

Министерство науки и высшего образования Российской Федерации
Муромский институт (филиал)
федерального государственного бюджетного образовательного учреждения высшего образования
**«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»**
(МИ ВлГУ)

Кафедра *ПИИ*

«УТВЕРЖДАЮ»
Заместитель директора по УР
Д.Е. Андрианов
_____ 25.05.2021

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Объектно-ориентированное программирование

Направление подготовки

09.03.04 Программная инженерия

Профиль подготовки

Программная инженерия

Семестр	Трудоем- кость, час./зач. ед.	Лек- ции, час.	Практи- ческие занятия, час.	Лабора- торные работы, час.	Консуль- тация, час.	Конт- роль, час.	Всего (контак- тная работа), час.	СРС, час.	Форма промежу- точного контроля (экз., зач., зач. с оц.)
2	252 / 7	32		36	5,2	0,35	73,55	151,8	Экз.(26,65)
3	252 / 7	28		40	4,8	0,35	73,15	152,2	Экз.(26,65)
Итого	504 / 14	60		76	10	0,7	146,7	304	53,3

Муром, 2021 г.

1. Цель освоения дисциплины

Цель дисциплины: приобретение студентами знаний об основных понятиях объектно-ориентированного программирования, изучение подходов программирования, проектирования и реализации изолированных классов, принципов объектно-ориентированного программирования, освоение принципов абстрагирования при выделении классов объектно-ориентированной программной системы, обзор современных технологий разработки объектно-ориентированных приложений.

Задачи дисциплины: освоение основ теории объектно-ориентированного программирования; освоение основных методов и технологий объектно-ориентированного программирования; получение студентами практических навыков в разработке на языке C#.

2. Место дисциплины в структуре ОПОП ВО

Дисциплина «Объектно-ориентированное программирование» базируется на знаниях, полученных студентами по курсу «Основы алгоритмизации и программирования», «Архитектура вычислительных систем». Углубление и расширение вопросов, изложенных в данном курсе, будет осуществляться во время работы студентов над дисциплинами: "Организация баз данных", а также при написании бакалаврских работ.

3. Планируемые результаты обучения по дисциплине

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции	Результаты обучения по дисциплине	
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов;	ОПК-6.1 Использует современные языки, утилиты и среды программирования	Знает современные языки программирования (ОПК-6.1) Умеет использовать современные языки программирования для решения практических задач (ОПК-6.1) Имеет навыки использования современных технологий разработки ПО (ОПК-6.1)	тест, задание на лабораторную работу

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 14 зачетных единиц, 504 часа.

4.1. Форма обучения: очная

Уровень базового образования: среднее общее.

Срок обучения 4г.

4.1.1. Структура дисциплины

№ п/п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником						Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)	
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация			Контроль
1	Основы программирования на языке C#	2	14		16					60	тестирование, отчет по лабораторным работам
2	Введение в ООП	2	18		20					91,8	тестирование, отчет по лабораторным работам
Всего за семестр		252	32		36			5,2	0,35	151,8	Экз.(26,65)
3	Стандартные классы .NET	3	22		16					0,2	тестирование, отчет по лабораторным работам
4	Шаблоны проектирования	3	6		24					152	тестирование, отчет по лабораторным работам
Всего за семестр		252	28		40			4,8	0,35	152,2	Экз.(26,65)
Итого		504	60		76			10	0,7	304	53,3

4.1.2. Содержание дисциплины

4.1.2.1. Перечень лекций

Семестр 2

Раздел 1. Основы программирования на языке C#

Лекция 1.

Введение в платформу .NET (2 часа).

Лекция 2.

Основные понятия языка C# (2 часа).

Лекция 3.

Линейные программы в языке C# (2 часа).

Лекция 4.

Управляющие операторы в языке C# (2 часа).

Лекция 5.

Массивы в языке C# (2 часа).

Лекция 6.

Строки в языке C# (2 часа).

Лекция 7.

Введение в технологию Windows Forms (2 часа).

Раздел 2. Введение в ООП

Лекция 8.

Классы и объекты в языке C# (2 часа).

Лекция 9.

Данные классов. Методы. Конструкторы. Свойства (2 часа).

Лекция 10.

Статические классы. Абстрактные классы (2 часа).

Лекция 11.

Полиморфизм конструкторов, методов (2 часа).

Лекция 12.

Наследование (2 часа).

Лекция 13.

Виртуальные методы. Позднее связывание (2 часа).

Лекция 14.

Класс object. Преобразования типов (2 часа).

Лекция 15.

Перегрузка операций в C# (2 часа).

Лекция 16.

Преобразование типов в C# (2 часа).

Семестр 3

Раздел 3. Стандартные классы .NET

Лекция 17.

Класс Array, ArrayList (2 часа).

Лекция 18.

Структуры. Записи. Кортежи (2 часа).

Лекция 19.

Классы работы с файловой системой (2 часа).

Лекция 20.

Обобщения. Обобщенные коллекции. Методы расширения (2 часа).

Лекция 21.

Интерфейсы (2 часа).

Лекция 22.

Стандартные интерфейсы в .NET (2 часа).

Лекция 23.

Делегаты. события (2 часа).

Лекция 24.

Делегаты. Анонимные методы. Лямбда выражения (2 часа).

Лекция 25.

LINQ (2 часа).

Лекция 26.

Обработка исключений. Класс Exception (2 часа).

Лекция 27.

Работа с XML. Сериализация и десериализация (2 часа).

Раздел 4. Шаблоны проектирования

Лекция 28.

Структурные шаблоны: Адаптер, Декоратор, Фасад, Заместитель (2 часа).

Лекция 29.

Поведенческие шаблоны: Наблюдатель, Стратегия, Состояние, Итератор, Шаблонный метод (2 часа).

Лекция 30.

Порождающие шаблоны: Абстрактная фабрика, Фабричный метод, Одиночка (2 часа).

4.1.2.2. Перечень практических занятий

Не планируется.

4.1.2.3. Перечень лабораторных работ

Семестр 2

Раздел 1. Основы программирования на языке C#

Лабораторная 1.

Линейные программы на языке C# (4 часа).

Лабораторная 2.

Операторы ветвления в языке C# (4 часа).

Лабораторная 3.

Массивы и строки в языке C# (4 часа).

Лабораторная 4.

Разработка приложений Windows Forms (4 часа).

Раздел 2. Введение в ООП

Лабораторная 5.

Введение в классы (4 часа).

Лабораторная 6.

Создание простейшего класса "структура-пара" (4 часа).

Лабораторная 7.

Полиморфизм методов класса (4 часа).

Лабораторная 8.

Наследование (4 часа).

Лабораторная 9.

Виртуальные методы классов (4 часа).

Семестр 3

Раздел 3. Стандартные классы .NET

Лабораторная 10.

Делегаты и события в классах (4 часа).

Лабораторная 11.

Интерфейсы (4 часа).

Лабораторная 12.

Реализация стандартных интерфейсов .NET (4 часа).

Лабораторная 13.

Использование LINQ для обработки коллекций (4 часа).

Раздел 4. Шаблоны проектирования

Лабораторная 14.

Шаблон проектирования Декоратор (4 часа).

Лабораторная 15.

Шаблон проектирования Адаптер (4 часа).

Лабораторная 16.

Шаблон проектирования Наблюдатель (4 часа).

Лабораторная 17.

Шаблон проектирования Стратегия (4 часа).

Лабораторная 18.

Шаблон проектирования Состояние (4 часа).

Лабораторная 19.

Шаблон проектирования Фабрика (4 часа).

4.1.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы

Перечень тем, вынесенных на самостоятельное изучение:

1. Понятие объекта и класса.
2. Основные принципы ООП.
3. Абстрагирование.
4. Инкапсуляция.
5. Агрегирование.
6. Объектная модель программы.
7. Универсальный язык моделирования UML.
8. Метаданные.
9. Промежуточный код (Intermediate Language).
10. Единая среда выполнения (Common Language Runtime).
11. Пространства имен (namespaces).
12. Единая библиотека типов (классов, интерфейсов, структур) платформы - Microsoft Framework Library, основные пространства имен.
13. Встроенный язык поисковых запросов LINQ.
14. Конструкторы класса.
15. Свойства класса.
16. Автоматически реализуемые свойства. Инициализация объектов класса.
17. Индексаторы. Статические поля и методы класса.
18. Переопределение операций класса. Определение преобразования типов.
19. Отношение вложенности. Отношение наследования.
20. Описание производных классов. Конструкторы производного класса.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

4.1.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР

Не планируется.

4.1.2.6. Примерный перечень тем курсовых работ (проектов)

Не планируется.

4.2 Форма обучения: заочная

Уровень базового образования: среднее общее.

Срок обучения 5л.

Семестр	Трудоемкость, час./ зач. ед.	Лекции, час.	Практические занятия, час.	Лабораторные работы, час.	Консультация, час.	Контроль, час.	Всего (контактная работа), час.	СРС, час.	Форма промежуточного контроля (экзамен, зачет, зачет с оценкой)
3	252 / 7	10	4	16	5	0,6	35,6	207,75	Экз.(8,65)
4	252 / 7	8		16	4	0,6	28,6	214,75	Экз.(8,65)
Итого	504 / 14	18	4	32	9	1,2	64,2	422,5	17,3

4.2.1. Структура дисциплины

№ п/п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Основы программирования на языке C#	3	4		4					84	тестирование, отчет по лабораторным работам
2	Введение в ООП	3	6	4	12					123,75	тестирование, отчет по лабораторным работам
Всего за семестр		252	10	4	16	+		5	0,6	207,75	Экз.(8,65)
3	Стандартные классы .NET	4	4		8					7,25	тестирование, отчет по лабораторным работам
4	Шаблоны проектирования	4	4		8					207,5	тестирование, отчет по лабораторным работам

Всего за семестр	252	8		16	+		4	0,6	214,75	Экз.(8,65)
Итого	504	18	4	32			9	1,2	422,5	17,3

4.2.2. Содержание дисциплины

4.2.2.1. Перечень лекций

Семестр 3

Раздел 1. Основы программирования на языке С#

Лекция 1.

Классы в С#. Управление доступом (2 часа).

Лекция 2.

Конструкторы. Деструкторы. Наследование (2 часа).

Раздел 2. Введение в ООП

Лекция 3.

Свойства классов в С# (2 часа).

Лекция 4.

Скалярные и индексированные свойства (2 часа).

Лекция 5.

Перегрузка операций в С# (2 часа).

Семестр 4

Раздел 3. Стандартные классы .NET

Лекция 6.

Преобразование типов в С# (2 часа).

Лекция 7.

Делегаты. события. Интерфейсы (2 часа).

Раздел 4. Шаблоны проектирования

Лекция 8.

Концепция объектно-ориентированного проектирования. Проектирование структурной схемы программы (2 часа).

Лекция 9.

Реорганизация иерархии и структуры классов. Организационная структура программы (2 часа).

4.2.2.2. Перечень практических занятий

Семестр 3

Раздел 2. Введение в ООП

Практическое занятие 1.

Концепции программирования под Windows. Структура Windows программ (2 часа).

Практическое занятие 2.

Использование Windows Forms для создания приложений с графическим интерфейсом пользователя (2 часа).

4.2.2.3. Перечень лабораторных работ

Семестр 3

Раздел 1. Основы программирования на языке С#

Лабораторная 1.

Структура программы. Общая характеристика языка С# (4 часа).

Раздел 2. Введение в ООП

Лабораторная 2.

Объектно-ориентированная методология программирования (4 часа).

Лабораторная 3.

Перегрузка операций. Преобразование типов (4 часа).

Лабораторная 4.

Наследование, базовый и производный классы. Простое и сложное наследование (4 часа).

Семестр 4

Раздел 3. Стандартные классы .NET

Лабораторная 5.

Виртуальные функции. Дружественные функции (4 часа).

Лабораторная 6.

Дружественные классы (4 часа).

Раздел 4. Шаблоны проектирования

Лабораторная 7.

Стандартная библиотека шаблонов (4 часа).

Лабораторная 8.

Шаблон проектирования Декоратор (4 часа).

4.2.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы

Перечень тем, вынесенных на самостоятельное изучение:

1. Понятие объекта и класса.
2. Основные принципы ООП.
3. Абстрагирование.
4. Инкапсуляция.
5. Агрегирование.
6. Объектная модель программы.
7. Универсальный язык моделирования UML.
8. Метаданные.
9. Промежуточный код (Intermediate Language).
10. Единая среда выполнения (Common Language Runtime).
11. Пространства имен (namespaces).
12. Единая библиотека типов (классов, интерфейсов, структур) платформы - Microsoft Framework Library, основные пространства имен.
13. Встроенный язык поисковых запросов LINQ.
14. Конструкторы класса.
15. Свойства класса.
16. втоматически реализуемые свойства. Инициализация объектов класса.
17. Индексаторы. Статические поля и методы класса.
18. Переопределение операций класса. Определение преобразования типов.
19. Отношение вложенности. Отношение наследования.
20. Описание производных классов. Конструкторы производного класса.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

4.2.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР

1. Указатели. Динамические переменные.
2. Процедуры и функции для работы с динамической памятью.
3. Динамические структуры данных (списки, стек, очередь).
4. Объекты. Классы.
5. Иерархия классов. Объявление классов.
6. Методы ? элементы класса (описание, вызов).
7. Перекрытие методов. Порядок вызова методов.
8. Основные принципы ООП: инкапсуляция, наследование (общая форма наследования), полиморфизм.
9. Ключевое слово this.
10. Режимы доступа к элементам класса.

11. Режимы доступа при наследовании классов.
12. Раннее и позднее связывание. Статические методы.
13. Виртуальные методы. Таблица виртуальных методов.
14. Применение виртуальных функций, связанное с использованием указателей на объекты.
15. Совместимость типов (между объектами и указателями на объект, между формальными и фактическими параметрами).
16. Конструкторы и деструкторы. Конструкторы по умолчанию. Списки инициализации. Виртуальные деструкторы.
17. Конструкторы с параметрами. Конструкторы с параметрами при наследовании.
18. Массивы объектов.
19. Указатель на объект. Операции динамического выделения памяти.
20. Чистые (pure) виртуальные функции. Абстрактные классы.
21. Статические (static) элементы класса.
22. Перегруженные функции.
23. Перегрузка операций.
24. Дружественные функции. Дружественные классы.
25. Аргументы функции (в т.ч. и конструктора), задаваемые по умолчанию.
26. Параметры и аргументы функции. Передача объектов как аргументов функций. Использование объекта в качестве возвращаемого значения.
27. Указатели и ссылки на объекты.
28. Шаблоны классов и функций.
29. Локальные классы.
30. Подставляемые (InLine) функции.

4.2.2.6. Примерный перечень тем курсовых работ (проектов)

Не планируется.

5. Образовательные технологии

При проведении лекционных занятий материал дается в устной форме с применением мультимедиа-проектора. На лабораторных занятиях каждый студент получает индивидуальное задание. Все задания носят практический характер и предполагают написание программного кода на языке высокого уровня. Активно применяются информационно-коммуникационные технологии.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

Фонды оценочных материалов (средств) приведены в приложении.

7. Учебно-методическое и информационное обеспечение дисциплины.

7.1. Основная учебно-методическая литература по дисциплине

1. Зыков, С. В. Введение в теорию программирования. Объектно-ориентированный подход : учебное пособие — Москва : ИНТУИТ, Ай Пи Ар Медиа, 2021. — 187 с. — URL: - <https://www.iprbookshop.ru/102007.html>
2. Зайцев, М. Г. Объектно-ориентированный анализ и программирование : учебное пособие — Новосибирск : НГТУ, 2017. — 84 с. — URL: - <https://www.iprbookshop.ru/91284.html>

7.2. Дополнительная учебно-методическая литература по дисциплине

1. Ланских, Ю. В. Основы объектно-ориентированного и компонентно-ориентированного программирования в C# / Ю. В. Ланских, Л. В. Пешнина. — Соликамск :

Соликамский государственный педагогический институт, 2017. — 84 с. — URL: - <https://www.iprbookshop.ru/86557.html>

2. Маляров, А. Н. Объектно-ориентированное программирование : учебник / А. Н. Маляров. — Саратов : Профобразование, 2021. — 331 с. URL: - <https://www.iprbookshop.ru/106837.html>

7.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

В образовательном процессе используются информационные технологии, реализованные на основе информационно-образовательного портала института (www.mivlgu.ru/iop), и инфокоммуникационной сети института:

- предоставление учебно-методических материалов в электронном виде;
- взаимодействие участников образовательного процесса через локальную сеть института и Интернет;
- предоставление сведений о результатах учебной деятельности в электронном личном кабинете обучающегося.

Информационные справочные системы:

Информационно-образовательный портал МИ ВлГУ - <https://www.mivlgu.ru/iop/>

Электронная библиотечная система iprBooks.ru - <http://www.iprbooks.ru>

Электронная библиотека «ЭВРИКА» - <https://evrika.mivlgu.ru/>

Электронная библиотека ВлГУ - <https://dspace.www1.vlsu.ru/>

Библиотека MSDN: <http://msdn.microsoft.com>

Программное обеспечение:

Microsoft Visual Studio (Программа Microsoft Azure Dev Tools for Teaching (Order Number: IM126433))

7.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

iprbookshop.ru
mivlgu.ru
iprbooks.ru
msdn.microsoft.com
mivlgu.ru/iop

8. Материально-техническое обеспечение дисциплины

Лаборатория программного обеспечения и сопровождения компьютерных систем

Сервер «Ай Тек» на базе 2 процессоров Intel Xeon; 12 шт. компьютеров Intel Core i5-10400 2,9 GHz/ 8Gb DDR-4/ SSD-480 Gb/ Hiper 21,5'; интерактивная доска SMART Board 480 со встроенным проектором V25; маршрутизатор Gigabit Switch TEG-S16S. Маркерная доска. Доступ к сети Интернет.

9. Методические указания по освоению дисциплины

Для успешного освоения теоретического материала обучающийся: знакомится со списком рекомендуемой основной и дополнительной литературы; уточняет у преподавателя, каким дополнительным пособиям следует отдать предпочтение; ведет конспект лекций и прорабатывает лекционный материал, пользуясь как конспектом, так и учебными пособиями.

До выполнения лабораторных работ обучающийся изучает соответствующий раздел теории. Перед занятием студент знакомится с описанием заданий для выполнения работы, внимательно изучает содержание и порядок проведения лабораторной работы. Лабораторная работа проводится в компьютерном классе. Обучающиеся выполняют индивидуальную задачу компьютерного моделирования в соответствии с заданием на лабораторную работу.

Полученные результаты исследований сводятся в отчет и защищаются по традиционной методике в классе на следующем лабораторном занятии. Необходимый теоретический материал, индивидуальное задание, шаги выполнения лабораторной работы и требование к отчету приведены в методических указаниях, размещенных на информационно-образовательном портале института.

Самостоятельная работа оказывает важное влияние на формирование личности будущего специалиста, она планируется обучающимся самостоятельно. Каждый обучающийся самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием дисциплины. Он выполняет внеаудиторную работу и изучение разделов, выносимых на самостоятельную работу, по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Форма заключительного контроля при промежуточной аттестации – экзамен. Для проведения промежуточной аттестации по дисциплине разработаны фонд оценочных средств и балльно-рейтинговая система оценки учебной деятельности студентов. Оценка по дисциплине выставляется в информационной системе и носит интегрированный характер, учитывающий результаты оценивания участия студентов в аудиторных занятиях, качества и своевременности выполнения заданий в ходе изучения дисциплины и промежуточной аттестации.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению *09.03.04 Программная инженерия* и профилю подготовки *Программная инженерия*
Рабочую программу составил *Холкина Н.Е.*_____

Программа рассмотрена и одобрена на заседании кафедры *ПИИ*

протокол № 13 от 14.05.2021 года.

Заведующий кафедрой *ПИИ* _____ *Жизняков А.Л.*
(Подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии факультета

протокол № 9 от 24.05.2021 года.

Председатель комиссии ФИТР _____ *Рыжкова М.Н.*
(Подпись) (Ф.И.О.)

**Фонд оценочных материалов (средств) по дисциплине
Объектно-ориентированное программирование**

1. Оценочные материалы для проведения текущего контроля успеваемости по дисциплине

Примеры заданий для выполнения лабораторных работ

Разработать класс "стек" и использовать его при решении задач.

Разработать класс дробь. На ее основе создать потомков - обыкновенную и десятичную дробь.

Общее распределение баллов текущего контроля по видам учебных работ для студентов

Рейтинг-контроль 1	защита отчетов по лабораторным работам	до 15 баллов
Рейтинг-контроль 2	защита отчетов по лабораторным работам	до 15 баллов
Рейтинг-контроль 3	защита отчетов по лабораторным работам	до 15 баллов
Посещение занятий студентом	Отметка в журнале посещений	до 5 баллов
Дополнительные баллы (бонусы)	Активность на занятии	до 5 баллов
Выполнение семестрового плана самостоятельной работы		до 5 баллов

2. Промежуточная аттестация по дисциплине

Перечень вопросов к экзамену / зачету / зачету с оценкой.

Перечень практических задач / заданий к экзамену / зачету / зачету с оценкой (при наличии)

Оценочные средства для промежуточной аттестации в виде итогового теста приведены на информационно-образовательном портале по ссылке
<https://www.mivlgu.ru/iop/course/view.php?id=381>

Методические материалы, характеризующие процедуры оценивания

На основе перечня вопросов формируются индивидуальные задания для студентов: 4 вопроса из блока 1, 3 вопроса из блока 2, 3 вопроса из блока 3. Результатом итогового контрольного теста является балл, рассчитанный на основе количества правильных ответов. С учетом индивидуального семестрового рейтинга студента формируется итоговый балл по курсу.

Максимальная сумма баллов, набираемая студентом по дисциплине равна 100.

Оценка в баллах	Оценка по шкале	Обоснование	Уровень сформированности компетенций
Более 80	«Отлично»	Содержание курса освоено полностью, без пробелов, необходимые практические навыки	Высокий уровень

		работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	
66-80	«Хорошо»	Содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	<i>Продвинутый уровень</i>
50-65	«Удовлетворительно»	Содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки	<i>Пороговый уровень</i>
Менее 50	«Неудовлетворительно»	Содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки	<i>Компетенции не сформированы</i>

3. Задания в тестовой форме по дисциплине

Примеры заданий:

Объект, принадлежащий множеству объектов, описываемых спецификацией абстрактного типа данных называется:

- A. экземпляром этого абстрактного типа данных
- B. потомком этого абстрактного типа данных
- C. прототипом этого абстрактного типа данных
- D. функцией этого абстрактного типа данных

Механизм, делающий определенные компоненты класса недоступными для клиентов:

- A. Шифрование информации.
- B. Скрытие информации.
- C. Авторизация доступа.
- D. Удаление мусора в памяти.

Спецификация абстрактного типа данных является _____ тогда и только тогда, когда для всякого правильно построенного выражения ее аксиомы позволяют вывести не более одного значения. (непротиворечивой)

Полный перечень тестовых заданий с указанием правильных ответов, размещен в банке вопросов на информационно-образовательном портале института по ссылке <https://www.mivlgu.ru/iop/question/edit.php?courseid=381&deleteall=1&category=54221%2C9245&qbshowtext=1&recurse=0&recurse=1&showhidden=0>

Оценка рассчитывается как процент правильно выполненных тестовых заданий из их общего числа.