

Минобрнауки России
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)**

УТВЕРЖДАЮ

Заведующий кафедрой
Борисов Дмитрий Николаевич
Кафедра информационных систем



05.05.2025

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Б1.В.01 Программно-распределенные системы

1. Код и наименование направления подготовки:

09.04.02 Информационные системы и технологии

2. Профиль подготовки:

Программные технологии в инфокоммуникационных системах

3. Квалификация (степень) выпускника:

Магистратура

4. Форма обучения:

Заочная

5. Кафедра, отвечающая за реализацию дисциплины:

Кафедра информационных систем

6. Составители программы:

Фертиков Вадим Валериевич, кандидат физ.-мат. наук, доцент

7. Рекомендована:

Научно-методическим советом ФКН, протокол НМС ФКН № 7 от 05.05.2025

8. Учебный год:

2026-2027

9. Цели и задачи учебной дисциплины:

формирование профессиональных компетенций будущих магистров информационных систем и технологий через изучение наиболее общих принципов организации распределенных приложений, вычислительных и операционных систем, технологий их разработки и реализации, практических приемов их применения для реализации распределенных приложений с использованием наиболее известных технологий и моделей программирования. В процессе освоения учебных материалов студент получит знание основных принципов организации распределенных приложений, вычислительных и операционных систем; представление о наиболее употребительных моделях распределенного программирования, в том числе, процедурной, компонентной и Grid, о принципах функционирования соответствующей инфраструктуры их поддержки в гетерогенной среде, о современных стандартах на последовательности протоколов разных уровней и архитектуру систем

программирования для реализации распределенных вычислений; умение применять перечисленные сведения для проектирования систем распределенных вычислений; навыки реализации распределенных приложений с использованием широко распространенных технологий и моделей программирования.

10. Место учебной дисциплины в структуре ООП:

обязательная дисциплина относится к части, формируемой участниками образовательных отношений. Блока 1. Дисциплины (модули) (Б1.В); входные знания в объеме базовых курсов по программам обучения ступени бакалавриата в области информационных технологий, управления данными, языков и систем программирования, методов и средств проектирования информационных систем и технологий, а также обязательных курсов, предусмотренных учебным планом для изучения в семестрах 1 – 2: «Математические методы в современных информационных технологиях», «Системная инженерия».

№ п/п	Наименование дисциплин учебного плана, с которым организована взаимосвязь дисциплины рабочей программы	№ разделов дисциплины рабочей программы, связанных с указанными дисциплинами
1	Математические методы в современных информационных технологиях	1, 2, 4, 9, 10
2	Системная инженерия	1, 5, 10, 11

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

Код и название компетенции	Код и название индикатора компетенции	Знания, умения, навыки
ПК-3 Способен проектировать структуру программ и реализует алгоритмы с учётом требований к надёжности, модульности и масштабируемости	ПК-3.2 Использует современные языки программирования для реализации алгоритмов	знать: основные принципы организации распределенных приложений, вычислительных и операционных систем; наиболее употребительные модели распределенного программирования, в том числе, процедурную, компонентную и Grid, принципы функционирования соответствующей инфраструктуры их поддержки в гетерогенной среде, стандарты на последовательности протоколов разных уровней и архитектуру систем программирования для реализации распределенных вычислений;

Код и название компетенции	Код и название индикатора компетенции	Знания, умения, навыки
	ПК-3.3 Проводит отладку и тестирование кода, оптимизирует его структуру и производительность	уметь: применять перечисленные сведения для проектирования систем распределенных вычислений; осуществлять моделирование проектируемых распределенных приложений и их отдельных компонентов на базе стандартных средств разработки ПО (SDK);
ПК-5 Способен описывать и сопровождать архитектуру вычислительных систем и комплексов, обеспечивает их работоспособность и безопасность	ПК-5.3 Выполняет оптимизацию распределения вычислительных ресурсов и компонентов вычислительной сети	владеть (иметь навык(и)): навыками реализации распределенных приложений с использованием широко распространенных технологий и моделей программирования.

12. Объем дисциплины в зачетных единицах/час:

3/108

Форма промежуточной аттестации:

Зачет с оценкой, Курсовая работа

13. Трудоемкость по видам учебной работы

Вид учебной работы	Триместр 5	Триместр 6	Всего
Аудиторные занятия	14	6	20
Лекционные занятия	8		8
Практические занятия		6	6
Лабораторные занятия	6		6
Самостоятельная работа	22	62	84
Курсовая работа			0
Промежуточная аттестация	0	0	0
Часы на контроль		4	4
Всего	36	72	108

13.1. Содержание дисциплины

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК
1	Эволюция технологии доступа к данным в информационных системах. Понятие DCE (Distributed Computing Environment).	Этапы развития технологии: модель "Host+терминалы"; настольные СУБД; их сетевые версии; архитектура "клиент/сервер"; трехзвенные и многозвенные системы; приложения для WEB-серверов; Grid-технологии. Понятие DCE - инфраструктуры для разработки и функционирования распределенных приложений в гетерогенных системах: модель программирования, место на уровнях модели OSI-RM (Open System Interconnection Reference Model).	Дистанционное чтение лекций https://edu.vsu.ru/course/view.php?id=3994
2	Архитектура средств DCE OSF (Open Software Foundation). Концепция вызова удаленных процедур (Remote Procedure Call, RPC) на примере RPC DCE OSF.	Архитектура средств DCE OSF как интегрированного набора спецификаций стандартов функций, независимого от платформ, операционных систем и сетевых средств: нити, RPC, служба каталогов, служба безопасности, служба времени, распределенная файловая система. Концепция вызова удаленных процедур на примере RPC DCE OSF. Обеспечение прозрачности вызова. Язык определения интерфейсов (Interface Definition Language, IDL) и схема сборки распределенного приложения.	Дистанционное чтение лекций https://edu.vsu.ru/course/view.php?id=3994
3	Механизм вызова RPC. Введение в компонентное программирование.	Механизм вызова RPC. Динамическое связывание. Особенности протокола RPC DCE OSF. Размещение RPC по уровням модели OSI-RM. Введение в компонентное программирование: основные низкоуровневые модели распределенных вычислений, этапы развития парадигмы программирования.	Конспект лекции. Материалы для самостоятельного изучения. https://edu.vsu.ru/course/view.php?id=3994

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК
4	Стратегия Microsoft в области распределенных вычислений. Основы COM (Component Object Model).	Этапы развития технологий по пути к распределенным вычислениям от DLL до Windows DNA (Distributed interNet Application) и .NET. Место COM в архитектуре OSI-RM. Последовательности протоколов. Объектная модель COM.	Материалы для самостоятельного изучения. https://edu.vsu.ru/course/view.php?id=3994
5	COM и объектно-ориентированный подход. COM-интерфейс: идентификация, спецификация, виды реализаций.	COM и объектно-ориентированный подход: полиморфизм, инкапсуляция, механизмы поддержки наследования. Идентификация COM-интерфейсов (механизмы глобальной идентификации), спецификация (стандарты двоичного интерфейса, язык описания интерфейсов IDL), виды реализаций (интерфейс с виртуальной таблицей, диспинтерфейс, дуальный интерфейс).	Материалы для самостоятельного изучения. https://edu.vsu.ru/course/view.php?id=3994
6	Классы COM-объектов. Создание экземпляров объектов. Многопоточность в COM. Механизмы повторного использования COM-объектов.	Классы COM-объектов. Библиотека COM: основные задачи. Фабрика класса. Модели многопоточности. Механизмы повторного использования: включение, агрегирование.	Материалы для самостоятельного изучения. https://edu.vsu.ru/course/view.php?id=3994
7	Маршалинг - механизм взаимодействия сервера и клиента. Информация о типе. Библиотека типа.	Маршалинг: назначение и реализация в различных конфигурациях распределенного приложения. Статическое и динамическое связывание. Понятие информации о типе. Библиотека типа. Стандартные интерфейсы библиотеки типа для доступа к информации о типе.	Материалы для самостоятельного изучения. https://edu.vsu.ru/course/view.php?id=3994

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК
8	<p>Постоянные данные объектов. Сервисы перманентности. Управление перманентностью. Объекты с подключением - технология реализации событий COM.</p>	<p>Постоянные данные (состояние) объектов. Сервисы перманентности (Persistent Service), структурированное хранилище. Управление перманентностью, интерфейсы IPersist*. Объекты с подключением (Connectable Objects): реализация событий, входящие и исходящие интерфейсы, объекты-точки связи и приемники (sink).</p>	<p>Материалы для самостоятельного изучения. https://edu.vsu.ru/course/view.php?id=3994</p>
9	<p>Распределенная модель компонентных объектов (Distributed COM). Понятие об Enterprise JavaBeans (EJB).</p>	<p>Особенности создания удаленного объекта, механизмы доступа к удаленному объекту (экспортер объектов, формат объектной ссылки, инфраструктура разрешателей OXID), обеспечение контроля доступа к объектам распределенного приложения (защита активизации и защита вызовов). Понятие об Enterprise JavaBeans - спецификации набора распределенных Java-компонентов для расширения Java средствами поддержки межплатформенной объектной технологии.</p>	<p>Материалы для самостоятельного изучения. https://edu.vsu.ru/course/view.php?id=3994</p>
10	<p>Введение в Common Object Request Broker Architecture (CORBA). Объектная модель CORBA. Язык спецификации интерфейсов IDL.</p>	<p>CORBA, как часть Object Management Architecture (OMA). Место CORBA в модели OSI-RM. Основные спецификации протоколов. Основные понятия объектной модели CORBA. Основные объектные сервисы. Язык спецификации интерфейсов IDL. Виды интерфейсов и механизмов взаимодействия объектов.</p>	<p>-</p>

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК
11	Структура клиентского приложения. Отображение IDL к языку программирования. Структура и реализации брокера ORB (Object Request Broker).	Структура клиентского приложения. Отображение IDL к языку программирования. Структура брокера ORB: ядро, интерфейс динамического вызова, объектные каркасы и стабы. Объектный адаптер (Object Adapter): назначение, понятие стиля объектной реализации, базовый и целевые объектные адаптеры. Основные реализации ORB.	-
12	Основы архитектуры .NET Microsoft. Система типов .NET.	Архитектура .NET и платформа .NET Framework. Общеязыковая среда выполнения CLR (Common Language Runtime) , общий промежуточный язык CIL (Common Intermediate Language). Управляемый код и управляемые данные. Система типов: общая система типов CTS (Common Type System), типы-значения и ссылочные типы. Обеспечение ООП: поля, методы, свойства, события. Тип делегат.	-
13	Система метаданных CLR. Система выполнения CLR. Механизмы развертывания сборок.	Система метаданных CLR: хранение метаданных вместе с типом, концепция отражения, механизмы динамического связывания, метапрограммирование, атрибуты – средство расширения метаданных. Состав и структура хранения сборки. Механизмы развертывания сборок. Глобальный кэш и кэш загрузки. Строгое имя сборки. Схемы цифрового подписания и проверки аутентичности сборки. Механизмы контроля версий сборки. Интернационализация и локализация приложений.	-

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК
14	Особенности разработки распределенных приложений для функционирования в среде Microsoft RPC.	Лабораторная работа №1. Изучение концепции вызова удаленных процедур (RPC), основ языка определения интерфейсов (IDL) и схемы сборки распределенного приложения. Изучение механизма вызова RPC и динамического связывания. Изучение особенностей разработки распределенных приложений для функционирования в среде Microsoft RPC. Реализация тестового распределенного приложения и демонстрация его работы при различных способах размещения клиентской и серверной частей и различных последовательностях протоколов. Выполнение индивидуального задания по разработке распределенного приложения на основе Microsoft RPC.	Демонстрация выполнения основных операций в формате конференции. https://edu.vsu.ru/course/view.php?id=3994
15	Начальные сведения о технологии OLE-автоматизации. Разработка простого контроллера автоматизации для одного из приложений Microsoft Office.	Лабораторная работа №2. Изучение основ технологии OLE-автоматизации. Реализация учебного примера контроллера автоматизации для одного из приложений Microsoft Office. Демонстрация его возможностей по формированию отчетов с использованием тестовой базы данных. Выполнение индивидуального задания по разработке контроллера автоматизации для одного из приложений Microsoft Office.	Демонстрация выполнения основных операций в формате конференции. https://edu.vsu.ru/course/view.php?id=3994

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК
16	Введение в технологию Borland MIDAS (Multi-tier Distributed Application Services Suite) для разработки многозвенных распределенных приложений и их эксплуатации в корпоративных системах.	Лабораторная работа №3. Изучение основ технологии Borland MIDAS для разработки многозвенных распределенных приложений и их эксплуатации в корпоративных системах. Реализация в технологии учебного распределенного приложения: сервер приложений с доступом к тестовой базе данных и "тонкий" клиент с интерфейсом к данным. Выполнение индивидуального задания по разработке трехзвенного распределенного приложения в технологии Borland MIDAS.	Демонстрация выполнения основных операций в формате конференции. https://edu.vsu.ru/course/view.php?id=3994

13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование темы (раздела)	Лекционные занятия	Практические занятия	Лабораторные занятия	Самостоятельная работа	Всего
1	Эволюция технологии доступа к данным в информационных системах. Понятие DCE (Distributed Computing Environment).	1			5	6
2	Архитектура средств DCE OSF (Open Software Foundation). Концепция вызова удаленных процедур (Remote Procedure Call, RPC) на примере RPC DCE OSF.	1			5	6
3	Механизм вызова RPC. Введение в компонентное программирование.	1			5	6
4	Стратегия Microsoft в области распределенных вычислений. Основы COM (Component Object Model).	1			5	6
5	COM и объектно-ориентированный подход. COM-интерфейс: идентификация, спецификация, виды реализаций.	1			5	6

№ п/п	Наименование темы (раздела)	Лекционные занятия	Практические занятия	Лабораторные занятия	Самостоятельная работа	Всего
6	Классы СОМ-объектов. Создание экземпляров объектов. Многопоточность в СОМ. Механизмы повторного использования СОМ-объектов.	1			5	6
7	Маршалинг - механизм взаимодействия сервера и клиента. Информация о типе. Библиотека типа.	1			5	6
8	Постоянные данные объектов. Сервисы перманентности. Управление перманентностью. Объекты с подключением - технология реализации событий СОМ.	1	1		5	7
9	Распределенная модель компонентных объектов (Distributed СОМ). Понятие об Enterprise JavaBeans (EJB).		1		6	7
10	Введение в Common Object Request Broker Architecture (CORBA). Объектная модель CORBA. Язык спецификации интерфейсов IDL.		1		6	7
11	Структура клиентского приложения. Отображение IDL к языку программирования. Структура и реализации брокера ORB (Object Request Broker).		1		6	7
12	Основы архитектуры .NET Microsoft. Система типов .NET.		1		6	7
13	Система метаданных CLR. Система выполнения CLR. Механизмы развертывания сборок.		1		6	7
14	Особенности разработки распределенных приложений для функционирования в среде Microsoft RPC.			2	6	8

№ п/п	Наименование темы (раздела)	Лекционные занятия	Практические занятия	Лабораторные занятия	Самостоятельная работа	Всего
15	Начальные сведения о технологии OLE-автоматизации. Разработка простого контроллера автоматизации для одного из приложений Microsoft Office.			2	6	8
16	Введение в технологию Borland MIDAS (Multi-tier Distributed Application Services Suite) для разработки многозвенных распределенных приложений и их эксплуатации в корпоративных системах.			2	6	8
		8	6	6	88	108

14. Методические указания для обучающихся по освоению дисциплины

При использовании дистанционных образовательных технологий и электронного обучения выполнять все указания преподавателей, вовремя подключаться к online занятиям, ответственно подходить к заданиям для самостоятельной работы.

Внеаудиторная самостоятельная работа студентов включает проработку материалов лекций, изучение рекомендованной литературы, подготовку к лабораторным работам и их защитам, подготовку к устному опросу.

Самостоятельная работа в аудитории выполняется под непосредственным руководством преподавателя на лабораторных занятиях, призванных обеспечить выборочное использование лекционного материала для более глубокого изучения организации распределенного вычислительного процесса, либо распределенного приложения на локальной сети персональных компьютеров с использованием изучаемых средств и технологий разработки. Для повышения эффективности руководства при проведении лабораторных занятий, демонстрирующих варианты использования отдельных стандартных технологических приемов для решения соответствующих практических задач, необходимо ссылаться на материалы лекций, показывая место решаемых задач в общем плане изложения. Для обеспечения каждого студента индивидуальным комплектом из трех обязательных для выполнения задач, имеется набор заданий соответствующего объема. Проверка результатов выполнения работы включает требования возможной масштабируемости приложения на количество компонентов гетерогенной системы, а также хорошего стиля программирования, специфического для используемого языка разработки. Демонстрации каждого выполненного индивидуального лабораторного задания предшествует обязательная проверка функционирования соответствующего набора тренировочных задач, самостоятельно реализованных студентом.

Учитывая разницу темпов выполнения индивидуальных заданий, преподаватель обеспечивает выполнение студентами дополнительных заданий (общих для студентов, решивших основную задачу) с целью углубленного изучения материала.

15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины

№ п/п	Источник
1	Чуешев, А.В. Распределенные информационные системы : учебно-методическое пособие : [16+] / А.В. Чуешев ; Кемеровский государственный университет. – Кемерово : Кемеровский государственный университет, 2019. – 252 с. : ил. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=571521

б) дополнительная литература:

№ п/п	Источник
1	Чуешев, А.В. Интеграция данных : учебно-методическое пособие / А.В. Чуешев ; Кемеровский государственный университет. – Кемерово : Кемеровский государственный университет, 2018. – 281 с. : ил. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=495177

в) информационные электронно-образовательные ресурсы:

№ п/п	Источник
1	http://www.lib.vsu.ru ЗНБ ВГУ
2	https://edu.vsu.ru Образовательный портал "Электронный университет ВГУ"
3	https://biblioclub.ru ЭБС «Университетская библиотека online»

16. Перечень учебно-методического обеспечения для самостоятельной работы

№ п/п	Источник
1	Рекомендованная программой литература.
2	Конспекты лекций.
3	Комплект учебных приложений с подробными инструкциями по их сборке для первоначального освоения предлагаемых для выполнения лабораторных задач технологий.

17. Образовательные технологии, используемые при реализации учебной дисциплины, включая дистанционные образовательные технологии (ДОТ), электронное обучение (ЭО), смешанное обучение):

В качестве интегрированной среды разработки при проведении лабораторных работ студентам рекомендуется использование установленного в компьютерных классах факультета компьютерных наук лицензионного программного обеспечения Microsoft Visual Studio и свободно распространяемой системы программирования Borland Turbo-Delphi для реализации собственных программных разработок. В качестве учебного сервера баз данных рекомендуется использование сервера Oracle факультета компьютерных наук с учебной базой данных, разработанной для

проведения занятий по управлению данными. Освоение технологии OLE-автоматизации предполагает использование лицензионного программного обеспечения Microsoft Office, установленного на компьютерах факультетских лабораторий.

Внеаудиторная самостоятельная работа, как правило, предполагает использование студентами среды разработки Microsoft Visual Studio, официальная процедура установки которой на собственные компьютеры студентов обеспечена факультетом компьютерных наук, а также среды разработки Turbo-Delphi. Тем не менее, студенты никак не ограничиваются требованиями к их выбору средств разработки для использования в самостоятельной работе. Для имитации сервера баз данных в условиях самостоятельной работы студентам рекомендуется использование локальных таблиц с данными и соответствующих стандартных интерфейсов и компонентов доступа к ним.

При реализации дисциплины могут использоваться технологии электронного обучения и дистанционные образовательные технологии на базе портала edu.vsu.ru, а также другие доступные ресурсы сети Интернет.

18. Материально-техническое обеспечение дисциплины:

компьютерные классы факультета компьютерных наук, лицензионное программное обеспечение Microsoft Visual Studio, свободно распространяемая система программирования Borland Turbo-Delphi, учебный сервер Oracle факультета с базой данных, разработанной для проведения занятий по управлению данными, лицензионное программное обеспечение Microsoft Office.

19. Оценочные средства для проведения текущей и промежуточной аттестаций

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

№ п/п	Разделы дисциплины (модули)	Код компетенции	Код индикатора	Оценочные средства для текущей аттестации
1	Разделы 1 - 3, 14 Эволюция технологии доступа к данным в информационных системах. Понятие DCE (Distributed Computing Environment). Архитектура средств DCE OSF (Open Software Foundation). Концепция вызова удаленных процедур (Remote Procedure Call, RPC) на примере RPC DCE OSF. Механизм вызова RPC. Введение в компонентное программирование. Особенности разработки распределенных приложений для функционирования в среде Microsoft RPC.	ПК-3	ПК-3.2	Комплект задач №1

№ п/п	Разделы дисциплины (модули)	Код компетенции	Код индикатора	Оценочные средства для текущей аттестации
2	Разделы 5 – 7, 15 COM и объектно-ориентированный подход. COM-интерфейс: идентификация, спецификация, виды реализаций. Классы COM-объектов. Создание экземпляров объектов. Многопоточность в COM. Механизмы повторного использования COM-объектов. Маршalling - механизм взаимодействия сервера и клиента. Информация о типе. Библиотека типа. Начальные сведения о технологии OLE-автоматизации. Разработка простого контроллера автоматизации для одного из приложений Microsoft Office.	ПК-3	ПК-3.3	Комплект задач №2
3	Разделы 8 – 9, 16 Постоянные данные объектов. Сервисы перманентности. Управление перманентностью. Объекты с подключением - технология реализации событий COM. Распределенная модель компонентных объектов (Distributed COM). Введение в технологию Borland MIDAS (Multi-tier Distributed Application Services Suite) для разработки многозвенных распределенных приложений и их эксплуатации в корпоративных системах.	ПК-5	ПК-5.3	Комплект задач №3

Промежуточная аттестация

Форма контроля - Зачет с оценкой, Курсовая работа

Оценочные средства для промежуточной аттестации

По результатам трех текущих аттестаций

20 Типовые оценочные средства и методические материалы, определяющие процедуры оценивания

20.1 Текущий контроль успеваемости

Контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств.

20.1.1. Примерная структура тренировочных учебных примеров

1. Реализация тестового распределенного приложения и демонстрация его работы при различных способах размещения клиентской и серверной частей и различных последовательностях протоколов.
2. Реализация учебного примера контроллера автоматизации для одного из приложений Microsoft Office с использованием различных механизмов связывания: позднее

(динамическое), раннее (импорт информации о типе). Демонстрация его возможностей по формированию отчетов с использованием тестовой базы данных.

3. Реализация в технологии Borland MIDAS учебного распределенного приложения: сервер приложений с доступом к тестовой базе данных и "тонкий" клиент с интерфейсом к данным. Использование различных протоколов обмена клиента с сервером приложений, включая SOAP. В последнем случае сервер приложений реализуется в виде web-службы и размещается на отладочном web-сервере.

20.1.2. Курсовые работы

Выполнение курсовой работы является необходимым условием допуска студентов к итоговому зачёту с оценкой. Курсовая работа выполняется в форме реферата на определенную тему, связанную с тематикой лекционного материала. Предусмотрены выступления магистрантов с докладами по материалам наиболее удачных курсовых работ на практических занятиях. Расписание выступлений согласовывается с темами текущих лекций, что способствует углубленному изучению наиболее важных вопросов.

Примерный перечень курсовых работ:

1. COM, COM+, CORBA: объектная модель, общие свойства и особенности ее реализации в перечисленных средах распределенных вычислений.
2. Инкапсуляция, наследование (включение и агрегирование), полиморфизм COM. Общие свойства и отличия от объектно-ориентированных языков программирования.
3. RPC, COM, CORBA – механизмы связывания («раннее», «позднее», статическое, динамическое связывание, инструменты их обеспечения).
4. Основные задачи распределенной вычислительной среды и их решение в DCE OSF, COM и CORBA.
5. Место DCE OSF, COM и CORBA в 7-уровневой модели OSI-RM. Протоколы взаимодействия компонентов распределенного приложения, возможности управления нижележащими уровнями.
6. Отображения языков определения интерфейса (IDL) к языкам программирования для RPC, COM и CORBA. Инструментальные средства поддержки отображений.
7. Методы (виды интерфейсов), свойства (проблема перманентности), события (технология объектов с подключением), как составляющие реализации объектной модели COM.
8. Основные средства и технологии, обеспечившие переход от COM к распределенной вычислительной среде (DCOM), а также совместимость их на программном уровне.
9. Основные типы архитектур распределенных систем с базами данных. Этапы развития технологии доступа к данным в информационных системах.
10. Общая схема архитектуры средств DCE OSF, характеристика основных ее компонентов: нити, RPC, службы времени, каталогов и безопасности, распределенная файловая система.
11. Стратегия Microsoft в области распределенных вычислений: технологии DLL, VBX, WOSA, ODBC, OLE, COM, DCOM, ActiveX, MTS и MSMQ, COM+ и Windows DNA, .NET .
12. Три вида стандартного двоичного формата реализации интерфейса COM: возможности и ограничения каждого вида.
13. Механизмы повторного использования COM-объектов: включение (делегирование) и агрегирование, подробности взаимодействия внешнего и внутреннего объектов.
14. Особенности и общие черты механизмов сопровождения информации о типе (метаинформации) в COM, CORBA и .NET Framework: понятия библиотеки типа, репозитория интерфейсов, системы метаданных.
15. Этапы эволюции моделей программирования в контексте механизмов повторного использования кода в новых приложениях (процедурное, объектно-ориентированное, компонентное программирование).

Формулировка тем курсовых работ вышеприведенного списка рассчитана на необходимость представления в работе обобщающих выводов. Например, сравнительная характеристика трех объектных моделей (COM, COM+, CORBA) требует анализа как общих принципов, так и особенного в каждом случае, способствует высказыванию собственного отношения автора к преимуществам и недостаткам этих особенностей.

20.1.3. Перечень практических заданий

Индивидуальное задание №1 Разработка специфического распределенного приложения для функционирования в среде Microsoft RPC. Основная цель – изучение концепции вызова удаленных процедур (RPC), основ языка определения интерфейсов (IDL) и схемы сборки распределенного приложения. Изучение механизма вызова RPC и динамического связывания. Оценка «зачтено» выставляется, если разработанное согласно индивидуальному заданию распределенное приложение реализует соответствующую модель информационного взаимодействия программных компонентов, при этом приложение должно включать, как минимум, два таких компонента: клиентскую и серверную части, выполненные в виде консольных приложений, соединенных по сетевому протоколу и работающих на разных компьютерах локальной сети. Пример варианта индивидуального задания из комплекта задач №1:

Средствами MS RPC реализуйте модель информационной системы с параллельно работающими несколькими серверами. Клиент непосредственно после запуска производит поиск доступного для подключения сервера.

Индивидуальное задание №2 Разработка контроллера автоматизации для одного из приложений Microsoft Office. Основная цель – изучение возможностей технологии OLE-автоматизации в контексте использования для формирования отчетов по результатам выборки из базы данных. Оценка «зачтено» выставляется, если разработанная согласно индивидуальному заданию программа, выступая в качестве контроллера автоматизации, выполняет необходимые действия по управлению OLE-сервером для формирования отчета по результатам выборки из учебной базы данных. Пример варианта индивидуального задания из комплекта задач №2:

Разработайте контроллер автоматизации, использующий механизм позднего связывания одновременно с двумя серверами: Microsoft Word и Excel; для решения следующей задачи. Таблица в книге Excel должна быть скопирована в документ Word и соответствующим образом отформатирована. При желании можно комбинировать различные варианты механизмов связывания с серверами автоматизации.

Индивидуальное задание №3 Разработка трехзвенного распределенного приложения в технологии Borland MIDAS. Цель: изучение основ технологии Borland MIDAS (Multi-tier Distributed Application Services Suite) для разработки многосвязных распределенных приложений и их эксплуатации в корпоративных системах. Оценка «зачтено» выставляется, если разработанное согласно индивидуальному заданию приложение выполняет необходимые действия в роли модельной трехзвенной информационной системы: на студента ложится ответственность за разработку звеньев тонкого клиента и сервера приложений, слой сервера базы данных необходимо реализовать подключением к учебному факультетскому серверу, либо имитацией при помощи локальных таблиц в файлах. Пример варианта индивидуального задания из комплекта задач №3:

Доработайте учебный пример трехзвенной системы так, чтобы оператор «тонкого» клиента имел возможность удобной работы в автономном режиме (без связи с сервером приложений). При этом необходимо дополнительно предусмотреть интерфейсные команды управления соединением с сервером приложений, команды сохранения кэша редактируемых таблиц в файлах на локальных дисках, восстановления содержимого кэша из файлов, а также команды на передачу результатов редактирования таблиц серверу приложений для сохранения их в базе данных (при переходе из автономного режима в режим связи с сервером приложений).

20.1.4. Описание технологии проведения

Текущая аттестация проводится в соответствии с Положением о текущей аттестации обучающихся по программам высшего образования Воронежского государственного университета. Текущая аттестация проводится в формах: устного опроса (индивидуальный опрос, фронтальная беседа) по лекционному материалу, проверки знания основ предложенных технологий по результатам демонстрации тренировочных комплектов приложений и защиты лабораторных работ. При оценивании результатов устного опроса и демонстрации тренировочных комплектов приложений используется качественная шкала оценок. Оценивание результатов и защиты лабораторных работ предполагает использование количественной шкалы.

20.1.5. Требования к выполнению заданий (или шкалы и критерии оценивания)

Конкретные требования к выполнению каждого практического задания изложены в п.20.1.2. Для оценивания качества выполнения заданий используется – зачтено, не зачтено.

Приведённые ниже задания рекомендуется использовать при проведении диагностических работ для оценки остаточных знаний по дисциплине

1) тестовые задания - 1 балл

1. Каковы признаки распределенной системы (2 варианта)?
 - a. система состоит из отдельных слабо связанных сущностей (узлов/процессов/компьютеров), каждый из которых имеет собственную локальную память
 - b. система имеет только одно вычислительное устройство (процессор/узел)
 - c. система состоит из отдельных сущностей (узлов/процессов/компьютеров), соединенных посредством некоторой сети
 - d. система имеет одно адресное пространство для всех устройств, входящих в нее
2. В чем выражается гетерогенность распределенной системы?
 - a. пропускная способность всех связывающих узлы системы сетей одинакова
 - b. все узлы в системе одинаковы по конфигурации
 - c. производительность и архитектура различных сетей и узлов в системе могут отличаться друг от друга
 - d. узлы и сети системы отказоустойчивы
3. Выберите верные утверждения (2 варианта).
 - a. при удаленном вызове процедуры (RPC) выполнение кода процедуры распределяется динамически между клиентом и сервером
 - b. при удаленном вызове процедуры (RPC) выполнение кода процедуры происходит на стороне сервера
 - c. при удаленном вызове процедуры (RPC) клиент может получать от сервера код ошибки в случае некорректного выполнения процедуры
 - d. при удаленном вызове процедуры (RPC) клиент не должен предоставлять значения аргументов для выполняемой процедуры
4. Что такое идемпотентный метод?
 - a. метод на клиентской стороне в реализации RMI
 - b. метод, возвращающий один и тот же результат на идентичные вызовы
 - c. XML-RPC
 - d. метод удаленного объекта, который может обмениваться с другими объектами информацией по бинарному протоколу
5. В чем отличие удаленных вызовов от локальных (2 варианта)?

- a. удаленные вызовы "непрозрачны", то есть клиент не владеет полной информацией о внутреннем устройстве сервера
 - b. удаленные вызовы имеют меньшую латентность, чем локальные
 - c. ошибки при удаленном вызове могут быть не связаны с ошибками в программной реализации самих методов
 - d. устройство сети никак не влияет на производительность RPC
6. Набор стандартов "архитектуры управления объектами" обозначается:
- a. OMA
 - b. AMO
 - c. MOA
7. Среда выполнения, которая реализует спецификацию CORBA, называется:
- a. компонентом
 - b. брокером
 - c. объектом
8. Из перечисленного CORBA имеет способы определения интерфейса объекта (2 варианта):
- a. временный
 - b. динамический
 - c. статический
 - d. постоянный
9. Языком объектных моделей OMA является язык
- a. IDL
 - b. DLL
 - c. ODL
10. Какого из основных стандартов промежуточного ПО для поддержки распределенных объектных вычислений не существует?
- a. DLINK
 - b. CORBA
 - c. DCOM
11. Что такое middleware
- a. персональное подпространство пользователя
 - b. средство компьютерной безопасности
 - c. программное обеспечение промежуточного уровня
12. Как называется последовательность взаимосвязанных действий, которые должны быть выполнены на нескольких серверах распределенной системы
- a. транзакция
 - b. интеграция
 - c. операция
13. Какие из перечисленных сред распределенных вычислений предоставляют объектно-ориентированную модель программирования?
- a. DCE OSF
 - b. DCOM
 - c. CORBA
 - d. .NET

14. Какие из перечисленных сред распределенных вычислений предоставляют процедурно-ориентированную модель программирования?
- a. DCE OSF
 - b. DCOM
 - c. CORBA
 - d. .NET
15. Отложенная регистрация протоколов в DCOM означает:
- a. объект регистрирует в системе протокол, по которому готов работать
 - b. объект загружает необходимый для протокола код, лишь когда клиент начинает работать с ним по данному протоколу
 - c. объект задерживает по времени начало прослушивания по данному протоколу
16. Хранилище информации о типе в COM –
- a. репозиторий реализаций
 - b. библиотека типа
 - c. сборка
 - d. репозиторий интерфейсов
 - e. интерфейс динамического вызова
17. Хранилище информации о типе в CORBA –
- a. репозиторий реализаций
 - b. библиотека типа
 - c. сборка
 - d. репозиторий интерфейсов
 - e. интерфейс динамического вызова
18. Статический вызов объекта DCOM производится через (2 варианта):
- a. клиентский стаб
 - b. серверный стаб
 - c. скелетон реализации
 - d. прокси-объект
19. Вызов удаленной процедуры RPC DCE OCF производится через (2 варианта):
- a. клиентский стаб
 - b. серверный стаб
 - c. скелетон реализации
 - d. прокси-объект
20. Статический вызов объекта CORBA производится через (2 варианта):
- a. клиентский стаб
 - b. серверный стаб
 - c. скелетон реализации
 - d. прокси-объект

Ответы на вопросы

Номер вопроса	Ответ (буква)
---------------	---------------

1.	a,c
2.	c
3.	b,c
4.	b
5.	a,c
6.	a
7.	b
8.	b,c
9.	a
10.	a
11.	c
12.	a
13.	b,c,d
14.	a
15.	b
16.	b
17.	d
18.	a,d
19.	a,b
20.	a,c

2) задания с коротким ответом - 2 балла

21. Идентификатор базового COM-интерфейса: ...

22. Аббревиатура, обозначающая глобально уникальный идентификатор: ...

23. Идентификатор COM-интерфейса для доступа к диспетчерскому интерфейсу: ...

24. Аббревиатура, обозначающая глобально уникальный идентификатор класса COM: ...

25. Аббревиатура, обозначающая глобально уникальный идентификатор COM-интерфейса: ...

Ответы на вопросы

Номер вопроса	Ответ (буква)
21.	IUnknown
22.	GUID
23.	IDispatch
24.	CLSID
25.	IID

3) задания с развернутым ответом - 3 балла

26. Опишите механизм, обеспечивающий прозрачность вызова удаленной процедуры в рамках распределенной вычислительной среды RPC DCE OSF и Microsoft RPC.

Ответы на вопросы

Номер вопроса	Ответ (буква)
26.	Программный код клиента и сервера включает пару заглушек (stub), взаимодействующих по сети. Вызовы процедуры из программы клиента прозрачно перенаправляются клиентской заглушке. Вызовы процедуры на стороне сервера выполняет серверная заглушка.

Критерии оценивания	Шкала оценок
Обучающийся приводит полное и безошибочное описание механизма, обеспечивающего прозрачность вызова удаленной процедуры в рамках распределенной вычислительной среды.	3 балла
Обучающийся приводит описание механизма, обеспечивающего прозрачность вызова удаленной процедуры, в котором допущены незначительные неточности, и верно формулирует принцип его функционирования.	2 балла
Представлено словесное описание идеи обеспечения прозрачности вызова удаленной процедуры, правильно отражающее основные принципы реализации механизма.	1 балл

В представленном ответе нет упоминания о заглушках (или стабах), не описан принцип прозрачного взаимодействия частей распределенного приложения.	0 баллов
--	----------

27. Перечислите задачи, выполняемые распределенной инфраструктурой разрешателей OXID в среде DCOM.

Ответы на вопросы

Номер вопроса	Ответ (буква)
27.	Отображение OXID (идентификатора экспортера объектов) в строку связывания; обеспечение отложенной регистрации протоколов объектами; группирование тестового опроса объектов клиентами.

Критерии оценивания	Шкала оценок
Обучающийся приводит полное и безошибочное описание всех трех задач, выполняемых распределенной инфраструктурой разрешателей OXID.	3 балла
Обучающийся приводит описание задач, выполняемых распределенной инфраструктурой разрешателей OXID, в котором допущены незначительные неточности, но верно формулирует цель постановки этих задач.	2 балла
Представлено неполное описание задач, выполняемых распределенной инфраструктурой разрешателей OXID, либо перечислены не все задачи.	1 балл
В представленном ответе нет перечисления задач, выполняемых распределенной инфраструктурой разрешателей OXID, отсутствует определение используемых в вопросе понятий.	0 баллов

28. Опишите принцип действия механизма подсчета ссылок объектами и особенности его реализации в распределенной вычислительной среде.

Ответы на вопросы

Номер вопроса	Ответ (буква)
---------------	---------------

28.	На примере COM: счетчик ссылок размещается в объекте и управляется явными вызовами методов базового интерфейса IUnknown. Нулевое значение счетчика служит для объекта сигналом возможности безопасного завершения работы. В распределенной среде механизм дополнен тестовым опросом из клиента каждого интерфейса для определения факта аварийного завершения работы сервера.
-----	---

Критерии оценивания	Шкала оценок
Обучающийся приводит полное и безошибочное описание механизма подсчета ссылок объектами с уточнением особенностей его реализации в распределенной вычислительной среде.	3 балла
Обучающийся приводит описание механизма подсчета ссылок объектами, в котором допущены незначительные неточности, но верно формулирует принцип его функционирования.	2 балла
Представлено неполное описание механизма подсчета ссылок объектами, либо не сформулированы особенности его реализации в распределенной вычислительной среде.	1 балл
В представленном ответе нет описания механизма подсчета ссылок, отсутствует определение используемых в вопросе понятий.	0 баллов

29. Опишите способ использования информации о типе объекта для реализации статического вызова методов его интерфейса в распределенной вычислительной среде.

Ответы на вопросы

Номер вопроса	Ответ (буква)
29.	Средство разработки включает процедуру импорта типа, используя которую можно получить описание объекта на языке реализации. По этому описанию при сборке проекта генерируется код заглушек, обеспечивающий прозрачность взаимодействия в распределенной среде.

Критерии оценивания	Шкала оценок
---------------------	--------------

Обучающийся приводит полное и безошибочное описание способа использования информации о типе объекта для реализации статического вызова методов его интерфейса в распределенной вычислительной среде.	3 балла
Обучающийся приводит описание способа использования информации о типе объекта, в котором допущены незначительные неточности, но верно формулирует цель его применения.	2 балла
Представлено неполное описание способа использования информации о типе объекта, либо не сформулированы цели его применения.	1 балл
В представленном ответе нет описания способа использования информации о типе объекта, отсутствует определение используемых в вопросе понятий.	0 баллов

30. Опишите технологию объектов с подключением (Connectable Objects) и ее основное назначение в модели COM.

Ответы на вопросы

Номер вопроса	Ответ (буква)
30.	Технология реализует механизм событий модели COM. Основывается на обратных вызовах клиента со стороны исходящих интерфейсов объектов. Клиент при таком вызове реализует отклик на событие. Описание исходящих интерфейсов включается в состав информации о типе объекта.

Критерии оценивания	Шкала оценок
Обучающийся приводит полное и безошибочное описание технологии объектов с подключением и ее основного назначения в модели COM.	3 балла
Обучающийся приводит описание технологии объектов с подключением, в котором допущены незначительные неточности, но верно формулирует цель ее применения.	2 балла
Представлено неполное описание технологии объектов с подключением, либо не сформулированы цели ее применения.	1 балл
В представленном ответе нет описания технологии объектов с подключением, отсутствует определение используемых в вопросе понятий.	0 баллов

20.2 Промежуточная аттестация

Промежуточная аттестация проводится в соответствии с Положением о промежуточной аттестации обучающихся по программам высшего образования. Промежуточная аттестация по итогам освоения дисциплины проводится в форме зачета. Условиями для положительной итоговой оценки являются: выполнение всех тренировочных комплектов приложений и успешные защиты лабораторных работ, перечисленных в приложении под наименованием «индивидуальное задание».

20.2.1. Шкалы и критерии оценивания

При оценивании используется следующая шкала:

5 баллов ставится, если обучающийся демонстрирует полное соответствие знаний, умений, навыков приведенным в таблицах показателям, свободно оперирует приобретенными знаниями, умениями, применяет их при решении практических задач;

4 балла ставится, если обучающийся демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателям, но допускает незначительные ошибки, неточности, испытывает затруднения при решении практических задач;

3 балла ставится, если обучающийся демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателям, допускает значительные ошибки при решении практических задач;

2 балла ставится, если обучающийся демонстрирует явное несоответствие знаний, умений, навыков приведенным в таблицах показателям.

При сдаче зачета с оценкой

оценка «отлично» - 5 баллов

оценка «хорошо» - 4 балла

оценка «удовлетворительно» - 3 балла

оценка «неудовлетворительно» - 2 балла.

Для оценивания результатов обучения на зачете с оценкой используются следующие показатели:

1. знание основных принципов организации распределенных приложений, вычислительных и операционных систем; наиболее употребительных моделей распределенного программирования, в том числе, процедурной, компонентной и Grid, принципов функционирования соответствующей инфраструктуры их поддержки в гетерогенной среде, стандартов на последовательности протоколов разных уровней и архитектуру систем программирования для реализации распределенных вычислений;
2. умение применять перечисленные сведения для проектирования систем распределенных вычислений;
3. владение навыками реализации распределенных приложений с использованием широко распространенных технологий и моделей программирования.

Соотношение показателей, критериев и шкалы оценивания результатов обучения.

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок
---------------------------------	--------------------------------------	--------------

<p>Сформированные знания об основных принципах организации распределенных приложений, вычислительных и операционных систем; наиболее употребительных моделях распределенного программирования, в том числе, процедурной, компонентной и Grid, принципов функционирования соответствующей инфраструктуры их поддержки в гетерогенной среде, стандартов на последовательности протоколов разных уровней и архитектуру систем программирования для реализации распределенных вычислений. Сформированное умение применять перечисленные сведения для проектирования систем распределенных вычислений. Сформированные навыки реализации распределенных приложений с использованием широко распространенных технологий и моделей программирования.</p>	<p>Повышенный уровень</p>	<p>Отлично</p>
<p>Сформированные, но содержащие отдельные пробелы представления об основных принципах организации распределенных приложений, вычислительных и операционных систем; наиболее употребительных моделях распределенного программирования, в том числе, процедурной, компонентной и Grid, принципах функционирования соответствующей инфраструктуры их поддержки в гетерогенной среде, стандартах на последовательности протоколов разных уровней и архитектуру систем программирования для реализации распределенных вычислений. Успешное, но содержащее отдельные пробелы умение применять перечисленные сведения для проектирования систем распределенных вычислений. Сформированные, но содержащие отдельные пробелы навыки реализации распределенных приложений с использованием широко распространенных технологий и моделей программирования.</p>	<p>Базовый уровень</p>	<p>Хорошо</p>
<p>Неполное представление об основных принципах организации распределенных приложений, вычислительных и операционных систем; наиболее употребительных моделях распределенного программирования, в том числе, процедурной, компонентной и Grid, принципах функционирования соответствующей инфраструктуры их поддержки в гетерогенной среде, стандартах на последовательности протоколов разных уровней и архитектуру систем программирования для реализации распределенных вычислений. Успешное, но не системное умение применять перечисленные сведения для проектирования систем распределенных вычислений. Неполное владение навыками реализации распределенных приложений с использованием широко распространенных технологий и моделей программирования средств.</p>	<p>Пороговый уровень</p>	<p>Удовлетворительно</p>
<p>Фрагментарные знания или отсутствие знаний. Фрагментарные умения или отсутствие умений. Фрагментарные навыки или отсутствие навыков.</p>	<p>-</p>	<p>Неудовлетворительно</p>

