

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ

Заведующий кафедрой ПиИТ



проф. Махортов С.Д.,

подпись, расшифровка подписи

05.03.2024

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Б1.О.35 Объектно-ориентированное программирование

1. Шифр и наименование направления подготовки / специальности: 10.05.01

Компьютерная безопасность

2. Профиль подготовки: Анализ безопасности компьютерных систем

3. Квалификация выпускника: специалист

4. Форма обучения: очная

5. Кафедра, отвечающая за реализацию дисциплины: ПиИТ

6. Составители программы: Соломатин Дмитрий Иванович, ассистент
(*ФИО, ученая степень, ученое звание*)

7. Рекомендована: Научно-методическим советом ФКН, протокол № 5 от 05.03.2024
(*наименование recommending structure, date, protocol number*)

8. Учебный год: 2025/2026

Семестр(ы): 3

9. Цели и задачи учебной дисциплины: Изучение современных объектно-ориентированных подходов и технологий в разработки ПО (обобщенное программирование, паттерны проектирования, компонентная разработка); углубленное изучение языка C# и знакомство с библиотекой .NET FCL; формирование практических навыков объектно-ориентированного программирования и проектирования ПО.

10. Место учебной дисциплины в структуре ООП:

Дисциплины (модули) / Вариативная часть / Обязательные дисциплины.

Требования к входным знаниям, умениям и компетенциям: Практические навыки программирования на языках высокого уровня.

11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников):

Код и название компетенции	Код и название индикатора компетенции	Знания, умения, навыки
ОПК-2. Способен применять программные средства системного и прикладного назначения, в том числе отечественного производства, для решения задач профессиональной деятельности.	ОПК-2.9. Умеет разрабатывать системное и прикладное программное обеспечение для многозадачных, многопользовательских и многопроцессорных сред, а также для сред с интерфейсом, управляемым сообщениями.	Уметь: разрабатывать программы сложных информационных систем с использованием технологии объектно-ориентированного программирования.
	ОПК-2.10. Умеет применять основные методы программирования в выбранной операционной среде.	Уметь: применять объектно-ориентированное программирование для разработки системных программ.
ОПК-7. Способен создавать программы на языках высокого и низкого уровня, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ.	ОПК-7.1. Знает общие принципы построения, области и особенности применения языков программирования высокого уровня.	Знать: общие принципы объектно-ориентированного программирования.
	ОПК-7.2. Знает язык программирования высокого уровня (объектно-ориентированное программирование).	Знать: объектно-ориентированный язык программирования C#.
	ОПК-7.3. Знает язык ассемблера персонального компьютера.	Знать: принципы представления данных в объектно-ориентированном языке программирования.
	ОПК-7.4. Умеет работать с интегрированной средой разработки программного обеспечения.	Уметь: работать со средой программирования языка программирования C#.
	ОПК-7.5. Умеет разрабатывать и реализовывать на языке высокого уровня алгоритмы решения типовых профессиональных задач.	Уметь: разрабатывать и реализовывать решения задач на объектно-ориентированном языке.
	ОПК-7.6. Владеет навыками разработки, документирования, тестирования и отладки программ.	Владеть: навыками разработки, документирования, тестирования и отладки программ.

	ОПК-7.7. Знает базовые структуры данных.	Знать: базовые структуры данных объектно-ориентированного языка.
	ОПК-7.8. Знает основные алгоритмы сортировки и поиска данных, комбинаторные и теоретико-графовые алгоритмы.	Знать: основные алгоритмы сортировки и поиска данных, комбинаторные и теоретико-графовые алгоритмы.
	ОПК-7.9. Знает общие сведения о методах проектирования, документирования, разработки, тестирования и отладки программного обеспечения.	Знать: общие сведения о методах проектирования объектно-ориентированных программ, документирования с использованием комментариев и мета-данных, разработки, тестирования и отладки программ.
	ОПК-7.10. Умеет применять известные методы программирования и возможности базового языка программирования для решения типовых профессиональных задач.	Уметь: применять современные методы объектно-ориентированного программирования и возможности языка программирования для решения задач.
	ОПК-7.11. Владеет навыками разработки алгоритмов решения типовых профессиональных задач.	Владеть: навыками разработки алгоритмов для разработки и реализации программ.
ОПК-13. Способен разрабатывать компоненты программных и программно-аппаратных средств защиты информации в компьютерных системах и проводить анализ их безопасности.	ОПК-13.4. Знает язык программирования высокого уровня (объектно-ориентированное программирование).	Знать: объектно-ориентированный язык программирования.
	ОПК-13.7. Владеет навыками использования инструментальных средств отладки и дизассемблирования программного кода.	Владеть: навыками использования инструментальных средств отладки и дизассемблирования программного кода.

12. Объем дисциплины в зачетных единицах/часах — 3/108.

Форма промежуточной аттестации: экзамен

13. Виды учебной работы

Вид учебной работы	Трудоемкость (часы)			
	Всего	По семестрам		
		3 сем.	№ сем.	
Аудиторные занятия	54	54		
в том числе:				
лекции	16	16		
практические	-	-		
лабораторные	34	34		
Самостоятельная работа	22	22		
Контроль	36	36		
Всего:	108	108		

13.1. Содержание разделов дисциплины

№ п/п	Наименование раздела дисциплины	Содержание раздела дисциплины
1	Введение в ООП	Инкапсуляция, наследование, полиморфизм. Классы в C#. Члены классов: константы, поля, readonly-поля, свойства, индексы, методы, конструкторы, "деструкторы" (финализаторы). Статические и динамические члены классов.
2	Обзор .NET Framework и библиотеки классов .NET FCL	Состав .NET Framework. Общезыковая среда исполнения, MSIL-код. Поддерживаемые типы данных: базовые типы, классы, структуры, перечисления, интерфейсы, строки, массивы (многомерные и ступенчатые). Библиотека классов .NET FCL. Понятие потока данных, класс System.IO.Stream. Ввод-вывод текстовых данных, пример чтения/записи текстовых файлов. Понятие сериализации.
3	Делегаты и события	Создание делегатов на основе именованных и неименованных методов. Функциональный стиль программирования на основе делегатов. Методы массивов и коллекций с параметрами-делегатами. Реализация своих событий.
4	Обобщенное программирование	Обобщенное программирование в C# на основе generics. Пример описания и использования обобщенного класса (например, стека). Ограничения в generics. Системные интерфейсы: ICloneable, IComparable<T>/IComparable, IDisposable, IEquatable<T>, ISerializable, IEnumerable<T>/IEnumerable, ICollection<T>/ICollection.
5	Понятие качества кода	Внешние и внутренние факторы качества ПО, связь факторов. Обзор типичных ошибок в архитектуре приложений. Классические примеры «плохого кода» в проектах студентов.
6	Графическая нотация ОМТ	Графическая нотация ОМТ: Диаграммы классов (инстанцирование, наследование, агрегирование, осведомленность), диаграммы объектов, диаграммы взаимодействия. Разбор примеров кода и соответствующих ему диаграмм.
7	Понятие паттернов проектирования	Механизмы повторного использования кода. Понятие паттернов проектирования ПО. Элементы паттернов: имя, задача, решение, результаты. Классификация GoF-паттернов.
8	Порождающие паттерны проектирования	Обзор порождающих паттернов проектирования: Фабричный метод (Factory Method), Абстрактная фабрика (Abstract Factory), Прототип (Prototype), Строитель (Builder), Одиночка (Singleton). Разбор примеров использования соответствующих паттернов.
9	Структурные паттерны проектирования	Обзор структурных паттернов: Адаптер (Adapter), Декоратор (Decorator), Заместитель (Proxy), Компонщик (Composite), Мост (Bridge), Приспособленец (Flyweight), Фасад (Facade). Разбор примеров использования соответствующих паттернов. Разбор примеров использования соответствующих паттернов.
10	Поведенческие паттерны проектирования	Обзор паттернов поведения: Интерпретатор (Interpreter), Итератор (Iterator), Команда (Command), Наблюдатель (Observer), Посетитель (Visitor), Посредник (Mediator), Состояние (State), Хранитель (Memento), Цепочка

		обязанностей (Chain of Responsibility), Шаблонный метод (Template Method).Разбор примеров использования соответствующих паттернов.
11	Разработка компонентов и элементов управления	Компоненты и элементы управления в .NET Framework.Создание компонентов и элементов управления. Пользовательские и специализированные элементы управления.Описание свойств и событий элементов управления, использование атрибутов.Поддержка времени разработки: конверторы типов, дизайнеры типов, дизайнеры компонентов.Разбор нескольких примеров создания элементов управления различными способами.
12	Реализация пользовательского интерфейса в C# приложениях. Паттерн MVC.	Принцип разделения логики приложения от отображения данных.Паттерн проектирования Model-View-Controller для визуальных приложений.Привязка данных к элементам управления в .NET Framework.Разбор примера MVC-приложения на языке C#.

13.2. Разделы дисциплины и виды занятий

№ п/п	Наименование раздела дисциплины	Виды занятий (часов)				Всего
		Лекции	Практические	Лабораторные	Самостоятельная работа	
1	Введение в ООП	1		1	1	3
2	Обзор .NET Framework и библиотеки классов .NET FCL	1		1	1	3
3	Делегаты и события	1		2	1	4
4	Обобщенное программирование	1		2	3	6
5	Понятие качества кода	1		2	1	4
6	Графическая нотация ОМТ	1		4	1	6
7	Понятие паттернов проектирования	1		4	3	8
8	Порождающие паттерны проектирования	1		2	3	6
9	Структурные паттерны проектирования	2		4	2	8
10	Поведенческие паттерны проектирования	2		4	2	8
11	Разработка компонентов и элементов управления	2		4	2	8
12	Реализация пользовательского интерфейса в C# приложениях. Паттерн MVC.	2		4	2	8
Итого:		16		34	22	72

14. Методические указания для обучающихся по освоению дисциплины

Презентации к курсу лекций, примеры к практическим заданиям (размещены на сетевом диске с общим доступом в компьютерных классах ФКН).

15. Учебная литература, необходимая для освоения дисциплины

а) основная литература:

№ п/п	Источник
1	Фаронов В.В. Программирование на языке С#. – СПб.: Питер, 2007. – 240 с.

б) дополнительная литература:

№ п/п	Источник
2	Джесс Либерти. Программирование на С#, 2-е изд. – СПб.: Символ-плюс, 2002. – 684 с.
3	Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования. – СПб: Питер, 2001. – 368 с.
4	Павловская Т.А. С#. Программирование на языке высокого уровня. Учебник для вузов СПб.: Питер, 2007. – 432 с.
5	Гради Буч. Объектно-ориентированный анализ и проектирование с примерами приложений на С++. 2-е изд. / Пер. с англ. - М.: "Издательство Бином", СПб: "Невский диалект", 1998 г.
6	Агуров Н.В. С#. Разработка компонентов в MS Visual Studio 2005/2008 /Н.В. Агуров. – СПб.: БХВ-Петербург, 2008. – 480 с.

в) информационные электронно-образовательные ресурсы:

№ п/п	Источник
7	Гамма Э. Приемы объектно ориентированного проектирования. Паттерны проектирования [Электронный ресурс] : справочник / Гамма Э., Хелм Р., Джонсон Р. [и др.]. — Электрон. дан. — М. : ДМК Пресс, 2007. — 376 с. — Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=1220
8	Комлев, Н.Ю. Объектно-Ориентированное Программирование. Хорошая книга для хороших людей [Электронный ресурс] : . — Электрон. дан. — М. : СОЛОН-Пресс, 2014. — 298 с. — Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=64985
9	Подбельский, В.В. Язык С#. Базовый курс [Электронный ресурс] : учебное пособие. — Электрон. дан. — М. : Финансы и статистика, 2015. — 408 с. — Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=65918
10	Подбельский, В.В. Язык С#. Решение задач [Электронный ресурс] : учебное пособие. — Электрон. дан. — М. : Финансы и статистика, 2014. — 296 с. — Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=65919

16. Перечень учебно-методического обеспечения для самостоятельной работы

№ п/п	Источник
1	Фаронов В.В. Программирование на языке С#. – СПб.: Питер, 2007. – 240 с.
2	Гамма Э. Приемы объектно ориентированного проектирования. Паттерны проектирования [Электронный ресурс] : справочник / Гамма Э., Хелм Р., Джонсон Р. [и др.]. — Электрон. дан. — М. : ДМК Пресс, 2007. — 376 с. — Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=1220
3	Комлев, Н.Ю. Объектно-Ориентированное Программирование. Хорошая книга для хороших людей [Электронный ресурс] : . — Электрон. дан. — М. : СОЛОН-Пресс, 2014. — 298 с. — Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=64985

17. Информационные технологии, используемые для реализации учебной дисциплины, включая программное обеспечение и информационно-справочные системы (при необходимости):

среда программирования Visual Studio с поддержкой языка С#.

18. Материально-техническое обеспечение дисциплины:

Мультимедийная лекционная аудитория, компьютерный класс, программное обеспечение: среда программирования Visual Studio с поддержкой языка C#.

19. Оценочные средства для проведения текущей и промежуточной аттестаций

Порядок оценки освоения обучающимися учебного материала определяется содержанием следующих разделов дисциплины:

№ п/п	Разделы дисциплины (модули)	Код компетенции	Код индикатора	Оценочные средства для текущей аттестации
1	Введение в ООП	ОПК-2 ОПК-7 ОПК-13	ОПК-2.9, ОПК-2.10; ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-7.4, ОПК-7.5, ОПК-7.6, ОПК-7.7, ОПК-7.8, ОПК-7.9, ОПК-7.10, ОПК-7.11; ОПК-13.4, ОПК-13.7;	Задания лабораторных работ, КИМ
2	Обзор .NET Framework и библиотеки классов .NET FCL	ОПК-2 ОПК-7 ОПК-13	ОПК-2.9, ОПК-2.10; ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-7.4, ОПК-7.5, ОПК-7.6, ОПК-7.7, ОПК-7.8, ОПК-7.9, ОПК-7.10, ОПК-7.11; ОПК-13.4, ОПК-13.7;	Задания лабораторных работ, КИМ
3	Делегаты и события	ОПК-2 ОПК-7 ОПК-13	ОПК-2.9, ОПК-2.10; ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-7.4, ОПК-7.5, ОПК-7.6, ОПК-7.7, ОПК-7.8, ОПК-7.9, ОПК-7.10, ОПК-7.11; ОПК-13.4, ОПК-13.7;	Задания лабораторных работ, КИМ
4	Обобщенное программирование	ОПК-2 ОПК-7 ОПК-13	ОПК-2.9, ОПК-2.10; ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-7.4, ОПК-7.5, ОПК-7.6, ОПК-7.7, ОПК-7.8, ОПК-7.9, ОПК-7.10, ОПК-7.11; ОПК-13.4, ОПК-13.7;	Задания лабораторных работ, КИМ
5	Понятие качества кода	ОПК-2 ОПК-7 ОПК-13	ОПК-2.9, ОПК-2.10; ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-7.4, ОПК-7.5, ОПК-7.6, ОПК-7.7, ОПК-7.8, ОПК-7.9, ОПК-7.10, ОПК-7.11; ОПК-13.4, ОПК-13.7;	Задания лабораторных работ, КИМ
6	Графическая нотация ОМТ	ОПК-2 ОПК-7 ОПК-13	ОПК-2.9, ОПК-2.10; ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-7.4, ОПК-7.5, ОПК-7.6, ОПК-7.7, ОПК-7.8, ОПК-7.9, ОПК-7.10, ОПК-7.11; ОПК-13.4, ОПК-13.7;	Задания лабораторных работ, КИМ
7	Понятие паттернов проектирования	ОПК-2 ОПК-7 ОПК-13	ОПК-2.9, ОПК-2.10; ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-7.4, ОПК-7.5, ОПК-7.6, ОПК-7.7, ОПК-7.8, ОПК-7.9, ОПК-7.10, ОПК-7.11; ОПК-13.4, ОПК-13.7;	Задания лабораторных работ, КИМ
8	Порождающие паттерны проектирования	ОПК-2 ОПК-7 ОПК-13	ОПК-2.9, ОПК-2.10; ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-7.4, ОПК-7.5, ОПК-7.6, ОПК-7.7, ОПК-7.8, ОПК-7.9, ОПК-7.10, ОПК-7.11; ОПК-13.4, ОПК-13.7;	Задания лабораторных работ, КИМ

9	Структурные паттерны проектирования	ОПК-2 ОПК-7 ОПК-13	ОПК-2.9, ОПК-2.10; ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-7.4, ОПК-7.5, ОПК-7.6, ОПК-7.7, ОПК-7.8, ОПК-7.9, ОПК-7.10, ОПК-7.11; ОПК-13.4, ОПК-13.7;	Задания лабораторных работ, КИМ
10	Поведенческие паттерны проектирования	ОПК-2 ОПК-7 ОПК-13	ОПК-2.9, ОПК-2.10; ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-7.4, ОПК-7.5, ОПК-7.6, ОПК-7.7, ОПК-7.8, ОПК-7.9, ОПК-7.10, ОПК-7.11; ОПК-13.4, ОПК-13.7;	Задания лабораторных работ, КИМ
11	Разработка компонентов и элементов управления	ОПК-2 ОПК-7 ОПК-13	ОПК-2.9, ОПК-2.10; ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-7.4, ОПК-7.5, ОПК-7.6, ОПК-7.7, ОПК-7.8, ОПК-7.9, ОПК-7.10, ОПК-7.11; ОПК-13.4, ОПК-13.7;	Задания лабораторных работ, КИМ
12	Реализация пользовательского интерфейса в C# приложениях. Паттерн MVC.	ОПК-2 ОПК-7 ОПК-13	ОПК-2.9, ОПК-2.10; ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-7.4, ОПК-7.5, ОПК-7.6, ОПК-7.7, ОПК-7.8, ОПК-7.9, ОПК-7.10, ОПК-7.11; ОПК-13.4, ОПК-13.7;	Задания лабораторных работ, КИМ

Промежуточная аттестация
Форма контроля – Зачёт с оценкой

20 Типовые оценочные средства и методические материалы, определяющие процедуры оценивания

20.1 Текущий контроль успеваемости

Контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств: рубежные аттестации лабораторные работы. Перечень заданий для лабораторных работ соответствует темам занятий. Решение каждого задания должно быть доведено до компьютерной реализации.

20.2 Промежуточная аттестация

Для оценивания результатов обучения на зачете с оценкой используются следующие показатели (ЗУНы из 19.1): владение теоретическими основами дисциплины, способностью иллюстрировать ответ примерами, применять теоретические знания для решения практических задач.

Для оценивания результатов обучения на зачете с оценкой используется 4-балльная шкала: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно». Соотношение показателей, критериев и шкалы оценивания результатов обучения.

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок
<i>Обучающийся в полной мере владеет теоретическими основами дисциплины, способен иллюстрировать ответ примерами, фактами, данными научных исследований, применять теоретические знания для решения практических задач.</i>	<i>Повышенный уровень</i>	<i>Отлично</i>
<i>Обучающийся владеет теоретическими основами дисциплины, способен выполнить несложные лабораторные задания, допускает ошибки при устном ответе.</i>	<i>Базовый уровень</i>	<i>Хорошо</i>

<i>Обучающийся владеет частично теоретическими основами дисциплины, фрагментарно способен ответить на вопросы, не умеет применять теоретические знания, но может выполнить несложные лабораторные задачи.</i>	<i>Пороговый уровень</i>	<i>Удовлетворительно</i>
<i>Результаты аттестации не соответствуют трем из перечисленных показателей. Обучающийся демонстрирует отрывочные, фрагментарные знания, допускает грубые ошибки.</i>	–	<i>Неудовлетворительно</i>

20.2.1. Перечень вопросов к зачету

- 1 Объектно-ориентированная парадигма программирования, отличия от структурного программирования. Понятие класса. Понятие объекта (экземпляра класса). Состояние, поведение и уникальность объекта. Основные принципы ООП (обзорно).
- 2 Классы в C#. Члены классов: константы, поля, readonly-поля, свойства, индексообразные, методы, конструкторы, "деструкторы" (финализаторы). Статические и динамические члены классов.
- 3 Язык C#. Общая характеристика языка. Типы данных (обзорно): базовые типы, классы, структуры, перечисления, интерфейсы, строки, массивы (многомерные и ступенчатые). Методы, передача параметров в методы (ref, out). Пространства имен. Управляющие конструкции и т.д. Точка входа в программу на языке C#.
- 4 Перехват исключений в языке C#. Порождение исключений. Основные принципы использования исключений.
- 5 Ссылочная модель и автоматическая сборка мусора в языке C#. Задача финализаторов. Ошибки, приводящие к утечкам памяти.
- 6 Инкапсуляция: принципы и реализация в языке C#. Модификаторы доступа. Ключевое слово this.
- 7 Наследование: принципы и реализация в языке C#. Множественно наследование (реализация) от нескольких интерфейсов. Ключевые слова this и super. Операторы as и is, модификатор sealed.
- 8 Полиморфизм: принципы и реализация в языке C#. Виртуальные методы, абстрактные методы, перегрузка методов. Ключевые слова this и super. Операторы as и is. Реализация полиморфизма: таблицы виртуальных методов.
- 9 Перегрузка операций.
- 10 Делегаты и события в языке C#. Создание делегатов на основе именованных и неименованных методов. Функциональный стиль программирования на основе делегатов. Реализация своих событий.
- 11 Обобщенное программирование: основная идея. Обобщенное программирование в C# на основе generics. Пример описания и использования обобщенного класса (например, стека). Ограничения в generics.
- 12 Коллекции и итераторы в языке C#. Интерфейсы IEnumerable/IEnumerable<T>, IEnumerator/IEnumerator<T>, ключевые слова foreach и yield. Основные коллекции в библиотеке FCL (System.Collections.Generic: List, Dictionary, Stack, Queue).
- 13 Ввод-вывод в FCL. Понятие потока данных, класс System.IO.Stream. Ввод-вывод текстовых данных, пример чтения/записи текстовых файлов. Понятие сериализации.
- 14 Понятие качества ПО.
- 15 Паттерны проектирования: описание, назначение, принципы использования, классификация.

- 16 Графическая нотация ОМТ.
- 17 Порождающие паттерны GoF.
- 18 Структурные паттерны GoF.
- 19 Поведенческие паттерны GoF.
- 20 Компоненты и элементы управления.
- 21 Паттерн Model-View-Controller и привязка данных.