



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Магнитогорский государственный технический университет им. Г.И.  
Носова»



УТВЕРЖДАЮ  
Директор ИЭиАС  
В.Р. Храмшин

03.02.2026 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

***ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ***

Направление подготовки (специальность)  
09.03.01 Информатика и вычислительная техника

Направленность (профиль/специализация) программы  
Программное обеспечение средств вычислительной техники и автоматизированных систем

Уровень высшего образования - бакалавриат

Форма обучения  
очная

Институт/ факультет	Институт энергетики и автоматизированных систем
Кафедра	Вычислительной техники и программирования
Курс	2
Семестр	4

Магнитогорск  
2026 год

Рабочая программа составлена на основе ФГОС ВО - бакалавриат по направлению подготовки 09.03.01 Информатика и вычислительная техника (приказ Минобрнауки России от 19.09.2017 г. № 929)

Рабочая программа рассмотрена и одобрена на заседании кафедры  
Вычислительной техники и программирования  
29.01.2026, протокол № 7

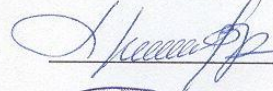
Зав. кафедрой



О.С. Логунова

Рабочая программа одобрена методической комиссией ИЭиАС  
03.02.2026 г. протокол № 5

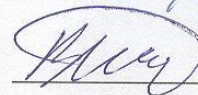
Председатель



В.Р. Храмшин

Рабочая программа составлена:

ст. преподаватель кафедры кафедры ВТиП,



В.Е. Торчинский

Рецензент:

директор НИИ «Промбезопасность», д-р техн. наук



М.Ю. Наркевич

## Лист актуализации рабочей программы

---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2027 - 2028 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от \_\_\_\_\_ 20\_\_ г. № \_\_\_\_  
Зав. кафедрой \_\_\_\_\_ О.С. Логунова

---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2028 - 2029 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от \_\_\_\_\_ 20\_\_ г. № \_\_\_\_  
Зав. кафедрой \_\_\_\_\_ О.С. Логунова

---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2029 - 2030 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от \_\_\_\_\_ 20\_\_ г. № \_\_\_\_  
Зав. кафедрой \_\_\_\_\_ О.С. Логунова

---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2030 - 2031 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от \_\_\_\_\_ 20\_\_ г. № \_\_\_\_  
Зав. кафедрой \_\_\_\_\_ О.С. Логунова

## 1 Цели освоения дисциплины (модуля)

Целями освоения дисциплины «Объектно-ориентированное программирование» являются:

формирование у студентов понятия об объектно-ориентированной парадигме моделирования бизнес-процессов и ее современных реализациях;

освоение методологии адаптации и применения объектно-ориентированного программного обеспечения для моделирования бизнес-процессов;

выработка компетенций, позволяющих определять применимость данного объектно-ориентированного программного обеспечения в конкретных условиях;

выработка компетенций, позволяющих создавать комплексные решения, в которых эффективно используется объектно-ориентированное программное обеспечение.

Для достижения поставленных целей в курсе «Объектно-ориентированное программное обеспечение» решаются задачи:

освоение методов объектно-ориентированного анализа предметной области;

освоение методов объектно-ориентированного моделирования;

изучение современных применений объектно-ориентированной парадигмы программирования;

изучение современных объектных систем моделирования бизнес-процессов.

## 2 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина Объектно-ориентированное программирование входит в часть учебного плана формируемую участниками образовательных отношений образовательной программы.

Для изучения дисциплины необходимы знания (умения, владения), сформированные в результате изучения дисциплин/ практик:

Программирование

Информатика

Прикладная математика

Математическая логика и дискретная математика

Знания (умения, владения), полученные при изучении данной дисциплины будут необходимы для изучения дисциплин/практик:

Архитектура виртуальной реальности

Проектная деятельность

Программная платформа RadixWare

Проектирование программных средств

Средства программирования мобильных приложений

Паттерное программирование

## 3 Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля) и планируемые результаты обучения

В результате освоения дисциплины (модуля) «Объектно-ориентированное программирование» обучающийся должен обладать следующими компетенциями:

Код индикатора	Индикатор достижения компетенции
ПК-2	Способность к анализу проблемной ситуации, разработке требований к системе, постановке целей создания системы, разработке концепции системы и технического задания на создание системы, представления концепции, технического задания на систему и изменений в них заинтересованным лицам
ПК-2.1	Оценивает выбор средств и методов для проведения системного анализа при проектировании программного обеспечения для

	автоматизированных систем
ПК-6 Способность к формализации и алгоритмизации поставленных задач, к написанию программного кода с использованием языков программирования, определения и манипулирования данными и оформлению программного кода в соответствии установленными требованиями	
ПК-6.1	Оценивает качество математической модели при формализации задачи предметной области
ПК-6.2	Оценивает качество разработанных алгоритмов для последующего кодирования
ПК-6.3	Оценивает выбор программных средств для программирования и манипулирования данными в соответствии установленными требованиями

#### 4. Структура, объём и содержание дисциплины (модуля)

Общая трудоемкость дисциплины составляет 4 зачетных единиц 144 академических часов, в том числе:

- контактная работа – 72 академических часов;
- аудиторная – 68 академических часов;
- внеаудиторная – 4 академических часов;
- самостоятельная работа – 36,3 академических часов;
- в форме практической подготовки – 0 академических часов;
- подготовка к экзамену – 35,7 академических часов

Форма аттестации - экзамен

Раздел/ тема дисциплины	Семестр	Аудиторная контактная работа (в академических часах)			Самостоятельная работа студента	Вид самостоятельной работы	Форма текущего контроля успеваемости и промежуточной аттестации	Код компетенции
		Лек.	лаб. зан.	практ. зан.				
1. Внешние и внутренние факторы качества ПО								
1.1 Внешние и внутренние факторы качества ПО: корректность, устойчивость, расширяемость, повторное использование, совместимость, эффективность, переносимость, простота использования, функциональность	4	2	2		6	1. Самостоятельное изучение учебной и научной литературы. 2. Подготовка к лабораторному занятию	Проверка индивидуальных заданий	ПК-6.1, ПК-6.2, ПК-6.3
Итого по разделу		2	2		6			
2. Объектная модель								
2.1 Основные понятия. Абстрагирование. Контрактная модель программирования. Примеры абстракций	4	2	2		5	1. Самостоятельное изучение учебной и научной литературы. 2. Подготовка к лабораторному занятию	Проверка индивидуальных заданий	ПК-6.1, ПК-6.2, ПК-6.3
2.2 Инкапсуляция. C++, JavaScript. Примеры		6	6		10	1. Самостоятельное изучение учебной и научной литературы. 2. Подготовка к лабораторному занятию	Проверка индивидуальных заданий	ПК-6.1, ПК-6.2, ПК-6.3
2.3 Модульность.		6	6		10	1.	Проверка	ПК-6.1, ПК-

Иерархия. Наследование и агрегация. С++, JavaScript. Примеры						Самостоятельное изучение учебной и научной литературы. 2. Подготовка к лабораторному занятию	индивидуальных заданий	6.2, ПК-6.3
2.4 Типизация. Полиморфизм	4	4	4		2	1. Самостоятельное изучение учебной и научной литературы. 2. Подготовка к лабораторному занятию	Проверка индивидуальных заданий	ПК-6.1, ПК-6.2, ПК-6.3
2.5 Параллелизм. Сохраняемость. Формат JSON в JavaScript		4	4		1	1. Самостоятельное изучение учебной и научной литературы. 2. Подготовка к лабораторному занятию	Проверка индивидуальных заданий	ПК-6.1, ПК-6.2, ПК-6.3
Итого по разделу		22	22		28			
3. Процесс объектно-ориентированного проектирования								
3.1 Микропроцесс проектирования. Выявление классов и объектов на данном уровне абстракции, выяснение семантики этих классов и объектов, выявление связей между этими классами и объектами, спецификация интерфейса и реализация этих классов и объектов	4	6	6		1	1. Самостоятельное изучение учебной и научной литературы. 2. Подготовка к лабораторному занятию	Проверка индивидуальных заданий	ПК-6.1, ПК-6.2, ПК-6.3
3.2 Макропроцесс проектирования. Концептуализация, анализ, проектирование, эволюция, сопровождение		4	4		1,3	1. Самостоятельное изучение учебной и научной литературы. 2. Подготовка к лабораторному занятию	Проверка индивидуальных заданий	ПК-6.1, ПК-6.2, ПК-6.3
Итого по разделу		10	10		2,3			
Итого за семестр		34	34		36,3		экзамен	
Итого по дисциплине		34	34		36,3		экзамен	

## **5 Образовательные технологии**

1. Традиционные образовательные технологии, ориентированные на организацию образовательного процесса и предполагающую прямую трансляцию знаний от преподавателя к студенту.

Формы учебных занятий с использованием традиционных технологий:

Информационная лекция – последовательное изложение материала в дисциплинарной логике, осуществляемое преимущественно вербальными средствами (монолог преподавателя).

Практическое занятие, посвященное освоению конкретных умений и навыков по предложенному алгоритму.

2. Технологии проблемного обучения – организация образовательного процесса, которая предполагает постановку проблемных вопросов, создание учебных проблемных ситуаций для стимулирования активной познавательной деятельности студентов.

Формы учебных занятий с использованием технологий проблемного обучения:

Проблемная лекция – изложение материала, предполагающее постановку проблемных и дискуссионных вопросов, освещение различных научных подходов, авторские комментарии, связанные с различными моделями интерпретации изучаемого материала.

3. Интерактивные технологии – организация образовательного процесса, которая предполагает активное и нелинейное взаимодействие всех участников, достижение на этой основе лично-значимого для них образовательного результата.

Формы учебных занятий с использованием специализированных интерактивных технологий:

Лекция «обратной связи» – лекция-провокация (изложение материала с заранее запланированными ошибками), лекция-беседа, лекция-дискуссия, лекция-прессконференция.

Практическое занятие в форме практикума – организация учебной работы, направленная на решение комплексной учебно-познавательной задачи, требующей от студента применения как научно-теоретических знаний, так и практических навыков.

## **6 Учебно-методическое обеспечение самостоятельной работы обучающихся**

Представлено в приложении 1.

## **7 Оценочные средства для проведения промежуточной аттестации**

Представлены в приложении 2.

## **8 Учебно-методическое и информационное обеспечение дисциплины**

### **а) Основная литература:**

1. Буч, Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++, 3 е изд. / Г. Буч – М.: «Издательство Бином», СПб.: «Невский диалект», 2023. – 560 с.

2. Колесов Ю.Б. Моделирование систем. Объектно-ориентированный подход [Электронный ресурс]. / Ю.Б. Колесов, Ю.Б. Сениченков – СПб.: БХВ – Петербург, 2024. – 192 с. - Режим доступа к ресурсу: <http://ibooks.ru/reading.php?productid=322387> . Заглавие с экрана ISBN 978-5-9912-0085-7

### **б) Дополнительная литература:**

1. Леоненков, А.В. Объектно-ориентированный анализ и проектирование с использованием UML и Rational Rose. Учебное пособие / А.В. Леоненков. – М.:

Интернет – Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2016. – 320 с.

2. Зарецкий, М.В. Объектно-ориентированное программирование: Учеб. пособие. / М.В. Зарецкий, Ю.Б. Кухта. Магнитогорск. МГТУ, 2008 63 с.

**в) Методические указания:**

**г) Программное обеспечение и Интернет-ресурсы:**

**Программное обеспечение**

Наименование ПО	№ договора	Срок действия лицензии
MS Visual Studio 2017 Community Edition	свободно распространяемое ПО	бессрочно
Браузер Mozilla Firefox	свободно распространяемое ПО	бессрочно

**Профессиональные базы данных и информационные справочные системы**

Название курса	Ссылка

**9 Материально-техническое обеспечение дисциплины (модуля)**

Материально-техническое обеспечение дисциплины включает:

Лекционная аудитория — мультимедийные средства хранения, передачи и представления информации.

Компьютерный класс — персональные компьютеры с компиляторами C++, пакетом Office, выходом в Интернет и с доступом в электронную информационно-образовательную среду университета.

Аудитории для самостоятельной работы: компьютерные классы; читальные залы библиотеки — все классы УИТиАСУ с персональными компьютерами, выходом в Интернет и с доступом в электронную информационно-образовательную среду университета.

Аудитории для групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации — ауд. 282 и классы УИТиАСУ.

Помещения для самостоятельной работы обучающихся, оснащенных компьютерной техникой с возможностью подключения к сети «Интернет» и наличием доступа в электронную информационно-образовательную среду организации — классы УИТиАСУ.

Помещения для хранения и профилактического обслуживания учебного оборудования — ауд. 379.

По дисциплине «Объектно-ориентированное программирование» предусмотрена аудиторная и внеаудиторная самостоятельная работа обучающихся.

Аудиторная самостоятельная работа студентов предполагает решение контрольных задач на лабораторно-практических занятиях.

Примерные аудиторные контрольные работы (АКР):

### **Раздел 1.**

Реализовать на C++ и JavaScript класс "Длинное целое".

Переопределить операции

сложения, вычитания (базовый уровень);

умножения, деления, остаток от деления (повышенный уровень).

Переопределить операции вывода, присваивания и сравнения.

Один из конструкторов должен иметь параметр строкового типа.

Протестировать проект на наборе случайных значений

*Отчет по заданию*

В Word создать спецификацию реализованного класса (описание полей и методов).

### **Раздел 2.**

Реализовать иерархии классов "Гладиаторы", "Оружие гладиаторов" и "Защитные средства".

Промоделировать бой между гладиаторами.

*Отчет по заданию*

Создать презентацию с описанием реализованных иерархий (не менее шести слайдов).

### **Раздел 3.**

Визуализация задания из раздела 2

*Отчет по заданию*

Создать презентацию с описанием разработанной программы (не менее шести слайдов).

а) Планируемые результаты обучения и оценочные средства для проведения промежуточной аттестации:

Код индикатора	Индикатор достижения компетенции	Оценочные средства
<p><b>ПК-6: Способность к формализации и алгоритмизации поставленных задач, к написанию программного кода с использованием языков программирования, определения и манипулирования данными и оформлению программного кода в соответствии установленными требованиями</b></p>		
<p>ПК-6.1</p>	<p>Оценивает качество математической модели при формализации задачи предметной области</p>	<p><i>Перечень теоретических вопросов</i></p> <ol style="list-style-type: none"> <li>1. Абстрагирование. Примеры абстракций.</li> <li>2. Инкапсуляция, примеры на типизированных и нетипизированных языках.</li> <li>3. Иерархия, примеры на типизированных и нетипизированных языках.</li> <li>4. Типизация, примеры полиморфизма.</li> <li>5. Сохраняемость. Формат JSON, другие возможные форматы для сериализации</li> </ol> <p><i>Практические задания</i></p> <ol style="list-style-type: none"> <li>1. Реализовать на C++ иерархию классов «Шахматные фигуры». Создать класс «Позиция».</li> <li>2. Реализовать на JavaScript иерархию объектов «Водный транспорт». Учесть тип двигателя.</li> <li>3. Реализовать на C++ иерархию классов «Водный транспорт». Учесть тип двигателя</li> <li>4. Реализовать на C++ иерархию классов «Воздушный транспорт». Учесть тип двигателя.</li> <li>5. Реализовать на JavaScript иерархию объектов «Воздушный транспорт».</li> </ol>

Код индикатора	Индикатор достижения компетенции	Оценочные средства
		<p>Учесть тип двигателя.</p> <p>6. Реализовать на JavaScript иерархию объектов «Наземный транспорт». Учесть тип двигателя.</p> <p>7. Реализовать на C++ иерархию классов «Наземный транспорт». Учесть тип двигателя</p>

Код индикатора	Индикатор достижения компетенции	Оценочные средства
ПК-6.2	Оценивает качество разработанных алгоритмов для последующего кодирования	<p><i>Перечень теоретических вопросов</i></p> <ol style="list-style-type: none"> <li>1. Абстрагирование. Примеры абстракций.</li> <li>2. Инкапсуляция, примеры на типизированных и нетипизированных языках.</li> <li>3. Иерархия, примеры на типизированных и нетипизированных языках.</li> <li>4. Типизация, примеры полиморфизма.</li> <li>5. Сохраняемость. Формат JSON, другие возможные форматы для сериализации</li> </ol> <p><i>Практические задания</i></p> <ol style="list-style-type: none"> <li>1. Реализовать на С++ иерархию классов «Шахматные фигуры». Создать класс «Позиция».</li> <li>2. Реализовать на JavaScript иерархию объектов «Водный транспорт». Учесть тип двигателя.</li> <li>3. Реализовать на С++ иерархию классов «Водный транспорт». Учесть тип двигателя</li> <li>4. Реализовать на С++ иерархию классов «Воздушный транспорт». Учесть тип двигателя.</li> <li>5. Реализовать на JavaScript иерархию объектов «Воздушный транспорт». Учесть тип двигателя.</li> <li>6. Реализовать на JavaScript иерархию объектов «Наземный транспорт». Учесть тип двигателя.</li> <li>7. Реализовать на С++ иерархию классов «Наземный транспорт». Учесть тип двигателя</li> </ol>

Код индикатора	Индикатор достижения компетенции	Оценочные средства
ПК-6.3	Оценивает выбор программных средств для программирования и манипулирования данными в соответствии установленными требованиями	<p><i>Перечень теоретических вопросов</i></p> <ol style="list-style-type: none"> <li>1. Абстрагирование. Примеры абстракций.</li> <li>2. Инкапсуляция, примеры на типизированных и нетипизированных языках.</li> <li>3. Иерархия, примеры на типизированных и нетипизированных языках.</li> <li>4. Типизация, примеры полиморфизма.</li> <li>5. Сохраняемость. Формат JSON, другие возможные форматы для сериализации</li> </ol> <p><i>Практические задания</i></p> <ol style="list-style-type: none"> <li>1. Реализовать на С++ иерархию классов «Шахматные фигуры». Создать класс «Позиция».</li> <li>2. Реализовать на JavaScript иерархию объектов «Водный транспорт». Учесть тип двигателя.</li> <li>3. Реализовать на С++ иерархию классов «Водный транспорт». Учесть тип двигателя</li> <li>4. Реализовать на С++ иерархию классов «Воздушный транспорт». Учесть тип двигателя.</li> <li>5. Реализовать на JavaScript иерархию объектов «Воздушный транспорт». Учесть тип двигателя.</li> <li>6. Реализовать на JavaScript иерархию объектов «Наземный транспорт». Учесть тип двигателя.</li> <li>7. Реализовать на С++ иерархию классов «Наземный транспорт». Учесть тип двигателя</li> </ol>

Код индикатора	Индикатор достижения компетенции	Оценочные средства
<b>ПК-2: Способность к анализу проблемной ситуации, разработке требований к системе, постановке целей создания системы, разработке концепции системы и технического задания на создание системы, представления концепции, технического задания на систему и изменений в них заинтересованным лицам</b>		
ПК-2.1	Оценивает выбор средств и методов для проведения системного анализа при проектировании программного обеспечения для автоматизированных систем	<p><i>Перечень теоретических вопросов</i></p> <ol style="list-style-type: none"> <li>1. Внешние и внутренние факторы качества ПО.</li> <li>2. Процесс OOD. Микро и макропроцесса проектирования</li> </ol> <p><i>Практические задания</i></p> <ol style="list-style-type: none"> <li>1. Реализовать на С++ иерархию классов «Шахматные фигуры». Создать класс «Позиция».</li> <li>2. Реализовать на JavaScript иерархию объектов «Водный транспорт». Учесть тип двигателя.</li> <li>3. Реализовать на С++ иерархию классов «Водный транспорт». Учесть тип двигателя</li> <li>4. Реализовать на С++ иерархию классов «Воздушный транспорт». Учесть тип двигателя.</li> <li>5. Реализовать на JavaScript иерархию объектов «Воздушный транспорт». Учесть тип двигателя.</li> <li>6. Реализовать на JavaScript иерархию объектов «Наземный транспорт». Учесть тип двигателя.</li> <li>7. Реализовать на С++ иерархию классов «Наземный транспорт». Учесть тип двигателя</li> </ol>

## **б) Порядок проведения промежуточной аттестации, показатели и критерии оценивания:**

Промежуточная аттестация по дисциплине «Объектно-ориентированное программирование» включает теоретические вопросы, позволяющие оценить уровень усвоения обучающимися знаний, и практические задания, выявляющие степень сформированности умений и владений, проводится в форме экзамена.

Экзамен по данной дисциплине проводится в устной форме по экзаменационным билетам, каждый из которых включает 1 теоретический вопрос и одно практическое задание.

### **Показатели и критерии оценивания экзамена:**

- на оценку **«отлично»** (5 баллов) – обучающийся демонстрирует высокий уровень сформированности компетенций, всестороннее, систематическое и глубокое знание учебного материала, свободно выполняет практические задания, свободно оперирует знаниями, умениями, применяет их в ситуациях повышенной сложности.
- на оценку **«хорошо»** (4 балла) – обучающийся демонстрирует средний уровень сформированности компетенций: основные знания, умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
- на оценку **«удовлетворительно»** (3 балла) – обучающийся демонстрирует пороговый уровень сформированности компетенций: в ходе контрольных мероприятий допускаются ошибки, проявляется отсутствие отдельных знаний, умений, навыков, обучающийся испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.
- на оценку **«неудовлетворительно»** (2 балла) – обучающийся демонстрирует знания не более 20% теоретического материала, допускает существенные ошибки, не может показать интеллектуальные навыки решения простых задач.
- на оценку **«неудовлетворительно»** (1 балл) – обучающийся не может показать знания на уровне воспроизведения и объяснения информации, не может показать интеллектуальные навыки решения простых задач.