

Министерство науки и высшего образования
Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Луганский государственный университет
имени Владимира Даля»

Стахановский инженерно-педагогический институт (филиал)
федерального государственного бюджетного образовательного
учреждения высшего образования
«Луганский государственный университет имени Владимира Даля»

Кафедра информационных систем

УТВЕРЖДАЮ:
Директор СИПИ (филиала)
ФГБОУ ВО «ЛГУ им. В. Даля»
А.А. Авершин
«21 апреля» 2023 года

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

**«ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ И
ТЕХНОЛОГИИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ»**

По направлению подготовки 44.03.04 Профессиональное обучение (по
отраслям)

Профиль: «Информационные технологии и системы»

Лист согласования РПУД

Рабочая программа учебной дисциплины «Объектно-ориентированное программирование и технологии разработки программного обеспечения» по направлению подготовки 44.03.04 Профессиональное обучение (по отраслям). – 27 с.

Рабочая программа учебной дисциплины «Объектно-ориентированное программирование и технологии разработки программного обеспечения» разработана в соответствии с Федеральным государственным образовательным стандартом высшего образования по направлению подготовки 44.03.04 Профессиональное обучение (по отраслям), утвержденным приказом Министерства образования и науки Российской Федерации от 22 февраля 2018 г. № 124 (с изменениями и дополнениями от 26 ноября 2020 г., 27 февраля 2023 г.)

СОСТАВИТЕЛЬ:

канд. техн. наук, доцент Карчевский В.П.

Рабочая программа дисциплины утверждена на заседании кафедры информационных систем «18» апреля 2023 г., протокол № 9.

Заведующий кафедрой информационных систем  В.П. Карчевский

Переутверждена: « » 20 г., протокол № .

Переутверждена: « » 20 г., протокол № .

Рекомендована на заседании учебно-методической комиссии Стахановского инженерно-педагогического института (филиала) федерального государственного бюджетного образовательного учреждения высшего образования «Луганский государственный университет имени Владимира Даля» 21 апреля 2023 г., протокол № 3.

Председатель учебно-методической комиссии СИПИ (филиала) ФГБОУ ВО «ЛГУ им. В. Даля»  Н.В. Банник

© Карчевский В.П., 2023 год

© ФГБОУ ВО «ЛГУ им. В. Даля», 2023 год

Структура и содержание дисциплины

1. Цели и задачи дисциплины, ее место в учебном процессе

Цель изучения дисциплины – изучение современного подхода к программированию в объектах, приобретение навыков написания программ на объектно–ориентированных языках, знакомства с методами разработки, тестирования, отладки, анализа, обеспечения безопасности и надежности программ.

Задачи:

совершенствование навыков визуального программирования в объектно-ориентированной среде разработки;

умение разрабатывать алгоритмы решения задач, знания важных приемов и методов создания классов;

умение применять объектно-ориентированные языки программирования для решения задач предметной области;

умение составления прикладных пакетов программ, приобретение практических навыков в объектно-ориентированном программировании;

изучение основ объектно-ориентированного проектирования и анализа;

изучение принципов автоматизации программирования;

овладение эффективными навыками разработки и тестирования проектов;

получение навыков индивидуальной самостоятельной разработки проектов приложений для решения математических, графических задач.

2. Место дисциплины в структуре ООП ВО

Дисциплина «Объектно-ориентированное программирование и технологии разработки программного обеспечения» относится к части, формируемой участниками образовательных отношений. Необходимыми условиями для освоения дисциплины являются: знания этапов проектирования и разработки модели, источники получения информации, необходимой для решения поставленной задачи; умения излагать мысли, находить ответы на вопросы анализировать рабочий процесс; использовать полученные знания при практической работе на ПК; навыки к восприятию информации; постановки цели и выбора путей ее достижения; использования различных способов поиска (в справочных источниках и открытом учебном информационном пространстве сети Интернет), сбора, обработки, анализа, организации, передачи и интерпретации информации в соответствии с коммуникативными и познавательными задачами и технологиями учебного предмета;

Содержание дисциплины является логическим продолжением содержания дисциплин: «Архитектура компьютеров и информационных систем», «История развития техники. Техническое и методическое творчество», «Информатика и информационные технологии», «Программное обеспечение систем управления и обучения», «Автоматизированные системы организационного управления», «Ремонт и модернизация персональных компьютеров», «Программная инженерия», «Управление информацией и интеллектуальные системы» и служит основой для освоения дисциплин: «Web-

дизайн и программирование», «Методы и средства защиты компьютерной информации», «Компьютерные технологии в учебном процессе».

3. Требования к результатам освоения содержания дисциплины

Код и наименование компетенции	Индикаторы достижений компетенции (по реализуемой дисциплине)	Перечень планируемых результатов
УК-6. Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни	УК-6.1. Определяет свои личные ресурсы, возможности и ограничения для достижения поставленной цели	Знать: принципы применения компьютерных технологий в учебном процессе; основы объектно-ориентированного проектирования и анализа.
	УК-6.2. Создает и достраивает индивидуальную траекторию саморазвития при получении основного и дополнительного образования	Уметь: анализировать и осуществлять объектно-ориентированную декомпозицию и объектно-ориентированный анализ предметной области, используя иерархию классов, диаграммы процессов и объектов.
	УК-6.3. Владеет умением рационального распределения временных и информационных ресурсов	Владеть: навыками создания программных кодов, модулей и проектов для решения типовых задач обработки данных; навыками отладки программ с использованием систем программирования на компьютере; навыками использования современных интегрированных сред разработчика приложений и программных систем; индивидуальной самостоятельной разработки проектов приложений для решения математических, графических задач.
ОПК-1. Способен осуществлять профессиональную деятельность в соответствии с нормативными правовыми актами в сфере образования и нормами профессиональной этики	ОПК-1.1. Демонстрирует знания нормативно-правовых актов в сфере образования и норм профессиональной этики	Знать: особенности алгоритмизации и программирование вычислительных процессов на языках программирования высокого уровня.
	ОПК-1.2. Строит образовательные отношения в соответствии с правовыми и этическими нормами профессиональной деятельности	Уметь: создавать программные коды, модули и проекты решения производственных задач, отлаживать программы с использованием систем программирования на компьютере.
	ОПК-1.3. Организует образовательную среду в соответствии с правовыми и этическими нормами профессиональной деятельности	Владеть: навыками участия в разработке программного обеспечения автоматизированных систем, используя технологии и языки программирования, работая в коллективе разработчиков, на основании технического задания на разработку системы и данных обследования объекта
	ОПК-1.4. Выстраивает образовательный процесс в соответствии с правовыми и этическими нормами	

	профессиональной деятельности	автоматизации.
ПК-2. Способен выполнять работы по созданию, модификации и сопровождению информационных систем	ПК-2.1. – Осуществляет создание пользовательской документации к модифицированным элементам типовой ИС	Знать: основные принципы объектно-ориентированного программирования; топологии языков программирования; объектно-ориентированные языки программирования.
	ПК-2.2. – Осуществляет техническое обеспечение процесса обучения пользователей ИС	Уметь: применять объектно-ориентированные языки программирования для решения задач предметной области.
	ПК-2.3. – Выполняет установку и настройку системного и прикладного ПО, необходимого для функционирования ИС ПК-2.4. – Настраивает оборудование, необходимое для работы ИС ПК-2.5. – Выполняет кодирование на языках программирования ПК-2.6. – Создает пользовательскую документацию к ИС ПК-2.7. – Обучает пользователей ИС	Владеть: навыками работы с компьютером для решения практических и исследовательских задач; навыками разработки и тестирования проектов.

4. Структура и содержание дисциплины

4.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов (зач. ед.)		
	Очная форма	Очно-заочная форма	Заочная форма
Объем учебной дисциплины (всего)	144 (4 зач. ед.)	-	144 (4 зач. ед.)
Обязательная контактная работа (всего)	96	-	20
в том числе:			
Лекции	36	-	8
Семинарские занятия	-	-	-
Практические занятия	12	-	4
Лабораторные работы	48	-	8
Курсовая работа (курсовой проект)	-	-	-
Другие формы и методы организации образовательного процесса (<i>расчетно-графические работы, групповые дискуссии, ролевые игры, тренинг, компьютерные симуляции, интерактивные лекции, семинары, анализ деловых ситуаций и т.п.</i>)	-	-	-
Самостоятельная работа студента (всего)	48	-	124
Итоговая аттестация	Экзамен	-	Экзамен

4.2. Содержание разделов дисциплины

Тема 1. Теоретические основы объектно-ориентированного программирования. История языков программирования от процедурного программирования к объектному.

История языков программирования. Языки программирования низкого и высокого уровней. Топологии языков программирования.

Тема 2. Визуализация и реализация проектов объектно-ориентированного программирования.

Использование диаграмма классов в ООП. Понятие классов, методов, объектов.

Тема 3. Особенности работы с ООП в Delphi. Листинг программ.

Особенности работы в ООП, создание объектов. Конструкторы и деструкторы. Модификация в объектно-ориентированном программировании. Оформление исходных листингов на Delphi.

Тема 4. Основные принципы объектно-ориентированного программирования. Наследование.

Основные принципы объектно-ориентированного программирования. Понятие наследования. Реализация наследования.

Тема 5. Основные принципы объектно-ориентированного программирования. Инкапсуляция.

Инкапсуляция. Классы и директивы в инкапсуляции. Реализация инкапсуляции посредством методов объектов.

Тема 6. Методы классов. Обработка сообщений.

Связывание методов. Введение в обработку сообщений. Обработчики сообщений.

Тема 7. Создание приложений Windows.

Семейство операционных систем Windows с точки зрения программиста. Особенности архитектуры программы в ОС Windows. Архитектура программы в ОС Windows. Программа, управляемая событиями. Взаимодействие программы с операционной системой.

Тема 8. Основные принципы объектно-ориентированного программирования. Полиморфизм. Виртуальные методы.

Полиморфизм. Виртуальные методы. Реализация полиморфизма. Виртуальные методы.

Тема 9. Полиморфизм: перегрузка и переопределение операторов.

Полиморфизм. Взаимодействие всех принципов ООП в Delphi: инкапсуляция, наследование и полиморфизм. Случаи использования полиморфизма.

Тема 10. Исключения в Delphi, классы исключений.

Исключения в Delphi, классы исключений. Процедуры, вызывающие исключения в Delphi.

Тема 11. Иерархия классов в объектно-ориентированном программировании.

Понятие иерархии классов и дерева наследования. Применение иерархии классов. Сферы использования иерархий в примерах.

Тема 12. Библиотека классов VCL среды проектирования Delphi.

Библиотека классов VCL Delphi. Отображения иерархических данных в Windows. Файлы проекта Delphi.

Тема 13. Объектно-ориентированное программирование в графическом режиме.

Поддержка графики в Windows. Вывод графической информации в Delphi. Классы и компоненты для работы с графикой.

Тема 14. Потoki и процессы в Delphi. Классы для представления потока данных. Сокет как программный интерфейс. Потoki и процессы.

Понятие о потоках и процессах. Приоритеты потоков. Классы для представления потока данных.

Тема 15. Коллекции в программировании.

Понятие коллекций в программировании. Классификация коллекций. Операции над коллекциями.

Тема 16. Основы com-технологий (Component object model).

Технология COM. Технология автоматизации. Динамические библиотеки.

Тема 17. Графические движки в программировании.

Понятие графического движка. История графических движков. Аппаратная абстракция.

Тема 18. Система Android. Введение в программирование мобильных устройств.

Введение в программирование мобильных устройств. История возникновения Android. Основные компоненты Android.

4.3. Лекции

№ темы	Название темы	Объем часов		
		Очная форма	Очно-заочная форма	Заочная форма
1	Теоретические основы объектно-ориентированного программирования. История языков программирования.	2	-	2
2	Визуализация и реализация проектов объектно-	2	-	-

	ориентированного программирования.			
3	Особенности работы с ООП в Delphi. Листинг программ.	2	-	-
4	Основные принципы объектно-ориентированного программирования. Наследование.	2	-	2
5	Основные принципы объектно-ориентированного программирования. Инкапсуляция.	2	-	2
6	Методы классов. Обработка сообщений.	4	-	-
7	Основные принципы объектно-ориентированного программирования. Полиморфизм.	2	-	2
8	Процедуры и функции вызова диалоговых окон.	2	-	-
9	Исключения в Delphi, классы исключений.	2	-	-
10	Иерархия классов в объектно-ориентированном программировании.	2	-	-
11	Библиотека классов VCL среды проектирования Delphi.	2	-	-
12	Объектно-ориентированное программирование в графическом режиме.	2	-	-
13	Потоки и процессы в Delphi..	2	-	-
14	Система управления версиями git.	2	-	-
15	Коллекции в программировании.	2	-	-
16	Основы com-технологий (Component object model).	2	-	-
17	Графические движки в программировании.	2	-	-
18	Система Андроид. Введение в программирование мобильных устройств.	2	-	-
Итого:		36	-	8

4.4. Практические (семинарские) занятия

№ темы	Название темы	Объем часов		
		Очная форма	Очно- заочная форма	Заочная форма
1.	Алгоритмы и структуры данных.	4	-	2
2.	Графические движки.	4	-	-
3.	Системы управления версиями Git.	4	-	2
Итого:		12	-	4

4.5. Лабораторные работы

№ п/п	Название темы	Объем часов		
		Очная форма	Очно- заочная форма	Заочная форма
1	Средства создания диаграмм классов в ООП.	2	-	2
2	Объектно-ориентированное программирование. Определение класса.	2	-	-
3	Реализация собственного класса.	2	-	2
4	Наследование классов.	4	-	-
5	Принципы объектно-ориентированного программирования. Инкапсуляция. Классы и объекты в Delphi.	2	-	-
6	Разработка приложения с использованием класса.	2	-	-
7	Принципы ООП. Наследование классов.	4	-	-
8	Принципы ООП. Полиморфизм в ООП.	4	-	-

	Виртуальные методы.			
9	Принципы ООП. Полиморфизм в ООП. Перегрузка операторов.	2	-	-
10	Разработка приложения, создающего и обрабатывающего собственный класс.	2	-	-
11	Создание иерархий классов.	4	-	2
12	Применение наследования и полиморфизма в графическом режиме.	4	-	-
13	Использование принципов ООП в графическом режиме. Центрические окружности.	2	-	-
14	Многопоточность. Работа с потоками в Delphi.	2	-	-
15	Реализация текстового редактора в Delphi.	2	-	-
16	Создание графического редактора изображений.	2	-	-
17	Создание пользовательского приложения для создания экранных копий.	2	-	-
18	Графические движки в Delphi.	4	-	2
Итого:		48	-	8

4.6. Самостоятельная работа студентов

№ п/п	Название темы	Вид СРС	Объем часов		
			Очная форма	Очно-заочная форма	Заочная форма
1	История языков программирования. Возникновение объектной декомпозиции.	Проработка конспекта лекций, проработка справочной и учебной литературы, подготовка ко входному контролю	1	-	8
2	Поля и свойства классов.	Проработка конспекта лекций, проработка справочной и учебной литературы, подготовка к лабораторным работам, подготовка к контрольному опросу	1	-	8
3	Методы, конструкторы и деструкторы классов.	Проработка конспекта лекций, проработка справочной и учебной литературы, подготовка к лабораторным работам	1	-	7
4	Процедурная и объектная декомпозиция предметной области задачи	Проработка конспекта лекций, проработка справочной и учебной литературы, подготовка к лабораторным и практическим работам	1	-	7
5	Понятие программного объекта. Операции над программными объектами	Проработка конспекта лекций, проработка справочной и учебной литературы, подготовка к лабораторным работам, подготовка к текущему контролю знаний	1	-	7
6	Основные этапы разработки программных систем	Проработка конспекта лекций, проработка справочной и учебной литературы, подготовка к лабораторным работам	1	-	7

	использованием ООП				
7	Обработка исключительных ситуаций.	Проработка конспекта лекций, проработка справочной и учебной литературы, подготовка к лабораторным и практическим работам	1	-	7
8	Структурные операторы, применяемые в Object Pascal, и их особенности	Проработка конспекта лекций, проработка справочной и учебной литературы, подготовка к лабораторным и практическим работам	1	-	8
9	Диалоговые компоненты Windows.	Проработка конспекта лекций, проработка справочной и учебной литературы, подготовка к лабораторным и практическим работам	1	-	7
10	Основы организации многооконных приложений в Delphi.	Проработка конспекта лекций, проработка справочной и учебной литературы, подготовка к лабораторным и практическим работам	1	-	7
11	Стандартные элементы управления Windows.	Проработка конспекта лекций, проработка справочной и учебной литературы, подготовка к лабораторным и практическим работам	1	-	7
12	Способы отображения иерархической информации в Delphi.	Проработка конспекта лекций, проработка справочной и учебной литературы, подготовка к лабораторным работам, подготовка к итоговому контролю знаний	1	-	8
13	Экзамен		36	-	36
	Итого:		48	-	124

4.7. Курсовые работы/проекты по дисциплине «Объектно-ориентированное программирование и технологии разработки программного обеспечения» не предусмотрены учебным планом.

5. Образовательные технологии

Преподавание дисциплины ведется с применением следующих видов образовательных технологий: объяснительно-иллюстративного обучения (технология поддерживающего обучения, технология проведения учебной дискуссии), информационных технологий (презентационные материалы), развивающих и инновационных образовательных технологий.

Практические занятия проводятся с использованием развивающих, проблемных, проектных, информационных (использование электронных образовательных ресурсов (электронный конспект) образовательных технологий.

6. Формы контроля освоения дисциплины

Текущая аттестация студентов производится в дискретные временные интервалы лектором и преподавателем (-ями), ведущими лабораторные и практические занятия по дисциплине в следующих формах: контрольные работы.

Промежуточная аттестации по результатам освоения дисциплины проходит в форме устного/письменного экзамена (включает в себя ответы на теоретические вопросы и ответы на задания). Студенты, выполнившие 75% текущих и контрольных мероприятий на «отлично», а остальные 25 % на «хорошо», имеют право на получение итоговой оценки.

В экзаменационную ведомость и зачетную книжку выставляются оценки по шкале, приведенной в таблице.

Шкала оценивания (экзамен)	Характеристика знания предмета и ответов	Зачеты
отлично (5)	Студент глубоко и в полном объеме владеет программным материалом. Грамотно, исчерпывающе и логично его излагает в устной или письменной форме. При этом знает рекомендованную литературу, проявляет творческий подход в ответах на вопросы и правильно обосновывает принятые решения, хорошо владеет умениями и навыками при выполнении практических задач.	зачтено
хорошо (4)	Студент знает программный материал, грамотно и по сути излагает его в устной или письменной форме, допуская незначительные неточности в утверждениях, трактовках, определениях и категориях или незначительное количество ошибок. При этом владеет необходимыми умениями и навыками при выполнении практических задач.	
удовлетворительно (3)	Студент знает только основной программный материал, допускает неточности, недостаточно чёткие формулировки, непоследовательность в ответах, излагаемых в устной или письменной форме. При этом недостаточно владеет умениями и навыками при выполнении практических задач. Допускает до 30% ошибок в излагаемых ответах.	
неудовлетворительно (2)	Студент не знает значительной части программного материала. При этом допускает принципиальные ошибки в доказательствах, в трактовке понятий и категорий, проявляет низкую культуру знаний, не владеет основными умениями и навыками при выполнении практических задач. Студент отказывается от ответов на дополнительные вопросы.	не зачтено

7. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

1. Златопольский, Д. М. Программирование : типовые задачи, алгоритмы, методы / Златопольский Д. М. - 4-е изд. - Москва : Лаборатория знаний, 2020. - 226 с. Систем. требования: Adobe Reader XI ; экран 10". - ISBN 978-5-00101-789-9. - Текст : электронный // ЭБС "Консультант студента" : [сайт]. - URL : <https://www.studentlibrary.ru/book/ISBN9785001017899.html> (дата обращения: 12.04.2023).

2. Нагаева, И. А. Программирование: delphi: учеб. пособие для академического бакалавриата / И. А. Нагаева, И. А. Кузнецов ; под ред. И. А. Нагаевой. — Москва: Издательство Юрайт, 2019. — 302 с. — (Серия: Бакалавр. Академический курс). — ISBN 978-5-534-07098-9. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://biblio-online.ru/book/programmirovaniye-delphi-444273>

3. Тузовский, А. Ф. Объектно-ориентированное программирование: учеб. пособие для прикладного бакалавриата / А. Ф. Тузовский. — Москва: Издательство Юрайт, 2019. — 206 с. — (Серия: Университеты России). — ISBN 978-5-534-00849-4. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://biblio-online.ru/book/obektno-orientirovannoe-programmirovaniye-434045>

4. Канг, Ш. Цифровая дисциплина / Ш. Канг; Пер. с англ. - Москва: Альпина Паблишер, 2022. - 364 с. - ISBN 978-5-9614-7305-6. - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <https://www.studentlibrary.ru/book/ISBN9785961473056.html> (дата обращения: 12.04.2023).

б) дополнительная литература:

1. Черпаков, И. В. Основы программирования: учебник и практикум / И. В. Черпаков. — Москва: Издательство Юрайт, 2019. — 219 с. — (Серия: Профессиональное образование). — ISBN 978-5-9916-9984-6. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://biblio-online.ru/book/osnovy-programmirovaniya-436557>

2. Искусственный интеллект, аналитика и новые технологии / - Москва: Альпина Паблишер, 2022. - 200 с. (Серия "Harvard Business Review: 10 лучших статей") - ISBN 978-5-9614-4791-0. - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <https://www.studentlibrary.ru/book/ISBN9785961447910.html> (дата обращения: 12.04.2023).

3. Борисов, С. В. Программирование на языке Delphi Win32 Language : учебно-методическое пособие / С. В. Борисов, О. Б. Пашенко, И. Л. Серебрякова. - Москва: Издательство МГТУ им. Н. Э. Баумана, 2019. - 86 с. - ISBN 978-5-7038-5212-5. - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <https://www.studentlibrary.ru/book/ISBN9785703852125.html> (дата обращения: 02.04.2023).

в) методическая литература:

1. Карчевский В.П., Волков А.П., Чёрная Е.С., Авершина М.В., Тимошенко Д.С., Ганзенко И.В., Труфанова М.К., Владарский И.В.

Исследование тенденций развития и инноваций в образовании с использованием искусственного интеллекта: учебное пособие для дополнительного изучения информационных технологий, робототехники и искусственного интеллекта в инженерно-педагогическом образовании для студентов очной и заочной форм обучения направления подготовки «Профессиональное обучение. Информационные технологии и системы» / В.П. Карчевский, А.П. Волков, Е.С. Чёрная, М.В. Авершина, Д.С. Тимошенко, И.В. Ганзенко, М.К. Труфанова, И.В. Владарский; под общ. редакцией В.П. Карчевского. – Луганск: СИПМ ЛГУ им. В.ДАЛЯ, 2021. – 1024 с.

г) интернет-ресурсы:

Министерство науки и высшего образования РФ – <https://minobrnauki.gov.ru/>

Федеральная служба по надзору в сфере образования и науки – <http://obrnadzor.gov.ru/>

Портал Федеральных государственных образовательных стандартов высшего образования – <http://fgosvo.ru>

Федеральный портал «Российское образование» – <http://www.edu.ru/>

Информационная система «Единое окно доступа к образовательным ресурсам» – <http://window.edu.ru/>

Федеральный центр информационно-образовательных ресурсов – <http://fcior.edu.ru/>

Электронные библиотечные системы и ресурсы

1. Электронно-библиотечная система «Консультант студента» – <http://www.studentlibrary.ru/cgi-bin/mb4x>

2. Электронная библиотека ФГБОУ ВО «ЮРГПУ (НПИ) имени М.И. Платова» «МегаПро» <https://libweb.srspu.ru/MegaProWeb/Web>.

Информационный ресурс библиотеки образовательной организации

3. Научная библиотека имени А. Н. Коняева – <http://biblio.dahluniver.ru/>

8. Материально-техническое и программное обеспечение дисциплины

Освоение дисциплины «Объектно-ориентированное программирование и технологии разработки программного обеспечения» предполагает использование академических аудиторий, соответствующих действующим санитарным и противопожарным правилам и нормам.

Прочее: рабочее место преподавателя, оснащенное компьютером с доступом в Интернет.

Программное обеспечение:

Функциональное назначение	Бесплатное программное обеспечение	Ссылки
Офисный пакет	Libre Office 6.3.1	https://www.libreoffice.org/ https://ru.wikipedia.org/wiki/LibreOffice

Операционная система	UBUNTU 19.04	https://ubuntu.com/ https://ru.wikipedia.org/wiki/Ubuntu
Браузер	Firefox Mozilla	http://www.mozilla.org/ru/firefox/fx
Браузер	Opera	http://www.opera.com
Почтовый клиент	Mozilla Thunderbird	http://www.mozilla.org/ru/thunderbird
Файл-менеджер	Far Manager	http://www.farmanager.com/download.php
Архиватор	7Zip	http://www.7-zip.org/
Графический редактор	GIMP (GNU Image Manipulation Program)	http://www.gimp.org/ http://gimp.ru/viewpage.php?page_id=8 http://ru.wikipedia.org/wiki/GIMP
Редактор PDF	PDFCreator	http://www.pdfforge.org/pdfcreator
Аудиоплеер	VLC	http://www.videolan.org/vlc/

9. Оценочные средства по дисциплине

Паспорт

оценочных средств по учебной дисциплине

«Объектно-ориентированное программирование и технологии разработки программного обеспечения»

Перечень компетенций (элементов компетенций), формируемых в результате освоения учебной дисциплины (модуля) или практики

№ п/п	Код контролируемой компетенции	Формулировка контролируемой компетенции	Индикаторы достижений компетенции (по реализуемой дисциплине)	Контролируемые темы учебной дисциплины, практики	Этапы формирования (семестр изучения)
1	УК-6.	Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни	УК-6.1. УК-6.2. УК-6.3. УК-6.4.	Тема 1.	7
				Тема 2.	7
				Тема 3.	7
				Тема 4.	7
				Тема 5.	7
				Тема 6.	7
				Тема 7.	7
				Тема 8.	7
				Тема 9.	7
				Тема 10.	7
				Тема 11.	7
				Тема 12.	7
				Тема 13.	7
				Тема 14.	7
				Тема 15.	7
				Тема 16.	7
				Тема 17.	7

				Тема 18.	7
2	ОПК-1.	Способен осуществлять профессиональную деятельность в соответствии с нормативными правовыми актами в сфере образования и нормами профессиональной этики	ОПК-1.1. ОПК-1.2. ОПК-1.3. ОПК-1.4.	Тема 1.	7
				Тема 2.	7
				Тема 3.	7
				Тема 4.	7
				Тема 5.	7
				Тема 6.	7
				Тема 7.	7
				Тема 8.	7
				Тема 9.	7
				Тема 10.	7
				Тема 11.	7
				Тема 12.	7
				Тема 13.	7
				Тема 14.	7
				Тема 15.	7
				Тема 16.	7
				Тема 17.	7
				Тема 18.	7
3	ПК-2	Способен выполнять работы по созданию, модификации и сопровождению информационных систем	ПК-2.1. ПК-2.2. ПК-2.3. ПК-2.4. ПК-2.5. ПК-2.6. ПК-2.7.	Тема 1.	7
				Тема 2.	7
				Тема 3.	7
				Тема 4.	7
				Тема 5.	7
				Тема 6.	7
				Тема 7.	7
				Тема 8.	7
				Тема 9.	7
				Тема 10.	7
				Тема 11.	7
				Тема 12.	7
				Тема 13.	7
				Тема 14.	7
				Тема 15.	7
				Тема 16.	7
				Тема 17.	7
				Тема 18.	7

Показатели и критерии оценивания компетенций, описание шкал оценивания

№ п/п	Код контролируемой компетенции	Индикаторы достижений компетенции (по реализуемой дисциплине)	Перечень планируемых результатов	Контролируемые темы учебной дисциплины	Наименование оценочного средства
1	УК-6	УК-6.1. УК-6.2. УК-6.3. УК-6.4.	Знать: принципы применения компьютерных технологий в учебном процессе; основы объектно-ориентированного проектирования и анализа. Уметь: осуществлять	Тема 1; Тема 2; Тема 3; Тема 4; Тема 5; Тема 6; Тема 7;	Вопросы и задания к лабораторным работам, вопросы к контрольным

			объектно-ориентированную декомпозицию и объектно-ориентированный анализ предметной области в условиях ее формализации с помощью процедур объектно-ориентированного анализа, проектирования и программирования., используя иерархию классов, диаграммы процессов и объектов. Владеть: навыками создания программных кодов, модулей и проектов для решения типовых задач обработки данных; навыками отладки программ с использованием систем программирования на компьютере; навыками использования современных интегрированных сред разработчика приложений и программных систем; индивидуальной самостоятельной разработки проектов приложений для решения математических, графических задач.	Тема 8; Тема 9; Тема 10; Тема 11; Тема 12; Тема 13; Тема 14; Тема 15; Тема 16; Тема 17; Тема 18.	работам, вопросы к экзамену.
2	ОПК-1	ОПК-1.1. ОПК-1.2. ОПК-1.3. ОПК-1.4.	Знать: особенности алгоритмизации и программирование вычислительных процессов на языках программирования высокого уровня. Уметь: создавать программные коды, модули и проекты решения производственных задач, отлаживать программы с использованием систем программирования на компьютере. Владеть: навыками участия в разработке программного обеспечения автоматизированных систем, используя технологии и языки программирования, работая в коллективе разработчиков, на основании технического задания на разработку системы и данных обследования объекта автоматизации.	Тема 1; Тема 2; Тема 3; Тема 4; Тема 5; Тема 6; Тема 7; Тема 8; Тема 9; Тема 10; Тема 11; Тема 12; Тема 13; Тема 14; Тема 15; Тема 16; Тема 17; Тема 18.	Вопросы и задания к лабораторным работам, вопросы к контрольным работам, вопросы к экзамену.
3	ПК-2	ПК-2.1. ПК-2.2.	Знать: основные принципы объектно-ориентированного	Тема 1; Тема 2;	Вопросы и задания к

		ПК-2.3. ПК-2.4. ПК-2.5. ПК-2.6. ПК-2.7.	программирования; топологии языков программирования; объектно-ориентированные языки программирования. Уметь: применять объектно-ориентированные языки программирования для решения задач предметной области. Владеть: навыками работы с компьютером для решения практических и исследовательских задач; навыками разработки и тестирования проектов.	Тема 3; Тема 4; Тема 5; Тема 6; Тема 7; Тема 8; Тема 9; Тема 10; Тема 11; Тема 12; Тема 13; Тема 14; Тема 15; Тема 16; Тема 17; Тема 18.	лаборатор- ным работам, вопросы к контрольным работам, вопросы к экзамену.
--	--	---	--	---	---

**Оценочные средства по дисциплине
«Объектно-ориентированное программирование и технологии разработки
программного обеспечения».**

Вопросы к контрольным работам

1. Что называют языками высокого и низкого уровней?
2. Назовите этапы развития языков программирования высокого уровня.
3. Какие существуют топологии языков программирования?
4. Дайте определение понятию «декомпозиция».
5. В чём заключается метод пошаговой детализации?
6. Приведите пример процедурной декомпозиции и её реализации.
7. Дайте определение понятиям «объектно-ориентированное программирование», «класс».
8. Какие существуют принципы ООП? Особенности их использования.
9. Что называют иерархией в ООП?
10. Какие языки программирования относят к объектно-ориентированным?
11. Назовите этапы разработки программных средств с использованием ООП.
12. Что называют диаграммой классов?
13. Что называют ассоциацией в ООП? Приведите примеры.
14. Что называют агрегацией в ООП? Приведите примеры.
15. Что называют классом? Приведите примеры.
16. Что называют методами? Приведите примеры.
17. Как осуществляется описание класса, например, окно с позиции ООП? Приведите примеры описания классов.
18. Что называют конструктором и деструктором? Как происходит их инициализация?
19. Что называют модификацией? Особенности использования модификаций в ООП.
20. Что называют листингом программы? Как происходит оформление

исходных листингов?

21. С помощью какого класса обрабатываются системные сообщения Windows?

22. Какие существуют особенности использования процедур и функций вызова диалоговых окон?

23. Что значит понятие объектно-ориентированного программирования?

24. Какие преимущества объектно-ориентированного программирования?

25. Что называют инкапсуляцией в ООП?

26. Что называют полиморфизмом в ООП?

27. Что называют наследованием в ООП? Приведите примеры.

28. Как реализуется наследование в ООП Delphi?

29. Что предусматривает простое наследование?

30. Что предусматривает множественное наследование?

31. Что такое метод поиска в иерархии классов?

32. Что называют полем в объектно-ориентированном программировании?

33. Что называют свойствами в ООП? Расскажите принцип их работы.

34. Что называют сеттером и геттером? Приведите примеры их использования.

35. Что называют абстрактными методами? Приведите примеры.

36. Что включает в себя понятие «программное обеспечение»?

37. Назовите и характеризуйте основные категории программного обеспечения.

38. В чем отличие прикладных программ от системных и инструментальных?

39. Какие принципы ООП включены в программное рисование в Delphi? Приведите примеры.

40. Как осуществляется работа с изображениями в Delphi?

41. Что такое массив?

42. Какие бывают массивы?

43. Что называют сокетом с точки зрения программирования? Приведите примеры.

44. Что называют потоками данных? Приведите примеры.

45. Опишите структуру потока, состояние потока.

46. Что называется переключение контекста в программировании?

47. Какие существуют средства синхронизации потоков? Назначение и принцип их выполнения?

Критерии и шкала оценивания по оценочному средству «контрольная работа»

Шкала оценивания (интервал баллов)	Критерий оценивания
5	Контрольная работа выполнена на высоком уровне (правильные ответы даны на 90-100% вопросов/задач)
4	Контрольная работа выполнена на среднем уровне (правильные ответы даны на 75-89% вопросов/задач)
3	Контрольная работа выполнена на низком уровне (правильные ответы даны на 50-74% вопросов/задач)
2	Контрольная работа выполнена на неудовлетворительном уровне (правильные ответы даны менее чем на 50%)

Раздел «Введение в объектно-ориентированное программирование»

1. Построить концептуальную модель предметной области, состоящую из 2-3 классов, один из которых соответствует номеру задания по варианту. Пример варианта.

№ Варианта	Класс
1	Студент

2. Согласно своему варианту, создать класс и заполнить его свойства (5 объектов).
3. Вывести по своему усмотрению результат вычисления с выбором подходящего объекта. Пример варианта задания

№ п/п	Объект	Свойства
1	Успеваемость	Номер зачетной книжки, Шифр группы, Название дисциплины, Дата получения оценки, Оценка, Фамилия преподавателя

4. Выполнить вывод экземпляра класса (код программы) по нажатию на кнопку.
5. Создать второй объект класса (исходными данными задаться самостоятельно).
6. Выполнить считывание фамилии первого жителя с Edit 1 и Edit 2, а
7. номера квартиры - с Edit3 и Edit4. Добавьте компоненту Memo1.
8. Преобразовать программу (удалить Edit3-Edit4). Выводить данные о жильце и номере его квартиры в Memo1.
9. Добавить на форму компонент Stringgrid (таблица 2x9) без фиксированных столбцов и строк.
10. Дополнительное задание: выводить номера квартир по возрастанию с соответствующей фамилией жильца)
11. Согласно своему варианту по формуле: номер варианта по списку +1 , создайте класс и заполните его свойства (5 объектов), вводя данные с формы
12. Вывести по своему усмотрению результат вычисления с выбором подходящего объекта (поиск максимального, минимального, поиск по букве в фамилии или начальной букве и т.д.). Оформить отчет.

Вариант задания

№ п/п	Объект	Свойства
1	Книга	Автор, Название, Инвентарный номер, Издательство, Количество страниц, Цена

13. По изученным структурам создать рабочую область – форму с четырьмя кнопками: Students, Teachers, Marks and Exit.
14. В конструктор constructor TStudent.CreateWithFirstMark дописать: ShowMessage('Ученику '+NewFirstName+' '+NewLastName+' '+' оценка '+IntToStr(FirstMark)); end;
15. Написать функцию для подсчета количества лет студента. Для этого подвести курсор к function Age: byte; и нажать Ctrl+Shift+C

Задание индивидуальное

Согласно номеру своего варианта добавить каждому классу ровно столько объектов, каков номер вашего варианта.

Например, вариант 15:

2 – по условию $teachers + 15 = 17$

2 – по условию $students + 15 = 17$

2 – по условию $marks + 15 = 17$

Дополнительное задание

Выводить все объекты в Edit (например, в цикле с помощью FindComponent('Edit'+IntToStr(i)) as TEdit) или в другие компоненты.

Контрольные вопросы к лабораторным занятиям

1. Что называют диаграммой класса?
2. Какие существуют программные средства создания диаграмм классов?
3. Какие типы данных существуют в Delphi?
4. Что называют классом? Приведите примеры.
5. Особенности создания блок-схем в процедурном программировании и диаграмм классов в объектно-ориентированном.
6. Что называют методами? Приведите примеры.
7. Как осуществляется описание класса, например, окно с позиции ООП?

Приведите примеры описания классов.

8. Как создаются классы в ООП Delphi?
9. Что называют полями и свойствами классов? Приведите примеры.
10. Что называют событиями в ООП? Особенности событийности в операционных системах Windows. Приведите примеры.
11. Что называют наследованием в ООП? Приведите примеры.
12. Как реализуется наследование в ООП Delphi?
13. Что предусматривает простое наследование?
14. Что предусматривает множественное наследование?

Задания к лабораторным работам

Раздел «Реализация принципов объектно-ориентированного программирования»

1. Выполнить создание класса «Треугольник» Класс содержит три поля с соответствующими сторонами и два метода, которые позволяют найти площадь и периметр этого треугольника.

2. Далее, стоит задача сделать так, чтобы была возможность устанавливать значения сторон треугольника, присваивая ему периметр. Реализуем еще один метод, который будет присваивать поля a, b и c в зависимости от переданного ему значения периметра:

3. Для того, чтобы объединить такие методы GetPerimeter и SetPerimeter в единую структуру, разработаны свойства (property). Объявляются они так же как и поля или методы внутри класса, их синтаксис выглядит следующим образом:

property {Название свойства}: {Тип данных} read {Getter} write {Setter};

4. Согласно своему варианту, создайте класс и заполните его свойства. Выполните проверку работы программы. Оформить отчет.

Вариант задания

№ п/п	Класс	Поля	Свойства
1	Правильный многоугольник	Количество сторон, радиус	Площадь многоугольника

5. Дополнительные задания к лабораторной работе:
- Построить фигуру (Triangle) на форме относительно прямоугольной системы координат. Данные должны быть динамическими.
 - Построить фигуру на форме относительно прямоугольной системы координат согласно своему варианту. Данные должны быть динамическими.
6. Написать приложение, которое создает массив объектов и выполняет его обработку.

7. Написать приложение по варианту, которое создает массив объектов и выполняет его обработку. Поля и методы класса выбрать в зависимости от свойств самого объекта и тех действий, которые необходимо выполнить. Обязательные методы:

конструктор, предназначенный для инициализации элементов массива объектов; метод вывода списка объектов.

8. Выполнить проверку работы программы. Оформить отчет.

Вариант задания

Объект: пассажир с багажом. Создать массив объектов и выбрать пассажиров с максимальным весом багажа.

9. Создать иерархию классов из предложенной области.

Поля и методы класса выбрать в зависимости от свойств объектов и тех действий, которые необходимо выполнить, но не менее трех полей.

Обязательные методы: конструктор, предназначенный для инициализации элементов массива объектов; деструктор метод вывода результатов.

Количество дочерних классов – не менее двух. Написать приложение по варианту.

Вариант задания

№ варианта	Область	Обязательный тип поля	Обязательный вывод метода
7	Классы в ООП	tdate	

10. Создать иерархию классов из предложенной области.

Поля и методы класса выбрать в зависимости от свойств объектов и тех действий, которые необходимо выполнить, но не менее трех полей.

Количество дочерних классов – не менее трех.

Обязательное наличие вычисляемых полей.

Заполнить в таблице согласно варианту метод вызова функции родительского класса и всех дочерних классов

Написать приложение по варианту.

Вариант задания

№ варианта	Область	Метод вызова функции родительского класса и всех дочерних классов
1	Память компьютера	

11. Используя 3 метода, решить: даны 3 числа, в зависимости от метода, найти сумму пяти чисел, разность двух чисел по модулю или произведение трех чисел. Ввод данных выполнить с Label, Edit или других компонент.

12. Разработайте приложение, обрабатывающее исключительную ситуацию, согласно вашему варианту индивидуального задания.
Вариант задания

№ Варианта	№ Задания
1	1,2

13. Создать программу, позволяющую пользователю ввести два числа, над которыми программа выполнит деление. Необходимо поместить на форму три объекта класса TEdit - два для операндов, один – для результата и кнопку (объект класса TButton), нажимая на которую пользователь выполняет деление. Исключить попытку деления на ноль, а так же введения символов вместо цифр. Выдать сообщение о типе возникшей ошибке.

14. Создать программу, вычисляющую корни квадратного уравнения ($ax^2+bx+c=0$). Необходимо поместить на форму четыре объекта класса TEdit - три для коэффициентов квадратного уравнения, один – для результата и кнопку (объект класса TButton), нажимая на которую пользователь выполняет нахождение корней. Исключить ввод символов вместо цифр, получение отрицательного дискриминанта и ввод $a = 0$. Вывести при всех типах ошибок одно и то же сообщение.

15. Создать Класс Animal, который содержит в разделе public объявление конструктора Create и объявление метода-функции: Voice – звук, издаваемый животным. Тип результата возвращаемого функцией, – string. Метод Voice объявить виртуальным и абстрактным.

16. Класс Dog объявить потомком класса Animal:

17. Класс Cat объявить потомком класса Animal.

18. Раздел public класса содержит те же определения, что и соответствующий раздел класса Dog.

19. В реализациях конструктора каждого класса переменной Kind присваивается имя соответствующего животного, например, для класса Animal.

20. В реализациях методов Voice возвращается звук, издаваемый животным.

21. В реализациях методов Eat возвращается название пищи, которой питается соответствующее животное.

22. Задать имя модуля и имя проекта, в который этот модуль будет включен.

Контрольные вопросы к лабораторным занятиям

1. Какие бывают арифметические операции?
2. Что называют инкапсуляцией в ООП? Примеры реализации.
3. Что называют методами в ООП?
4. Какие бывают типы методов в ООП? Приведите примеры.
5. Преимущества и недостатки использования инкапсулирования в ООП.
6. Инкапсуляция и описание классов в Delphi.
7. Классы и директивы в инкапсуляции.
8. Директивы protected и private.

9. Как вызываются методы классов в программе?
10. Могут ли идентификаторы полей объекта совпадать с идентификаторами формальных параметров методов?
11. Отличаются ли заголовки методов в описании объекта и при описании метода?
12. Понятие наследования.
13. Как изменяются атрибуты элементов класса при наследовании?
14. Опишите механизм наследования классов.
15. Понятие полиморфизма. В чем состоит суть полиморфизма функций?
16. Что такое перекрывание функций?
17. Для чего выполняется перекрывание функций в производных классах?
18. Как проявляется скрытие методов базового класса, при перекрывании в производном?
19. Как выполнить скрытый метод базового класса?
20. Можно ли в указатель на базовый класс занести указатель производного объекта?
21. Обработка исключительных ситуаций.
22. Использование диаграмма классов в ООП.
23. Понятие классов, методов, объектов.
24. Реализация классов, вызов методов.
25. Поля и свойства классов.

Задания к лабораторным работам

Раздел «Технологии разработки программного обеспечения»

1. С помощью Microsoft Visio создайте иерархии классов по предложенным областям информационных технологий. В классах предусмотреть атрибуты, типы данных, параметры, методы, а так же связи.

№ варианта	Область информационных технологий
1	интеллектуальные системы (Artificial intelligence);

2. Согласно варианту выполнить задание: добавить в программу предложенный объект и выполнить заливку всех объектов. Добавить по вашему усмотрению объект и выполнить его заливку (рисунок выбранного объект внести в таблицу – поле «Свой объект»). Написать приложения и нарисовать диаграмму классов, их методов и свойств.

№ варианта	Объект	Тип заливки	Свой объект
1		bsClear	

3. Согласно варианту выполнить задание: добавить в программу

предложенный объект; добавить новый объект; и выполнить заливку всех объектов. Заполнить таблицу – поле «Свой объект»). Написать приложения и нарисовать блок-схему алгоритма работы программы/

№ варианта	Объект	Свой объект
1		

Контрольные вопросы к лабораторным занятиям

1. Что называют диаграммой класса?
2. Что называют иерархией?
3. Что представляет собой иерархия классов?
4. Что называют апкастингом в ООП?
5. Приведите примеры использования иерархий классов.
6. Как реализуется наследование в ООП Delphi?
7. Что называют листингом программы? Как происходит оформление исходных листингов?
8. Как подключить модуль OpenGL?
9. Как задать формат пикселя?
10. Как изменить размер сцены, на которой будут отображаться объекты?
11. Процесс инициализации OpenGL.
12. Назначение функции «wglCreateContext».
13. Как создать графическим примитив средствами OpenGL?

Критерии и шкала оценивания по оценочному средству «лабораторная работа»

Шкала оценивания (интервал баллов)	Критерий оценивания
5	Задание выполнено на высоком уровне (студент в полном объеме осветил рассматриваемую проблематику, привел аргументы в пользу своих суждений, владеет профильным понятийным (категориальным) аппаратом и т.п.)
4	Задание выполнено на среднем уровне (студент в целом осветил рассматриваемую проблематику, привел аргументы в пользу своих суждений, допустив некоторые неточности и т.п.)
3	Задание выполнено на низком уровне (студент допустил существенные неточности, изложил материал с ошибками, не владеет в достаточной степени профильным категориальным аппаратом и т.п.)
2	Задание выполнено на неудовлетворительном уровне или не представлен (студент не готов, не выполнил задание и т.п.)

Оценочные средства для промежуточной аттестации (экзамен)

Теоретические вопросы

1. История языков программирования. Возникновение объектной

декомпозиции.

2. История возникновения RAD – систем. Delphi. Основные понятия.
3. Основные принципы ООП.
4. Инкапсуляция и описание классов в Delphi.
5. Понятие инкапсуляции. Примеры реализации.
6. Полиморфизм.
7. Понятие «свойство». С какой целью целесообразно использовать механизм свойств?
8. Понятие «объект» в ООП.
9. Понятие «класс» в ООП.
10. Основные этапы разработки программных систем с использованием ООП.
11. Представление графического контекста Windows в Delphi.
12. Файлы, соответствующие каждой форме приложения в Delphi.
13. Структуру приложения на Delphi.
14. Правила именования переменных и компонентов.
15. Методы расположения визуальных компонентов на экране и управления их размерами.
16. Свойства видимости и доступности визуальных компонентов.
17. Компоненты, предназначенные для ввода и вывода на экран текстовой информации.
18. Возможности элементов управления – кнопки.
19. Диалоговые компоненты Windows.
20. Понятие формы и основы использования форм в Delphi.
21. Основы организации многооконных приложений в Delphi.
22. Стандартные элементы управления Windows.
23. Понятие иерархии. Примеры иерархии классов.
24. Приоритеты потоков.
25. Стандартные функции для работы с файлами.
26. Стандартные диалоговые окна Windows.
27. Объектно-ориентированная методология разработки программ.
28. Событийное программирование. События мыши. Переменная Shift. Переменная Button. Переменная Sender.
29. Событийное программирование. События мыши. Обработка нажатия и отпускания клавиши мыши.
30. События клавиатуры. Обработка нажатия/отпускания клавиши.
31. События клавиатуры. Приоритет (OnShortCut) и перехват (KeyPreview) обработки.
32. Событийное программирование. События клавиатуры. Переменная Sender.
33. Что называет «графический движок»? Основные отличия графического движка от игрового?
34. Расскажите историю графических движков.
35. Аппаратная абстракция графических движков.
36. Понятие графического интерфейса пользователя. Структура взаимодействия пользователя с графическим интерфейсом. Примеры.
37. Структура игрового движка.

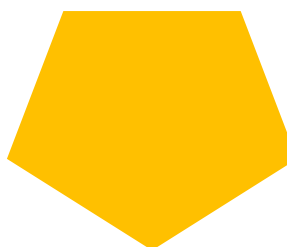
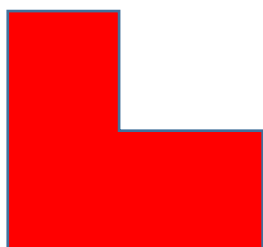
38. Назначение использования искусственного интеллекта в игровых движках.

Практические задания

1. С помощью ООП написать программу, обрабатывающую следующие данные об объекте:

Объект	Свойства
Квартира	Общая площадь, Жилая площадь, Площадь кухни, Наличие лоджии, Наличие санузла и его характеристики (совмещенный или нет), Панельный или кирпичный дом, Этаж, Общее количество этажей, Адрес, Стоимость, Район города

- Выполнить вычисления в полях данных. Количество объектов – 5.
2. С помощью средств ООП в Delphi создать движущиеся фигуры №3, №6.



Свойство заливки фигуры и контур фигуры.

Критерии и шкала оценивания по оценочному средству промежуточный контроль («экзамен»)

Шкала оценивания (интервал баллов)	Критерий оценивания
отлично (5)	Студент глубоко и в полном объёме владеет программным материалом. Грамотно, исчерпывающе и логично его излагает в устной или письменной форме. При этом знает рекомендованную литературу, проявляет творческий подход в ответах на вопросы и правильно обосновывает принятые решения, хорошо владеет умениями и навыками при выполнении практических задач.
хорошо (4)	Студент знает программный материал, грамотно и по сути излагает его в устной или письменной форме, допуская незначительные неточности в утверждениях, трактовках, определениях и категориях или незначительное количество ошибок. При этом владеет необходимыми умениями и навыками при выполнении практических задач.
удовлетворительно (3)	Студент знает только основной программный материал, допускает неточности, недостаточно чёткие формулировки, непоследовательность в ответах, излагаемых в устной или письменной форме. При этом недостаточно владеет умениями и навыками при выполнении практических задач. Допускает до 30% ошибок в излагаемых ответах.
неудовлетворительно (2)	Студент не знает значительной части программного материала. При этом допускает принципиальные ошибки в доказательствах, в трактовке понятий и категорий, проявляет низкую культуру знаний, не владеет основными умениями и навыками при выполнении практических задач. Студент отказывается от ответов на дополнительные вопросы

Лист изменений и дополнений

№ п/п	Виды дополнений и изменений	Дата и номер протокола заседания кафедры (кафедр), на котором были рассмотрены и одобрены изменения и дополнения	Подпись (с расшифровкой) заведующего кафедрой (заведующих кафедрами)