

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
"Амурский государственный университет"

УТВЕРЖДАЮ

Проректор по учебной и научной
работе

Лейфа А.В. Лейфа

18 июня 2024 г.

РАБОЧАЯ ПРОГРАММА
«ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ АНАЛИЗ, ПРОГРАММИРОВАНИЕ»

Направление подготовки 09.03.04 Программная инженерия

Направленность (профиль) образовательной программы – Программная инженерия

Квалификация выпускника – Бакалавр

Год набора – 2024

Форма обучения – Очная

Курс 1,2 Семестр 2,3

Экзамен 3 сем

Зачет 2 сем

Общая трудоемкость дисциплины 288.0 (академ. час), 8.00 (з.е)

Составитель Т.А. Галаган, доцент, канд. техн. наук

Институт компьютерных и инженерных наук

Кафедра информационных и управляющих систем

Рабочая программа составлена на основании Федерального государственного образовательного стандарта ВО для направления подготовки 09.03.04 Программная инженерия, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.17 № 920

Рабочая программа обсуждена на заседании кафедры информационных и управляющих систем

17.04.2024 г. , протокол № 8

Заведующий кафедрой Бушманов А.В. Бушманов

СОГЛАСОВАНО

Учебно-методическое управление

Чалкина Н.А. Чалкина

18 июня 2024 г.

СОГЛАСОВАНО

Научная библиотека

Петрович О.В. Петрович

18 июня 2024 г.

СОГЛАСОВАНО

Выпускающая кафедра

Бушманов А.В. Бушманов

18 июня 2024 г.

СОГЛАСОВАНО

Центр цифровой трансформации и
технического обеспечения

Тодосейчук А.А. Тодосейчук

18 июня 2024 г.

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель дисциплины:

изучение принципов объектно-ориентированного программирования, возможностей современных инструментальных средств разработки программного обеспечения; привитие навыков использования методов исследования и инструментальных средств разработки программного обеспечения на основе объектно-ориентированного подхода.

Задачи дисциплины:

Задачи дисциплины:

изучение основных принципов объектно-ориентированного программирования, изучение возможностей языка программирования C#, его стандартных библиотек классов и шаблонов;

изучение возможностей инструментальных средств разработки программного обеспечения: Dev C++, Microsoft Visual Studio, платформы пользовательского интерфейса Windows Forms;

привитие умений использования методов и инструментальных средств Microsoft Visual Studio для исследования программного обеспечения.

привитие устойчивых практических навыков применения современных инструментальных средств при разработке, отладке программного продукта, проектирования и создания программных интерфейсов.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

В учебном плане дисциплина «Объектно-ориентированное программирование» содержится в перечне дисциплин части, формируемой участниками образовательных отношений.

Для изучения дисциплины «Объектно-ориентированное программирование» студент должен обладать стартовыми навыками создания программ на языке высокого уровня, уметь анализировать и обобщать информацию, желательно обладать аналитическим складом мышления, что могло быть получено в результате изучения дисциплин «Цифровая грамотность», «Программирование» в объеме образовательной программы данного направления.

Изучение дисциплины «Объектно-ориентированное программирование» является основой для изучения дальнейших дисциплин, использующих ЭВМ и программирование, таких как «Базы данных», «Тестирование программного обеспечения», "Проектирование программного обеспечения" и др, а также при выполнении выпускной квалификационной работы.

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ И ИНДИКАТОРЫ ИХ ДОСТИЖЕНИЯ

3.1 Профессиональные компетенции и индикаторы их достижения

Код и наименование профессиональной компетенции	Код и наименование индикатора достижения профессиональной компетенции
ПК-5. Готовность к использованию методов и инструментальных средств исследования объектов профессиональной деятельности, способность к формализации в своей предметной области с учетом ограничений используемых	ИД-1 ПК-5. знать: современные инструментальные средства программного обеспечения; ИД-2 ПК-5. уметь: анализировать и выбирать инструментальные средства программного обеспечения; ИД-3 ПК-5. иметь навык использования методов и инструментальных средств исследования программного обеспечения.

4. СТРУКТУРА ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 8.00 зачетных единицы, 288.0 академических часов.

1 – № п/п

2 – Тема (раздел) дисциплины, курсовая работа (проект), промежуточная аттестация

3 – Семестр

4 – Виды контактной работы и трудоемкость (в академических часах)

4.1 – Л (Лекции)

4.2 – Лекции в виде практической подготовки

4.3 – ПЗ (Практические занятия)

4.4 – Практические занятия в виде практической подготовки

4.5 – ЛР (Лабораторные работы)

4.6 – Лабораторные работы в виде практической подготовки

4.7 – ИКР (Иная контактная работа)

4.8 – КТО (Контроль теоретического обучения)

4.9 – КЭ (Контроль на экзамене)

5 – Контроль (в академических часах)

6 – Самостоятельная работа (в академических часах)

7 – Формы текущего контроля успеваемости

1	2	3	4									5	6	7
			4.1	4.2	4.3	4.4	4.5	4.6	4.7	4.8	4.9			
1	Основные принципы объектно-ориентированного программирования	2	2										13,8	Тестовое задание
2	Особенности построения объектно-ориентированных программ на языке C++	2	6				6						20	Тестовое задание
3	Принцип наследования	2	4				4						20	Тестовое задание
4	Принцип полиморфизма	2	6				8						20	Тестовое задание
5	Событийно-управляемое программирование. Особенности разработки Windows-приложений на языке C#	3	8		2		2						10	Тестовое задание

6	Создание пользовательских классов в языке C#	3	4		2		6					20	Тестовое задание
7	Особенности реализации наследования и полиморфизма в языке C#	3	16		6		16					20	Тестовое задание
8	Структуры данных и шаблоны языка C#	3	4				4						Тестовое задание
9	Паттерны проектирования	3	2		6		4					10.8	Тестовое задание Сообщение
10	Зачет	2							0.2				
11	Экзамен	3								0.3	35.7		
	Итого			52.0		16.0		50.0	0.0	0.2	0.3	35.7	133.8

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

5.1. Лекции

№ п/п	Наименование темы (раздела)	Содержание темы (раздела)
1	Основные принципы объектно-ориентированного программирования	Принципы декомпозиции и абстрагирования. Инкапсуляция, наследования, полиморфизм. UML- диаграммы как средство проектирования объектно- ориентированных программ. Виды отношений между классами
2	Особенности построения объектно-ориентированных программ на языке C++	Элементы класса: поля, методы. Спецификаторы доступа в языке программирования C++.Создание объектов. Правила создания конструкторов и деструкторов. Виды конструкторов. Внутренний указатель this. Статические поля
3	Принцип наследования	Понятие производного класса, синтаксис его описания. Особенности объявления конструкторов производного класса. Иерархия классов. Дружественность
4	Принцип полиморфизма	Перегрузка и замещение методов. Функции-операции. Виртуальные методы. Абстрактные классы.
5	Событийно- управляемое программирование. Особенности разработки Windows- приложений на языке C#	Язык C#: состав, основные операции, структура простейшей программы. Типы данных языка C#. Особенности применения операторов ввода-вывода при организации программных интерфейсов. Событийно- управляемое программирование. Основные компоненты визуального конструктора для создания интерфейса Windows приложений. Понятие "исключительная ситуация". Обработка исключительных ситуаций в языке программирования C#.

6	Создание пользовательских классов в языке C#	Элементы класса: поля, методы, индексы, свойства. Спецификаторы доступа в языке программирования C#. Особенности использования спецификаторов доступа для различных элементов класса. Создание объектов. Правила создания конструкторов и деструкторов.
7	Особенности реализации наследования и полиморфизма в языке C#	Особенности объявления производного класса в C#. Иерархия классов. Замещение методов. Функции-операции. Объявление и переопределение виртуальных методов. Абстрактные методы и классы. Бесплодные классы. Правила использования интерфейсов и коллекций языка C#.
8	Структуры данных и коллекции языка C#	Правила использования коллекций языка C#. Коллекции пространства имен System.Collections.Generic. Обработка списков. Стеки.
9	Паттерны проектирования	Понятие «паттерн проектирования». История появления и использования паттерн проектирования. Основные паттерны проектирования программ. Делегаты. Методы делегирования. Принципы SOLID

5.2. Практические занятия

Наименование темы	Содержание темы
Особенности применения компонент Windows Forms	Элементы Windows Forms. Свойства и события основных элементов. Создание прямоугольных форм. Переход между формами
Объявление и реализация классов	Поля, методы, свойства классов. Использование внутреннего указателя. Реализация классов в программе
Реализация принципа наследования	Базовый и производный классы. Правила доступа к закрытым элементам. Особенности объявления конструкторов в производных классах. Особенности реализации классов-потомков
Перегрузка операций	Особенности перегрузки унарных и бинарных операций.
Реализация принципа полиморфизма	Виртуальные и абстрактные методы, их переопределение. Объявление абстрактных классов. Интерфейсы, множественный полиморфизм. Стандартные интерфейсы библиотеки .NET
Порождающие паттерны	Фабричный метод (Factory Method). Абстрактная фабрика (Abstract Factory). Одиночка (Singleton). Прототип (Prototype). Строитель (Builder)
Паттерны поведения	Стратегия (Strategy). Наблюдатель (Observer). Команда (Command). Шаблонный метод (Template)

	Method). Итератор (Iterator). Состояние (State). Цепочка Обязанностей (Chain of responsibility). Интерпретатор (Interpreter). Посредник (Mediator). Хранитель (Memento). Посетитель (Visitor)
Структурные паттерны	Декоратор (Decorator). Адаптер (Adapter). Фасад (Facade). Компоновщик (Composite). Заместитель (Прокси). Мост (Bridge). Приспособленец (Flyweight)

5.3. Лабораторные занятия

Наименование темы	Содержание темы
Классы	Описание элементов класса. Распределение прав доступа. Реализация классов
Конструкторы и деструкторы	Создание конструкторов различных видов. Использование деструктора в программе
Реализация наследования в языке C++	Создание производных классов, их реализация в консольных приложениях.
Перегрузка операций	Описание и применение унарных и бинарных функций-операций
Виртуальные методы	Определение и переопределение виртуальных функций. Замещение методов
Абстрактные классы	Создание абстрактных классов и чисто виртуальных функций. Создание классов на основе абстрактного предка. Обработка списков объектов.
Создание Windows-приложений	Создание проекта в среде Visual Studio на языке C#. Использование компонент Windows Forms. Обработка исключительных ситуаций при организации пользовательских интерфейсов
Классы	Создание элементов класса на языке C#. Реализация классов Использование свойств
Конструкторы	Применение различных видов конструкторов в реализации класса
Наследование	Создание производных классов. Реализация много файловых программ. Перегрузка операций
Виртуальные методы, абстрактные методы и абстрактные классы	Определение и переопределение виртуальных функций.
Интерфейсы	Множественный полиморфизм. Стандартные интерфейсы и коллекции библиотеки .NET
Делегирование	Создание делегатов. Реализация делегирования. Лямбда-выражения
Применение шаблонов проектирования	Применение паттернов проектирования в графических программах на языке C#

6. САМОСТОЯТЕЛЬНАЯ РАБОТА

№ п/п	Наименование темы (раздела)	Содержание темы (раздела)	Трудоемкость в академических часах
1	Основные принципы объектно-ориентированного программирования	Изучение учебной литературы Приобретение практических навыков работы в среде Dev C++ Подготовка к тестированию	13,8
2	Особенности построения объектно-ориентированных программ на языке C++	Изучение учебной литературы Приобретение практических навыков работы в среде Dev C++ Подготовка к тестированию	20
3	Принцип наследования	Изучение учебной литературы Приобретение практических навыков работы в среде Dev C++ Подготовка к тестированию	20
4	Принцип полиморфизма	Изучение учебной литературы Приобретение практических навыков работы в среде Dev C++ Подготовка к тестированию	20
5	Событийно-управляемое программирование. Особенности разработки Windows-приложений на языке C#	Изучение учебной литературы Приобретение практических навыков работы в среде программирования Visual Studio.NET на языке C# Подготовка к тестированию	10
6	Создание пользовательских классов в языке C#	Изучение учебной литературы Подготовка к тестированию Развитие практических навыков работы в среде программирования Visual Studio.NET на языке C#	20
7	Особенности реализации наследования и полиморфизма в языке C#	Изучение учебной литературы Подготовка к тестированию Развитие практических навыков работы в среде программирования Visual Studio.NET на языке C#	20
8	Паттерны проектирования	Изучение учебной литературы Подготовка сообщения	10.8

7. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

К образовательным технологиям, используемым в преподавании данной дисциплины, относятся лекции, лабораторные занятия, курсовое проектирование..

В изложении лекционного материала наряду с традиционной лекцией используются такие неимитационные методы обучения, как:

проблемная лекция, начинающаяся с постановки проблемы, которую необходимо решить в ходе изложения материала,

лекция- визуализация, учащая студента преобразовывать устную и письменную информацию к визуальной форме в виде схем, рисунков, чертежей,

лекция с заранее запланированными ошибками, которые студенты должны обнаружить самостоятельно в конце лекции.

На лекциях используются информационные технологии – презентации. Лабораторные занятия проводятся в компьютерных классах и предназначены для решения прикладных задач с использованием современных инструментальных средств.

При проведении лабораторных работ используются неигровые имитационные методы обучения:

контекстное обучение, направленное на решение профессиональных задач,

работа в команде – совместная деятельность студентов в группе, направленная на решение общей задачи с разделением ответственности и полномочий.

При оценивании результатов обучения используется балльно-рейтинговая технология.

8. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Описание показателей и критериев оценивания компетенций на различных этапах их формирования; описание шкал оценивания; типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков, характеризующих этапы формирования компетенций; а так же методические материалы, определяющие процедуры оценивания знаний, умений и навыков, отражены в фонде оценочных средств (ФОС).

Для оценки текущей успеваемости в данной дисциплине относятся: тестовые задания с закрытыми и открытыми видами вопросов; отчеты по выполнению лабораторных работ; зачет, экзамен.

Вопросы к зачету

1. Принципы декомпозиции и абстрагирования
2. Инкапсуляция, наследования, полиморфизм
3. UML- диаграммы как средство проектирования объектно- ориентированных программ
4. Элементы класса: поля, методы
5. Спецификаторы доступа в языке программирования C++
6. Создание объектов
7. Правила создания конструкторов и деструкторов
8. Виды конструкторов
9. Внутренний указатель this
10. Статические поля
11. Понятие производного класса, синтаксис его описания
12. Особенности объявления конструкторов производного класса
13. Множественное наследование
14. Дружественность
15. Перегрузка операций
16. Замещение методов в производном классе
17. Функции-операции (объявление, использование)
18. Виртуальные методы (определение и переопределение).
19. Чисто виртуальные методы
20. Абстрактные классы.

Вопросы к экзамену

1. Ключевые принципы объектно-ориентированного анализа и проектирования
2. UML- диаграммы как средство моделирования объектно- ориентированных программ
3. Диаграмма классов. Особенности изображения элементов класса на диаграмме

4. Виды отношений между классами: ассоциация, агрегация, наследование, зависимость. Их обозначение на UML-диаграммы
5. Язык С#: состав, основные операции
6. Библиотека математических функций
7. Операторы ввода-вывода в С#
8. Структура простейшей программы на языке С#
9. Операторы ветвления
10. Операторы цикла
11. Массивы
12. Определение класса (поля, методы, свойства, индексы, объекты)
13. Спецификаторы доступа, правила их использования
14. Конструкторы и деструкторы
15. Иерархия классов. Реализация принципа наследования в С#
16. Виртуальные функции. Принципы их переопределения
17. Абстрактные функции и классы
18. Функции-операции
19. Общий механизм обработки исключений
20. Событийно-управляемое программирование
21. Основные компоненты конструктора формы.
22. Правила использования интерфейсов
23. Стандартные интерфейсы .NET: IComparable, IComparer
24. Перечисляемый тип
25. Структурные типы данных
26. Пространство имен System.Collection
27. Коллекций языка С#
28. Делегаты
29. Принципы делегирования
30. Паттерны объектно-ориентированного проектирования приложений
31. Паттерны «строитель», «наблюдатель»
32. Принципы SOLID

9. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

а) литература

1. Зыков, С. В. Программирование: учебник и практикум для академического бакалавриата / С. В. Зыков. — 2-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2023. — 285 с. — (Бакалавр. Академический курс). — ISBN 978-5-534-16031-4. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/530294> (дата обращения: 02.04.2024).
2. Подбельский, В. В. Программирование. Базовый курс С#: учебник для вузов / В. В. Подбельский. — Москва: Издательство Юрайт, 2024. — 369 с. — (Высшее образование). — ISBN 978-5-534-10616-9. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/536775> (дата обращения: 02.04.2024).
3. Тузовский, А. Ф. Объектно-ориентированное программирование: учебное пособие для вузов / А. Ф. Тузовский. — Москва: Издательство Юрайт, 2024. — 213 с. — (Высшее образование). — ISBN 978-5-534-16316-2. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/537332> (дата обращения: 02.04.2024).
4. Огнева, М. В. Программирование на языке С++: практический курс: учебное пособие для вузов / М. В. Огнева, Е. В. Кудрина. — Москва: Издательство Юрайт, 2024. — 335 с. — (Высшее образование). — ISBN 978-5-534-05123-0. — Текст:

электронный // Образовательная платформа Юрайт [сайт]. — URL: [https:// urait.ru/bcode/539713](https://urait.ru/bcode/539713) (дата обращения: 02.04.2024).

5. Галаган, Т.А. Объектно-ориентированное программирование. Язык C++. Учебное пособие/ Т.А. Галаган – Благовещенск: изд-во Амур. гос. ун-та, 2016. – 56 с. [http:// irbis.amursu.ru/DigitalLibrary/AmurSU_Edition/9699.pdf](http://irbis.amursu.ru/DigitalLibrary/AmurSU_Edition/9699.pdf)

б) программное обеспечение и Интернет-ресурсы

№	Наименование	Описание
1	DevC++	Бесплатное распространение по стандартной общественной лицензии GNU AGPL http://www.gnu.org/licenses/agpl-3.0.html .
2	MS Visual Studio Community 2022	Лицензия на программное обеспечение Microsoft https://visualstudio.microsoft.com/ru/license-terms/vs2022-ga-community/
3	https://metanit.com/shar/	Сайт посвящен различным языкам и технологиям программирования, компьютерам, мобильным платформам и ИТ- технологиям. Данный раздел посвящен языку программирования C#.
4	http://www.window.edu.ru	Единое окно доступа к образовательным ресурсам/ каталог/ профессиональное образование
5	https://urait.ru/	Образовательная платформа Юрайт [сайт] Электронная библиотечная система «Юрайт»Фонд электронной библиотеки составляет более 4000 наименований и постоянно пополняется новинками, в большинстве своем это учебники и учебные пособия для всех уровней профессионального образования от ведущих научных школ с соблюдением требований новых ФГОСов.
6	http://www.iprbookshop.ru/	ЭБС IPRbooks объединяет новейшие информационные технологии и учебную лицензионную литературу. Контент IPRbooks отвечает требованиям стандартов высшей школы, дополнительного и дистанционного образования
7	https://metanit.com/cpp	Сайт посвящен различным языкам и технологиям программирования, компьютерам, мобильным платформам и ИТ- технологиям. Данный раздел посвящен языку программирования C++.

в) профессиональные базы данных и информационные справочные системы

№	Наименование	Описание
1	http:// www.ict.edu.ru/about	Портал "Информационно- коммуникационные технологии в образовании" входит в систему федеральных образовательных порталов и нацелен на обеспечение комплексной информационной поддержки образования в области современных информационных и телекоммуникационных технологий, а также деятельности по применению ИКТ в сфере образования
2	https://reestr.minsvyaz.ru	Единый реестр российских программ для электронных вычислительных машин и баз данных. Реестр создан в соответствии со статьей 12.1 Федерального закона «Об информации, информационных технологиях и о защите информации» в целях расширения использования рос-

		сийских программ для электронных вычислительных машин и баз данных, подтверждения их происхождения из Российской Федерации, а также в целях оказания правообладателям программ для электронных вычислительных машин или баз данных мер государственной поддержки
3	http://www.informika.ru	Сайт ФГАУ ГНИИ ИТТ «Информика». Институт является государственным научным предприятием, созданным для обеспечения всестороннего развития и продвижения новых информационных технологий в сферах образования и науки России. Институт создан для осуществления комплексной поддержки развития и использования новых информационных технологий и телекоммуникаций в сфере образования и науки России
4	www.elibrary.ru	Крупнейший российский информационный портал в области науки, технологии, медицины и образования
5	https://www.scopus.com	Международная реферативная база данных научных изданий Scopus
6	https://login.webofknowledge.com	Международная реферативная база данных научных изданий Web of Science

10. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

В качестве основных технических средств обучения по дисциплине «Объектно-ориентированное программирование» используются:

- мультимедийные лекционные аудитории, оснащенные проектором, обеспечивающим воспроизводство слайдов и текстов с экрана монитора компьютер лектора, управляющим компьютером, устройствами затемнения, обеспечения информационной безопасности и поддержания микроклимата;

- классы кафедры информационных и управляющих систем АмГУ, оборудованные компьютерами, подключенные к ЛВС университета с возможностью подключения сети Интернет и обеспечением доступа в электронную информационно-образовательную среду университета.

В качестве программного обеспечения используются средства, указанные в п.9 данного документа.

Самостоятельная работа обучающихся осуществляется в помещениях для самостоятельной работы, оснащенных компьютерной техникой с возможностью подключения сети Интернет и обеспечением доступа в электронную информационно-образовательную среду университета.