение и синтаксис методов класса System.IO.File: OpenRead, OpenWrite, ReadAllText, ReadAllLines, WriteAllText, WriteAllLines, Copy, Move, Delete, Exists.
19. Методы класса String: Length, ToUpper, ToLower, Substring, IndexOf, Remove, Insert, Trim и др.
20. Основные кодировки символов и их особенности.
21. Создание экземпляра DateTime и DateTimeOffset.
22. Получение текущих даты и времени. Манипулирование датой и временем.
23. Форматирование даты и времени. Преобразование даты в строку.
24. Стандартные форматные строки для даты, чувствительные к культуре.
25. Стандартные форматные строки для даты, не чувствительные к культуре.
26. Специальные форматные строки для даты и времени.

27. Преобразование строки в дату/время.

- | | | |
|---------|-----|--|
| Уметь | 28. | Настройка среды разработки Microsoft Visual Studio.NET. |
| Владеть | 29. | Создание, открытие и запуск программ в Microsoft Visual Studio.NET. |
| | 30. | Создание исполняемых файлов. |
| | 31. | Использование методики обработки строк для шифрования текстовых файлов. |
| | 32. | Определение в программах различных типов ошибок. |
| | 33. | Использование инструментов отладки Visual Studio .NET для установки точек останова и исправления ошибок. |
| | 34. | Использование окна Watch для проверки значений переменных во время выполнения программы. Использование окна Command (Окно команд) для изменения значений переменных и исполнения команд в Visual Studio. |
| | 35. | Исправление ошибок времени исполнения с помощью обработчика ошибок Try...Catch. |
| | 36. | Проверка конкретных условий возникновения ошибок с помощью оператора Catch When. |
| | 37. | Использование свойств Err.Number и Err.Description для определения исключений. |

4.4. Методические рекомендации по написанию научных текстов (докладов, рефератов, эссе, научных статей и т.д.)

Перед тем, как приступить к написанию научного текста, важно разобраться, какова истинная цель вашего научного текста - это поможет вам разумно распределить свои силы и время.

Во-первых, сначала нужно определиться с идеей научного текста, а для этого необходимо научиться либо относиться к разным явлениям и фактам несколько критически (своя идея – как иная точка зрения), либо научиться увлекаться какими-то известными идеями, которые нуждаются в доработке (идея – как оптимистическая позиция и направленность на дальнейшее совершенствование уже известного). Во-вторых, научиться организовывать свое время.

Писать следует ясно и понятно, стараясь основные положения формулировать четко и недвусмысленно (чтобы и самому понятно было), а также стремясь структурировать свой текст.

Систематизация и анализ изученной литературы по проблеме исследования позволяют студенту написать работу.

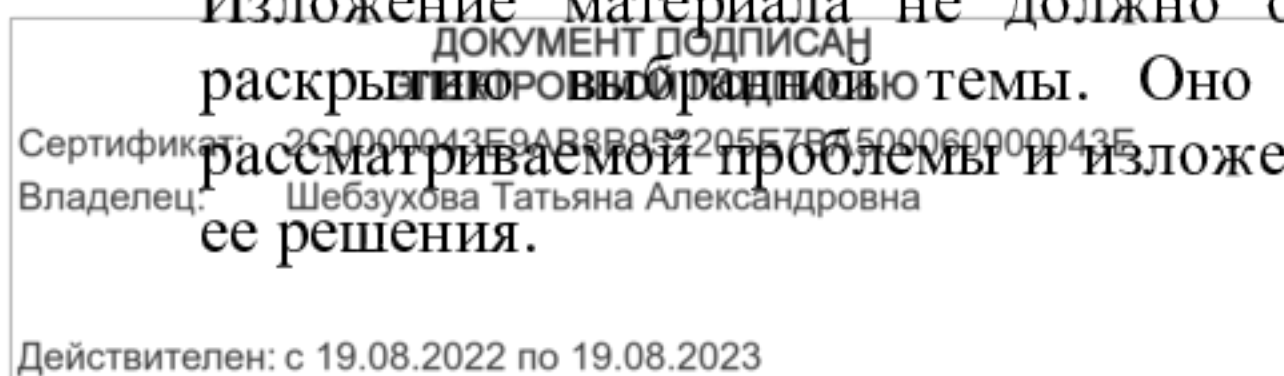
Рабочий вариант текста доклада предоставляется руководителю на проверку. На основе рабочего варианта текста руководитель вместе со студентом обсуждает возможности доработки текста, его оформление.

Структура доклада:

- Введение (не более 3-4 страниц). Во введении необходимо обосновать выбор темы, ее актуальность, очертить область исследования, объект исследования, основные цели и задачи исследования.

- Основная часть состоит из 2-3 разделов. В них раскрывается суть исследуемой проблемы, проводится обзор мировой литературы и источников Интернет по предмету исследования, в котором дается характеристика степени разработанности проблемы и авторская аналитическая оценка основных теоретических подходов к ее решению.

Изложение материала не должно ограничиваться лишь описательным подходом к раскрытию выбранной темы. Оно также должно содержать собственное видение рассматриваемой проблемы и изложение собственной точки зрения на возможные пути ее решения.



- Заключение (1-2 страницы). В заключении кратко излагаются достигнутые при изучении проблемы цели, перспективы развития исследуемого вопроса
- Список использованной литературы (не меньше 10 источников), в алфавитном порядке, оформленный в соответствии с принятыми правилами. В список использованной литературы рекомендуется включать работы отечественных и зарубежных авторов, в том числе статьи, опубликованные в научных журналах в течение последних 3-х лет и ссылки на ресурсы сети Интернет.
- Приложение (при необходимости).

Требования к оформлению:

- текст с одной стороны листа;
- шрифт Times New Roman;
- кегль шрифта 14;
- межстрочное расстояние 1,5;
- поля: сверху 2,5 см, снизу – 2,5 см, слева - 3 см, справа 1,5 см;
- реферат должен быть представлен в сброшюрованном виде.

Порядок защиты доклада:

На защиту доклада отводится 5-7 минут времени, в ходе которого студент должен показать свободное владение материалом по заявленной теме. При защите доклада приветствуется использование мультимедиа-презентации.

Доклад оценивается по следующим критериям: соблюдение требований к его оформлению; необходимость и достаточность для раскрытия темы приведенной в тексте доклада информации; умение студента свободно излагать основные идеи, отраженные в докладе; способность студента понять суть задаваемых преподавателем и сокурсниками вопросов и сформулировать точные ответы на них.

Критерии оценки:

Оценка «отлично» выставляется студенту, если в докладе студент исчерпывающе, последовательно, четко и логически стройно излагает материал; свободно справляется с задачами, вопросами и другими видами применения знаний; использует для написания доклада современные научные материалы; анализирует полученную информацию; проявляет самостоятельность при написании доклада.

Оценка «хорошо» выставляется студенту, если качество выполнения доклада достаточно высокое. Студент твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопросы по теме доклада.

Оценка «удовлетворительно» выставляется студенту, если материал доклада излагается частично, но пробелы не носят существенного характера, студент допускает неточности и ошибки при защите доклада, дает недостаточно правильные формулировки, наблюдаются нарушения логической последовательности в изложении материала.

Оценка «неудовлетворительно» выставляется студенту, если он не подготовил доклад или допустил существенные ошибки. Студент неуверенно излагает материал доклада, не отвечает на вопросы преподавателя.

Описание шкалы оценивания

Максимально возможный балл за весь текущий контроль устанавливается равным 55. Текущее контрольное мероприятие считается сданным, если студент получил за него не менее 60% от установленного для этого контроля максимального балла. Рейтинговый балл, выставляемый студенту за текущее контрольное мероприятие, сданное студентом в установленные графиком контрольных мероприятий сроки, определяется следующим образом:

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ Сертификат: 2C0000043E9AB8B952205E7BA500060000043E Владелец: Шебзухова Татьяна Александровна	
Уровень выполнения контрольного задания	Рейтинговый балл (в % от максимального балла за контрольное задание)
Действителен: с 19.08.2022 по 19.08.2023	

Отличный	100
Хороший	80
Удовлетворительный	60
Неудовлетворительный	0

Темы эссе (рефератов, докладов, сообщений)

Базовый уровень

1. Идентификаторы и литералы C#. Типы данных. Преобразование типов.
2. Арифметические и логические операторы C#.
3. Декларативные инструкции C#.
4. Инструкции выбора (if, switch).
5. Инструкции повторения (for, while, do-while, foreach).
6. Инструкции перехода (break, continue, goto, return).
7. Способы объявления и инициализации одномерных и двумерных массивов.
8. Свойства и методы класса Array.
9. Объявление класса. Модификаторы доступа.
10. Создание методов классов. Типы возвращаемых методом значений.
11. Конструкторы классов.
12. Задание свойств класса.
13. Наследование классов.

Повышенный уровень

1. Основные кодировки символов и их особенности.
2. Создание экземпляра DateTime и DateTimeOffset.
3. Получение текущих даты и времени. Манипулирование датой и временем.
4. Форматирование даты и времени. Преобразование даты в строку.
5. Стандартные форматные строки для даты, чувствительные к культуре.
6. Стандартные форматные строки для даты, не чувствительные к культуре.
7. Специальные форматные строки для даты и времени.
8. Преобразование строки в дату/время.

4.5. Методические рекомендации по подготовке к зачетам

Процедура зачета как отдельное контрольное мероприятие не проводится, оценивание знаний обучающегося происходит по результатам текущего контроля.

Зачет выставляется по результатам работы в семестре, при сдаче всех контрольных точек, предусмотренных текущим контролем успеваемости. Если по итогам семестра обучающийся имеет от 33 до 60 баллов, ему ставится отметка «зачтено». Обучающемуся, имеющему по итогам семестра менее 33 баллов, ставится отметка «не зачтено».

Количество баллов за зачет (Sзач) при различных рейтинговых баллах по дисциплине по результатам работы в семестре

Рейтинговый балл по дисциплине по результатам работы в семестре ($R_{сем}$)	Количество баллов за зачет ($S_{зач}$)
$50 \leq R_{сем} \leq 60$	40
$39 \leq R_{сем} < 50$	35
$33 \leq R_{сем} < 39$	27
$R_{сем} < 33$	0

Сертификат: 2C0000043E9AB8B3522851FA50008000043E
Владелец: Шебухова Татьяна Александровна

Действителен: с 19.08.2022 по 19.08.2023

Контроль самостоятельной работы студентов

Контроль самостоятельной работы проводится преподавателем в аудитории.

Предусмотрены следующие виды контроля: собеседование, оценка выполнения доклада и его презентации.

Подробные критерии оценивания компетенций приведены в Фонде оценочных средств для проведения текущей и промежуточной аттестации.

Список литературы для выполнения СРС

Основная литература:

1. Мейер Б. Объектно-ориентированное программирование и программная инженерия [Электронный ресурс] / Б. Мейер. — 3-е изд. — Электрон. текстовые данные. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019. — 285 с. — 978-5-4486-0513-0. — Режим доступа: <http://www.iprbookshop.ru/79706.html>.

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. Бабушкина И.А. Практикум по объектно-ориентированному программированию [Электронный ресурс]/ Бабушкина И.А., Окулов С.М.— Электрон. текстовые данные.— М.: БИНОМ. Лаборатория знаний, 2015.— 367 с. — Режим доступа: <http://www.iprbookshop.ru/12254>.— ЭБС «IPRbooks».

Методическая литература:

1. методические указания к лабораторным работам;
2. методические указания к самостоятельной работе.

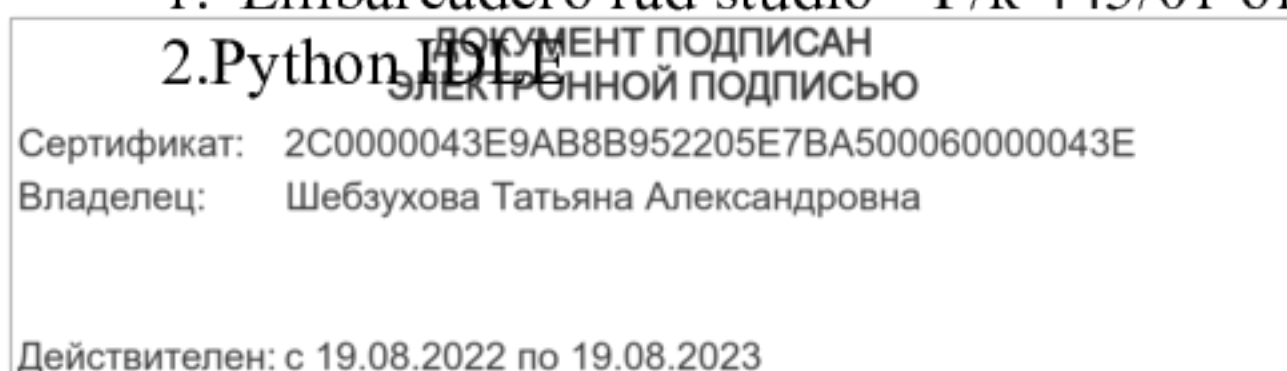
Интернет-ресурсы:

1. <http://www.intuit.ru> – сайт дистанционного образования в области информационных технологий
2. <http://e.lanbook.com> – ЭБС издательства «Лань».
3. <http://www.biblioclub.ru> – университетская библиотека онлайн.
4. <http://window.edu.ru> – образовательные ресурсы ведущих вузов
5. <http://ncfu.ru> – сайт СКФУ

Программное обеспечение:

1. Embarcadero rad studio - Г/к 445/01 от 30 июля 2010 г.,

2. Python, IDE



**ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ**

Сертификат: 2C0000043E9AB8B952205E7BA500060000043E

Владелец: Шебзухова Татьяна Александровна

Действителен: с 19.08.2022 по 19.08.2023