

Министерство науки и высшего образования  
Российской Федерации

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

Факультет математики и информационных технологий  
Кафедра прикладной математики и теории систем управления



УТВЕРЖДАЮ

Проректор

*Машаров*

П.А. Машаров

29 марта 2024 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

**ВВЕДЕНИЕ В ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ  
ПРОГРАММИРОВАНИЕ**

Укрупненная группа направлений  
подготовки  
Программа высшего образования  
Направление подготовки

Профиль подготовки

Квалификация  
Форма обучения

02.00.00 Компьютерные и  
информационные науки  
Программа бакалавриата  
02.03.02 Фундаментальная информатика и  
информационные технологии  
Фундаментальная информатика и  
информационные технологии  
Бакалавр  
Очная

Рабочая программа адаптирована для лиц  
с ограниченными возможностями здоровья и инвалидов

Донецк 2024

Рабочая программа дисциплины «Введение в объектно-ориентированное программирование» для обучающихся по направлению подготовки 02.03.02 Фундаментальная информатика и информационные технологии (Профиль подготовки: Фундаментальная информатика и информационные технологии), составлена на основании Федерального государственного образовательного стандарта высшего образования – бакалавриат по направлению подготовки 02.03.02 Фундаментальная информатика и информационные технологии, утвержденного приказом Министерства образования и науки Российской Федерации от 23 августа 2017 г. № 808 (с изм. и доп.), Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры, утвержденного приказом Министерства науки и высшего образования Российской Федерации от 06 апреля 2021 г. № 245 (с изм. и доп.), в соответствии с учебным планом, утвержденным Ученым советом ФГБОУ ВО «ДонГУ» для набора 2024 года.

Разработчик:

доцент кафедры прикладной математики и  
теории систем управления



Л.А. Рыбалко

Рабочая программа одобрена на заседании кафедры прикладной математики и теории систем управления.

Протокол от 26.03.2024 г. № 8

Заведующий кафедрой



Д.В. Шевцов

СОГЛАСОВАНО:

Декан факультета математики и  
информационных технологий  
28.03.2024 г.



И.А. Моисеенко

Учебно-методическая комиссия факультета математики и информационных технологий.  
Протокол от 27.03.2024 г. № 3.

Председатель



Л. И. Селякова

Руководитель основной профессиональной  
образовательной программы,

д-р техн. наук, доц.

26.03.2024 г.



Д.В. Шевцов

## 1. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

1.1. Требования к предварительной подготовке обучающихся, предшествующие и сопутствующие дисциплины, на которых основывается изучение данной:

базовая подготовка по математике в объеме программы средней школы;

дисциплины программы бакалавриата: Основы программирования, Архитектура вычислительных систем, Математический анализ, Алгебра и геометрия, Дискретная математика.

1.2. Дисциплины, курсовые работы и практики, для которых освоение данной дисциплины необходимо как предшествующее:

Математические модели в информационных технологиях 1-8, Прикладные информационные технологии 1-8, Языки программирования, Операционные системы, а также дисциплин магистратуры Современные методы цифровой обработки информации, Анализ и обработка изображений.

## 2. ОПИСАНИЕ ДИСЦИПЛИНЫ

### 2.1. Общая характеристика

Наименование показателя	Значение показателя
Название образовательной программы	02.03.02 Фундаментальная информатика и информационные технологии (Профиль подготовки: Фундаментальная информатика и информационные технологии)
Шифр и название в соответствии с учебным планом	Б1.Б.27. Введение в объектно-ориентированное программирование
Часть образовательной программы	Базовая часть
Количество зачетных единиц / всего часов	4 / 144

### 2.2. Распределение часов по периодам обучения

Форма обучения	курс	семестр	Общее количество часов					Форма контроля
			лекционных	лабораторных	практических	самостоятельной работы	всего	
Очная	2	3	34	34		76	144	экзамен

## 3. ЦЕЛИ ДИСЦИПЛИНЫ

Углубленная подготовка в области объектно-ориентированного программирования (ООП), алгоритмизации и создания классов; объектно-ориентированного подхода; изучение инструментов языка C# для реализации объектно-ориентированного программирования; составление несложных информационно-математических моделей, оперирование с абстрактными объектами; формирование у студентов научного подхода.

## 4. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ КОМПОНЕНТА ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ, ИХ ИНДИКАТОРЫ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

### 4.1. Компетенции

ОПК-3. Способен к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных

и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям;

#### 4.2. Индикаторы компетенций

ОПК-3.3. Применяет анализ математических моделей, создает инновационные методы решения прикладных задач профессиональной деятельности в области информатики и математического моделирования

#### 4.3. Результаты обучения

ОПК-3.3.1. Знает общенаучные базовые понятия математических и естественных наук, фундаментальной информатики и информационных технологий; способен применять в профессиональной деятельности современные языки программирования, операционные системы, электронные библиотеки и пакеты программ.

ОПК-3.3.2. Умеет применять в научно-исследовательской и прикладной деятельности современный математический аппарат, основные законы естествознания, современные языки программирования и программное обеспечение; операционные системы и сетевые технологии.

ОПК-3.3.3. Владеет методами системного и прикладного программирования, основными положениями и концепциями в области математических, информационных и имитационных моделей; аргументированно выбирает метод решения задачи, устанавливает свойства математических объектов, закономерности между ними, оценивает и анализирует полученный результат, строит математические модели для решения профессиональных задач.

## 5. ПРОГРАММА ДИСЦИПЛИНЫ

Название темы	Краткое содержание темы (вопросы темы)
Разработка программ и объектно-ориентированное программирование	Разработка программ и объектно-ориентированное программирование. Жизненный цикл программного продукта. Абстракция. Абстрактные типы данных. Основные парадигмы программирования. Процедурное программирование. Модульное программирование. Объектно-ориентированное программирование. Основные цели объектно-ориентированного программирования. Объекты. Классы. Базовые принципы ООП: пакетирование (инкапсуляция), наследование, полиморфизм, передача сообщений. Объектно-ориентированные языки программирования
Язык C#	Язык C#. Цели создания языка. Операции языка. Распределение памяти в C#. Операция расширения области видимости. Ссылки: особенности использования. Ввод-вывод в стиле C#.
Классы в языке C#	Классы в языке C# Поддержка АД в C#. Классы и объекты. Описание класса. Компонентные данные и компонентные функции. Спецификаторы доступа: открытые, защищенные и закрытые компоненты класса. Доступ к открытым и закрытым компонентным данным. Прямой и косвенный доступ к компонентам классов. Операции доступа. Скрытый указатель this.
Конструкторы и деструкторы. Конструктор копирования	Конструкторы и деструкторы. Конструктор по умолчанию. Конструктор инициализации. Конструктор копий. Свойства конструкторов. Перегрузка конструктора. Конструктор с присваиванием и со списком инициализации. Деструктор, его свойства. Конструктор копирования. Поверхностное копирование. Побочные эффекты при работе функций. Глубокое копирование. Конструктор копирования. Свойства конструктора копирования. Предотвращение передачи объектов класса по значению.

Статические компоненты классов. Константы	Константы в классе. Инициализация нестатических полей класса. Статические компоненты классов. Статические переменные и функции. Статические компоненты классов. Особенности статических полей. Инициализация статических полей. Статические методы. Инициализация статических массивов объектов. Паттерн проектирования «синглет»
Перегрузка операций	Перегрузка операций. Механизмы реализации перегрузки операций. Особенности перегрузки операций. Компонентные функции операции. Перегрузка унарных и бинарных операций. Перегрузка операций инкремента и декремента, префиксная и постфиксная формы. Перегрузка операции присваивания. Перегрузка операции индексирования. Функциональная и операторная формы перегрузки операций.
Наследование	Наследование. Базовые и производные классы. Простое наследование. Множественное наследование. Ненаследуемые члены класса. Управление доступом к базовому классу. Виды наследования. Моделирование объектно-ориентированных проектов с помощью языка UML. Конструкторы, деструкторы и наследование. Предопределенные стандартные преобразования. Отношения между классами: «является», «содержит» и «подобен»
Полиморфизм в языке C#	Полиморфизм в языке C#. Связывание. Статическое и динамическое связывание. Виртуальные функции. Полиморфные объекты. Особенности виртуальных методов. Таблица виртуальных методов VTBL. Указатель VPTR. Абстрактные базовые классы. Чистые виртуальные функции. Интерфейсы
Шаблоны	Полиморфизм функций. Перегрузка функций. Неоднозначность и перегрузка. Определение адреса перегруженной функции. Шаблоны классов. Определение метода вне класса–шаблона. Инстанцирование (актуализация) шаблонов. Шаблоны функций: глобальные локальные функции. Специализация шаблонов. Композиция и шаблоны. Шаблоны и наследование. Особенности шаблонов.
Исключительные ситуации в C#	Исключительные ситуации в C#. Обработка исключительных ситуаций. Обработка ошибок в C#. Генерирование исключений. Перехватывание исключений. Использование вложенных блоков try/catch. Раскрутка стека. Стандартные исключения в C#. Исключения и конструкторы. Информация о типе времени выполнения. Операторы приведения типа.
Стандартная библиотека шаблонов (STL)	Стандартная библиотека шаблонов (STL). Назначение и состав библиотеки. Контейнеры. Последовательные контейнеры. Векторы. Списки. Деки. Операции с деками. Объявление и инициализация дека. Стеки. Объявление и инициализация стека. Очереди. Объявление и инициализация очереди. Ассоциативные контейнеры. Универсальные алгоритмы.

## 6. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

### 6.1. Форма обучения – очная, курс – 2, семестр – 3

Наименования разделов и тем	Количество часов				
	Лекц.	Лабор.	Практ.	СРС+К	Всего
Разработка программ и объектно-ориентированное программирование	2	2		4	8

Язык C#	4	4		8	16
Классы в языке C#	4	4		8	16
Конструкторы и деструкторы. Конструктор копирования	2	2		6	10
Статические компоненты классов. Константы	2	2		6	10
Перегрузка операций	4	4		8	16
Наследование	4	4		8	16
Полиморфизм в языке C#	2	2		6	10
Шаблоны	4	4		8	16
Исключительные ситуации в C#	4	4		8	16
Стандартная библиотека шаблонов (STL)	2	2		6	10
ИТОГО ПО КОМПОНЕНТУ ОПОП	34	34		76	144

## 7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (СРЕДСТВА) ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

### 7.1. Контрольные вопросы

1. Средства ввода-вывода в языке C#
2. Операции языка C#.
3. Операции выделения и освобождения памяти в C#.
4. Абстракция данных и классы в языке C#.
5. Конструкторы и деструкторы в классах.
6. Наследование классов.
7. Композиция классов.
8. Статические члены классов.
9. Статическое и динамическое связывание.
10. Виртуальные функции.
11. Перегрузка операций. Особенности перегрузки унарных операций.
12. Перегрузка операций. Особенности перегрузки бинарных операций.
13. Абстрактные базовые классы.
14. Паттерны проектирования. Итератор и их важность в библиотеке STL.
15. Шаблоны функций в языке C#.
16. Шаблоны классов в языке C#.
17. Стандартная библиотека языка C#. Шаблоны классов.
18. Стандартная библиотека языка C#. Обобщенные алгоритмы.
19. Перегрузка функций в языке C#.

### 7.2. Темы индивидуальных заданий (примеры)

– Процедурное программирование в языке C#. (Разработать приложение с использованием функций ввода массива, обработки массива и вывода массива (если результат является массивом). Найти максимальный элемент в каждом массиве и максимальный среди максимальных).

– Реализация простого класса в комплексных числах (Реализовать методы: конструктор, деструктор, метод вычисления модуля комплексного числа и корня  $n$ -й степени, метод вывода комплексного числа. Выполнить поддержку свойств. Выполнить операторы перегрузки: сложение (+), вычитание (-), умножение (\*), деление (/) комплексных чисел, увеличение на 1 (++) действительной и мнимой части. Получить результаты:  $R=a^{1/3}++ +a+b*d$ . Найти модуль числа  $R$ . Увеличить на 1 действительную и мнимую часть  $R$ ).

– Реализация простого наследования класса (Создать базовый класс «массив», описав в нём функции добавления, удаления, вывода на экран и подсчета суммы



положительных элементов массива. Породить от него класс «упорядоченный массив», переопределив функцию добавления элементов массива).

– Имитационное объектно-ориентированное моделирование (Разработать класс, моделирующий объект. Любое изменение состояния объекта, описанного в задании, должно отображаться на экране).

– Работа со стандартными шаблонными классами (Разработать программу с интерфейсом на основе форм, предусмотреть проверки на вводимые значения с помощью исключений).

### 7.3. Темы письменных работ (типы задач)

Контрольные работы определяются темами индивидуальных заданий.

Контрольная работа по проверке теоретических знаний – по всем темам, с использованием указанных выше контрольных вопросов.

### 7.4. Образец содержания экзаменационного билета (при наличии экзамена по дисциплине)

Экзаменационный билет № \_

1. Операции языка C#.
2. Конструкторы и деструкторы.

В случае ведения учебного процесса с использованием электронного обучения и дистанционных образовательных технологий, содержание билета может отличаться от приведенного.

## 8. РАСПРЕДЕЛЕНИЕ БАЛЛОВ, КОТОРЫЕ ПОЛУЧАЮТ ОБУЧАЮЩИЕСЯ

Общая оценка знаний обучающихся по дисциплине проводится по 100-балльной шкале исходя из максимума, приведенного в таблице ниже. Организационно-учебная работа в аудитории оценивается на основе таких критериев как посещаемость занятий, своевременное и качественное выполнение домашних заданий, активность во время проведения лекционных и практических занятий (участие в обсуждении текущего и пройденного материала, решение задач и т.п.).

### 8.1. Семестр 3

Номера тем	Виды работ	Максимальное количество баллов
1-11	Организационно-учебная работа в аудитории	10
	Самостоятельная работа	10
	Выполнение индивидуальных заданий	50
	Контрольная работа по теоретическому материалу	10
ИТОГО		80
Экзамен		20
Общий итог за семестр		100

### Соответствие баллов оценке

Количество баллов из 100	ECTS	Оценка по пятибалльной шкале	
		Экзамен, дифференцированный зачет	Зачет
90-100	A	отлично	зачтено

80-89	B	хорошо	зачтено
75-79	C		зачтено
70-74	D	удовлетворительно	зачтено
60-69	E		зачтено
35-59	FX	неудовлетворительно	не зачтено
0-34	F		не зачтено

## 9. ОБЕСПЕЧЕНИЕ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ И ИНВАЛИДОВ

В ходе реализации дисциплины используются следующие дополнительные методы обучения, текущего контроля успеваемости и промежуточной аттестации обучающихся в зависимости от их индивидуальных особенностей:

- 1) для слепых и слабовидящих:
  - лекции оформляются в виде электронного документа, доступного с помощью компьютера со специализированным программным обеспечением;
  - для выполнения задания при необходимости предоставляется увеличивающее устройство; возможно также использование собственных увеличивающих устройств;
  - письменные задания оформляются увеличенным шрифтом.
- 2) для глухих и слабослышащих:
  - лекции оформляются в виде электронного документа;
  - письменные задания выполняются на компьютере в письменной форме;
  - экзамен проводится в письменной форме на компьютере; возможно проведение в форме тестирования.
- 3) для лиц с нарушениями опорно-двигательного аппарата:
  - лекции оформляются в виде электронного документа, доступного с помощью компьютера со специализированным программным обеспечением;
  - письменные задания выполняются на компьютере;
  - экзамен и зачёт проводятся в устной форме или выполняются в письменной форме на компьютере.

При необходимости предусматривается увеличение времени для подготовки ответа.

Процедура проведения промежуточной аттестации для обучающихся устанавливается с учётом их индивидуальных психофизических особенностей. Промежуточная аттестация может проводиться в несколько этапов.

Проведение процедуры оценивания результатов обучения допускается с использованием дистанционных образовательных технологий.

Обеспечивается доступ к информационным и библиографическим ресурсам в сети Интернет для каждого обучающегося в формах, адаптированных к ограничениям их здоровья и восприятия информации:

- 1) для слепых и слабовидящих:
  - в печатной форме увеличенным шрифтом;
  - в форме электронного документа;
- 2) для глухих и слабослышащих:
  - в печатной форме;
  - в форме электронного документа.
- 3) для обучающихся с нарушениями опорно-двигательного аппарата:
  - в печатной форме;
  - в форме электронного документа.



## 10. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОГО ПРОЦЕССА

Учебные занятия проводятся в Главном корпусе ДонГУ (г. Донецк, пр. Гурова, 6). Для проведения лабораторных занятий требуется аудитория, оборудованная меловой или маркерной доской, мультимедийный проектор и экран, ноутбук, комплект учебной мебели для студентов, рабочее место преподавателя, выход в Интернет – проводной или с использованием Wi-Fi.

Для самостоятельной работы используются текстовые и электронные ресурсы Научной библиотеки университета и других электронных библиотечных баз данных, учебно-методическое обеспечение, представленное в учебно-методическом кабинете Главного корпуса (ауд.401).

Обучающиеся имеют возможность использовать учебные материалы по дисциплине, размещенные на платформе Moodle Центра дистанционного образования ФГБОУ ВО «ДонГУ». При изучении дисциплины применяются электронное обучение и дистанционные образовательные технологии.

С использованием ресурсов платформы дистанционного образования осуществляется текущий контроль знаний обучающихся на основе тестирования и проверки результатов самостоятельной работы.

## 11. РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

### 11.1. Основная литература

1. Рихтер Д. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#. 4-е изд./ Рихтер Д. – СПб: Питер, 2014. – 896 с
2. Джонсон Р. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Джонсон Р., Гамма Э., Халм Р. – СПб: Питер, 2016. – 368 с.
3. Рамбо, Дж. UML 2.0. Объектно-ориентированное моделирование и разработка [Текст]: пер. с англ. / Джеймс Рамбо, Мартин Блаха. – СПб: Питер, 2007. – 544 с.
4. Буч, Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++ [Текст]: пер. с англ. / Гради Буч. – М: Бином, СПб: Невский диалект, 1999. – 560 с.
5. Пышкин, Е.В. Основные концепции и механизмы объектно-ориентированного программирования [Текст]/ Е.В.Пышкин. – СПб: БХВ-Петербург, 2005. – 640 с.
6. Шилдт, Г.. C# 4.0: полное руководство [Текст]: Пер. с англ. / Герберт Шилдт. – М.: ООО "И.Д. Вильямс", 2011. – 1056 с.
7. Дейтел, Х. C# в подлиннике. Наиболее полное руководство [Текст]: пер. с англ./ Харви Дейтел, Пол Дейтел. – СПб: БХВ-Петербург, 2006 г.. – 1056 с.
8. Троелсен, Э. Язык программирования C# 2010 и платформа .NET 4 [Текст]: Пер. с англ. / Эндрю Троелсен. – М.: ООО "И.Д. Вильямс", 2011. – 1392 с.

### 11.2. Дополнительная литература

9. Албахари Дж. C# 6.0. Справочник. Полное описание языка / Дж. Албахари, Б. Албахари – М.: ООО "И.Д. Вильямс", 2016. – 1040 с.
10. Холл Г. Адаптивный код на C#. Проектирование классов и интерфейсов, шаблоны и принципы SOLID./ Холл Г. - М.: ООО "И.Д. Вильямс", 2016. – 432 с.

## 12. ИНФОРМАЦИОННЫЕ РЕСУРСЫ

1. **Национальная электронная библиотека (НЭБ):** федеральная государственная информационная система / Министерство Культуры РФ; Российская государственная библиотека. – Москва, 2019- . – URL: <https://rusneb.ru/> (дата обращения:

01.09.2023). – Режим доступа: свободный, подписка. Необходима установка программного обеспечения. – Текст: электронный.

2. **eLIBRARY.RU**: научная электронная библиотека: сайт. – Москва, 2000- . – URL: <https://elibrary.ru> (дата обращения: 01.09.2023). – Режим доступа: для авторизов. пользователей. – Текст: электронный.

3. Научная электронная библиотека «**КиберЛенинка**»: сайт / Ассоциация «Открытая наука». – Москва, 2014- . – URL: <https://cyberleninka.ru/>. – Режим доступа: свободный. – Текст: электронный.

4. Электронно-библиотечная система «**Лань**»: [сайт]. – URL: <https://e.lanbook.com> (дата обращения: 01.09.2023). – Режим доступа: для авторизов. пользователей. – Текст: электронный.

5. **ЭБС Юрайт**: электронная библиотечная система: сайт. – Москва, 2013. – URL: <https://biblio-online.ru> (дата обращения: 01.09.2023). – Режим доступа: для авторизов. пользователей. – Текст: электронный.

6. **Электронно-библиотечная система ДонГУ**: сайт / ФГБОУ ВО «ДонГУ». – Донецк, 2016- . – URL: <http://library.donnu.ru/> (дата обращения: 01.09.2023). – Режим доступа: свободный. – Текст: электронный.

7. **Электронный каталог** Научной библиотеки ДонГУ: раздел сайта / НБ ДонГУ. – Текст: электронный // ЭБС ДонГУ: сайт. – URL: <http://library.donnu.ru/catalog/> (дата обращения: 01.09.2023). – Режим доступа: поиск свободный, электронные документы – для пользователей ДонГУ.

8. **Электронный архив** ДонГУ: раздел сайта / НБ ДонГУ. – Текст: электронный // ЭБС ДонГУ: сайт. – URL: <http://repo.donnu.ru/> (дата обращения: 01.09.2023). – Режим доступа: свободный.

### 13. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

1. Windows 7 PRO (корпоративная лицензия ДонГУ № 46484614)
2. Microsoft Office (корпоративная лицензия ДонГУ № 46472919)
3. Microsoft Visual Studio (лицензия программы Dream Spark для высших учебных заведений)
4. Антивирус Касперского, Adobe Acrobat Reader, xPDF (лицензии GPL, Apache, BSD для свободного программного обеспечения).