

Балаковский инженерно-технологический институт – филиал федерального
государственного автономного образовательного учреждения высшего образования
«Национальный исследовательский ядерный университет «МИФИ»

Факультет атомной энергетики и технологий
Кафедра «Информационные системы и технологии»

РАБОЧАЯ ПРОГРАММА

по дисциплине «Современные технологии разработки распределенного
программного обеспечения»

Направления подготовки

«09.03.02 Информационные системы и технологии»

Основная профессиональная образовательная программа

«Информационные системы и технологии»

Квалификация выпускника

Бакалавр

Форма обучения

Заочная

Цель освоения учебной дисциплины

Целями освоения учебной дисциплины в области обучения, воспитания, развития, соотносённые с общими целями ООП ВО и требованиями профессиональных стандартов «Руководитель проектов в области информационных технологий», «Специалист по тестированию в области информационных технологий»: ознакомление студентов с понятием распределённых систем; понимание задач распределённой системы; изучение концепций аппаратных и программных решений распределённых систем; обзор клиент-серверных моделей обработки данных.

Место учебной дисциплины в структуре ООП ВО

Необходимыми условиями для освоения дисциплины являются знания, умения и практические навыки по предшествующим дисциплинам и практикам в соответствии с требованиями освоения компетенций:

Администрирование информационных систем / Методы администрирования современных систем управления базами данных

Знания, умения и практические навыки, полученные при освоении дисциплины, необходимы при изучении следующих дисциплин:

Инфокоммуникационные системы и сети
а также государственной итоговой аттестации

При освоении данной дисциплины студент сможет частично продемонстрировать следующие трудовые функции: Разработка документов для тестирования ПО и анализа качества тестового покрытия; Обеспечение качества в проектах в области ИТ в соответствии с установленными регламентами.

Компетенции обучающегося, формируемые в результате освоения дисциплины

общепрофессиональные

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции
ОПК-2	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	З-ОПК-2 Знать принципы функционирования и применения современных информационных технологий У-ОПК-2 Уметь применять информационные технологии для решения профессиональных задач В-ОПК-2 Владеть навыками использования современных информационных технологий и программными средствами, в том числе отечественного производства, применять их для решения задач профессиональной деятельности

профессиональные

Задачи профессиональной деятельности (ЗПД)	Объект или область знания	Код и наименование компетенции	Индикаторы достижения компетенции
проектирование базовых и прикладных информационных технологий	информационные процессы, технологии, системы и сети, их инструментальное (программное, техническое, организационное) обеспечение, способы и методы проектирования, отладки, производства и эксплуатации информационных технологий и систем в различных областях и сферах деятельности.	ПК-15 Способен обеспечивать качество объекта проектирования при разработке и вводе в эксплуатацию ИС	З-ПК-15 Знать: методологии проектирования и тестирования информационных систем; требования к обеспечению информационной безопасности У-ПК-15 Уметь: проводить тестирование ИС и анализировать результаты тестирования; обеспечивать безопасность разрабатываемых приложений В-ПК-15 Владеть: средствами отладки программного кода

Задачи воспитания, реализуемые в рамках освоения дисциплины

Направление/цели	Создание условий, обеспечивающих	Использование воспитательного потенциала учебной дисциплины	Вовлечение в разноплановую внеучебную деятельность
Профессиональное и трудовое воспитание	формирование культуры исследовательской и инженерной деятельности (В16)	Использование воспитательного потенциала дисциплин профессионального модуля для формирования инженерного мышления и инженерной культуры за счёт практических студенческих исследований современных производственных систем; проектной деятельности студентов по разработке и оптимизации технологических систем, связанной с решением реальных производственных задач; прохождения через разнообразные игровые, творческие, фестивальные формы, требующие анализа сложного объекта, постановки относительно него преобразовательных задач для их оптимального решения.	1. Организация научно-практических конференций и встреч с ведущими специалистами предприятий города и ветеранами атомной отрасли. 2. Организация и проведение предметных олимпиад и участие в конкурсах профессионального мастерства.

Структура и содержание учебной дисциплины

Дисциплина преподается студентам в 9-ом семестре. Общая трудоемкость дисциплины составляет 3 зачетных единицы, 108 ак. часов.

Календарный план

№ Р а з д е л а	№ Т е м ы	Наименование раздела (темы) дисциплины	Виды учебной деятельности (в часах)					Аттестация раздела (форма *)	Максимальный балл за раздел **
			Всего	Лекции	Лабораторные	Практические	СРС/КРС		
1	1-4	Основные понятия распределенных систем	50	2		4	44	Т	20
2	5-9	Связь в распределенных системах	54	4/2		4/2	46	Т	50
Вид промежуточной аттестации			4					З	30
			108/4	6/2		8/2	90		100

* - сокращенное наименование формы контроля

** - сумма максимальных баллов должна быть равна 100 за семестр, включая зачет и (или) экзамен
Сокращенное наименование форм текущего контроля и аттестации разделов:

Обозначение	Полное наименование
Т	Тестирование
З	Зачет

Содержание лекционного курса

Темы лекции. Вопросы, отрабатываемые на лекции	Всего часов	Учебно-методическое обеспечение
Определение и задачи распределенных систем. Соединение пользователей с ресурсами. Прозрачность. Открытость. Масштабируемость.	1	1-2
Концепции аппаратных решений. Мультипроцессоры. Гомогенные мультимикомпьютерные системы. Гетерогенные мультимикомпьютерные системы.	1	1-2

Концепции программных решений. Распределенные операционные системы. Сетевые операционные системы. Программное обеспечение промежуточного уровня.	-	1-2
Модель клиент-сервер. Клиенты и серверы. Разделение приложений по уровням. Варианты архитектуры клиент-сервер.	-	1-2
Уровни протоколов. Низкоуровневые протоколы. Транспортные протоколы. Протоколы верхнего уровня.	1	1-2
Удаленный вызов процедур. Базовые операции RPC. Передача параметров. Расширенные модели RPC. DCE RPC.	1	1-2
Обращение к удаленным объектам. Распределенные объекты. Привязка клиента к объекту. Статическое и динамическое удаленное обращение к методам. Передача параметров. Удаленные объекты DCE. Java RMI.	1	1-2
Связь посредством сообщений. Сохранность и синхронность во взаимодействиях. Нерезидентная связь на основе сообщений. Сохранная связь на основе сообщений. IBM MQSeries.	1	1-2
Связь на основе потоков данных. Поддержка непрерывных сред. Поток данных и качество обслуживания. Синхронизация потоков данных.	-	1-2

Перечень практических занятий

Тема практического занятия. Вопросы, отрабатываемые на практическом занятии	Всего часов	Учебно-методическое обеспечение
Клиент-серверная архитектура java-приложений.	2	1-2
Разработка распределенного приложения с удаленным вызовом процедур RPC.	2	1-2
Разработка распределенного приложения с обращением к удаленным объектам RMI.	2	1-2
Разработка распределенного приложения, ориентированного на сообщения промежуточного уровня MOM.	2	1-2
Разработка распределенного приложения с использованием потоков данных.	4	1-2

Перечень лабораторных работ – не предусмотрен учебным планом

Задания для самостоятельной работы студентов

Вопросы для самостоятельного изучения (задания)	Всего часов	Учебно-методическое обеспечение
Алгоритмы синхронизации в распределенных системах	6	1-2
Алгоритмы голосования в распределенных системах	6	1-2
Взаимное исключение в распределенных системах	6	1-2
Распределенные транзакции	6	1-2
Модели непротиворечивости, ориентированные на данные	6	1-2
Модели непротиворечивости, ориентированные на клиента	6	1-2
Протоколы распределения	6	1-2
Протоколы непротиворечивости в распределенных системах	6	1-2
Понятие отказоустойчивости в распределенных системах	6	1-2
Надежная групповая рассылка в распределенных системах	6	1-2
Защищенные каналы в распределенных системах	6	1-2
Контроль доступа в распределенных системах	6	1-2
Общая архитектура брокера объектных запросов CORBA	3	1-2
Поддержка связи между объектами DCOM	3	1-2
Распределенные системы объектов Globe	3	1-2
Сетевая файловая система компании Sun	3	1-2
Файловая система Coda	2	1-2
Распределенная система согласования Jini	2	1-2

Расчетно-графическая работа не предусмотрена учебным планом

Курсовая работа не предусмотрена учебным планом

Образовательные технологии

При реализации учебного материала курса используются различные образовательные технологии, способствующие созданию атмосферы свободной и творческой дискуссии как между преподавателем и студентами, так и в студенческой группе. Целью при этом является выработка у студентов навыков и компетенций, позволяющих самостоятельно вести исследовательскую и научно-педагогическую работу.

Аудиторные занятия проводятся в виде лекций, практических занятий с использованием ПК и компьютерного проектора. Самостоятельная работа студентов проводится под руководством преподавателей, с оказанием консультаций и помощи при подготовке к практическим занятиям.

Фонд оценочных средств

Фонд оценочных средств по дисциплине обеспечивает проверку освоения планируемых результатов обучения (компетенций и их индикаторов) посредством мероприятий текущего, рубежного и промежуточного контроля по дисциплине.

Связь между формируемыми компетенциями и формами контроля их освоения представлена в следующей таблице:

№ п/п	Наименование контролируемых разделов (темы)	Код и наименование индикатора достижения компетенций	Наименование оценочного средства
Входной контроль			
1	Входной контроль		Вопросы входного контроля (письменно)
Аттестация разделов, текущий контроль успеваемости			
2	Основные понятия распределенных систем	З-ОПК-2, У- ОПК-2, В- ОПК-2, З-ПК-15, У-ПК-15, В-ПК-15	Тестирование (письменно)
3	Связь в распределенных системах	З-ОПК-2, У- ОПК-2, В- ОПК-2, З-ПК-15, У-ПК-15, В-ПК-15	Тестирование (письменно)
Промежуточная аттестация			
4	Зачет	З-ОПК-2, У-ОПК-2, В-ОПК-2, З-ПК-15, У-ПК-15, В-ПК-15	Вопросы к зачету (устно)

Входной контроль предназначен для выявления пробелов в знаниях студентов и готовности их к получению новых знаний. Оценочные средства для входного контроля представляют собой вопросы, которые задаются студентам в устной форме.

Перечень вопросов входного контроля

1. Перечислите области человеческой деятельности, где используются программные продукты.
2. Что такое программная система?
3. Что такое жизненный цикл программного обеспечения?
4. Перечислите этапы жизненного цикла программного обеспечения.
5. Перечислите основные подходы программирования.
6. Опишите структурный подход программирования.
7. Опишите процедурный подход программирования.
8. Опишите объектно-ориентированный подход программирования.
9. Дайте определение процедуры и функции.
10. Дайте определения программирования.
11. Дайте определение структуры данных.
12. Как Вы понимаете понятие «интерфейс».
13. Что такое поток управления и поток данных?
14. Что такое объект? Что может быть объектом?
15. Что такое класс? Чем отличается класс от объекта?
16. Чем характеризуется класс?

17. Что такое состояние класса?
18. Что такое поведение класса?
19. Перечислите основные типы операций.
20. Что такое «отношения между классами»?
21. Что такое модель и моделирование?
22. Что такое «классификация»?
23. Что такое качество программного обеспечения с Вашей точки зрения?
24. Что такое тестирование программных продуктов?

Аттестация раздела по дисциплине проводится в форме тестирования. Тест содержит от 10 вопросов. На выполнение задания отводится 30 минут. Тест – это форма контроля, направленная на проверку уровня освоения контролируемого теоретического и практического материала. Примерный перечень тестовых заданий:

Вариант 1

№	вопрос
1	Какой единственный компонент является необязательным в корпоративном EJB-приложении? 1) Класс реализации EJB 2) Бизнес-интерфейс EJB 3) Дескриптор развертывания 4) Классы первичных ключей сущностей (entities)
2	Какой единственный из представленных интерфейсов должен быть реализован классом JMS MDB? 1) javax.ejb.MDB 2) javax.jms.MessageListener 3) javax.ejb.MessageDriven 4) javax.jms.MessageDrivenBean
3	Выберите два сервиса или API, которые контейнер обязан предоставлять разработчику EJB-компонентов (несколько вариантов): 1) JTA 2) JSP 3) JPA 4) Servlets
4	Выберите единственное верное высказывание о MDB: 1) Экземпляр MDB будет обрабатывать сообщения из JMS-очереди (queue) в порядке появления их в очереди 2) Все экземпляры MDB получают все сообщения из очереди 3) Только один поток выполнения (thread) будет выполнять метод обработки сообщений у экземпляра MDB 4) Разработчик должен позаботиться о сигнализации о завершении обработки (acknowledgement) сообщения в случае программно-управляемых транзакций (bean-managed transactions)
5	Выберите единственный сервис или API, который контейнер не обязан предоставлять разработчику EJB-компонентов: 1) JDBC 2) Web Services 3) JMS 4) JTS
6	Клиент посылает сообщение, обрабатываемое MDB. Какое единственное из приведенных высказываний верное? 1) Клиент ничего не знает об MDB, который будет обрабатывать его запрос 2) Клиент напрямую вызывает MDB с уточнением, что вызов должен быть обработан асинхронно 3) Клиент воспользуется специальной фабрикой (factory) MDB для получения нужного объекта

	4) Клиент получает бизнес-интерфейс MDB из сервиса именованного и вызывает методы
7	<p>Клиент получает ссылку на EJB Session Object. Какое единственное высказывание, касающееся жизненного цикла EJB, из приведенных верно?</p> <p>1) Клиент может удалить экземпляр (bean instance) stateless session EJB, вызвав метод у session object. При этом клиент не может таким же образом удалить экземпляр stateful session EJB</p> <p>2) Клиент может удалить экземпляр stateful session EJB, вызвав метод у session object. При этом клиент не может таким же образом удалить экземпляр stateless session EJB</p> <p>3) Клиент может удалить экземпляр stateless session EJB, вызвав метод у session object. При этом клиент может таким же образом удалить экземпляр stateful session EJB</p> <p>4) Клиент не может удалить экземпляр stateless session EJB, вызвав метод у session object. При этом клиент так же не может таким же образом удалить экземпляр stateful session EJB</p>
8	<p>Выберите единственную верную комбинацию высказываний из приведенных.</p> <p>Высказывание 1: Клиент посылает сообщение MDB, получив его бизнес-интерфейс из сервиса именованного JNDI</p> <p>Высказывание 2: Клиент посылает сообщение в JMS-хранилище (destination), ссылку на которое он получает из сервиса именованного JNDI:</p> <p>1) Оба высказывания неверны</p> <p>2) Высказывание 1 — верное, Высказывание 2 — неверное</p> <p>3) Высказывание 1 — неверное, Высказывание 2 — верное</p> <p>4) Оба высказывания верны</p>
9	<p>Клиент получает ссылку на EJB Session Object. Какое единственное высказывание из приведенных верно?</p> <p>1) EJB Session Object должен реализовать бизнес-интерфейс</p> <p>2) EJB Session Object должен реализовать интерфейс javax.ejb.SessionBean</p> <p>3) EJB Session Object не должен реализовать какой-либо интерфейс</p> <p>4) EJB Session Object должен реализовать интерфейс javax.ejb.SessionSynchronization</p>
10	<p>Выберите единственно верный порядок вызовов методов у Stateful Session EJB:</p> <p>1) @PostConstruct, @PostActivate, @PrePassivate</p> <p>2) @PostConstruct, @PrePassivate, @PostActivate</p> <p>3) @PostActivate, @PrePassivate, @PostConstruct</p> <p>4) @PostActivate, @PostConstruct, @PrePassivate</p>
11	<p>DriverManager это - :</p> <p>1) Объект, представляет собой соединение с БД</p> <p>2) Класс - средство тестирования драйверов, если тестирование прошло успешно, то с большой долей вероятности можно говорить о том, что JDBC драйверы будут "правильно" работать с программой</p> <p>3) Класс, уровень управления JDBC, его основная функция присоединение приложения к БД</p> <p>4) объект, уникальный идентификатор для подключения к БД. Состоит из типа драйвера и местонахождения ресурса</p>
12	<p>Что такое JDBC?</p> <p>1) GUI для выполнения запросов к БД</p> <p>2) java программа для взаимодействия БД с пользователем</p> <p>3) API для выполнения SQL запросов к БД</p> <p>4) web-service, посредник между БД и пользователем</p>
13	<p>В таблице есть колонка типа LONGVARCHAR, содержащая значения размером 20 гигабайт и больше. Какими способами из перечисленных можно считать такие значения (несколько вариантов)?</p> <p>1) ResultSet.getAsciiStream</p> <p>2) ResultSet.getString</p> <p>3) ResultSet.getInputSream</p>

	4) <code>ResultSet.getUnicodeStream</code>
14	<p>Выберите состояние прокручиваемости по умолчанию объекта класса <code>ResultSet</code> (прохождение по его содержимому, как можно двигать курсор?):</p> <ol style="list-style-type: none"> 1) только вперед 2) вперед и назад 3) только назад 4) нельзя двигать курсор
15	<p>Выберите правильный синтаксис JDBC для вызова хранимой процедуры. ([] - все что находится внутри, можно не указывать) (несколько вариантов):</p> <ol style="list-style-type: none"> 1) <code>{ call procedure_name[(? , ? , ...)] }</code> 2) <code>{ call procedure_name (? , ? , ...) }</code> 3) <code>{ ? = call procedure_name (? , ? , ...) }</code> 4) <code>{ ? = call procedure_name (? , ? , ...) }</code>
16	<p>Что из перечисленного является типом драйвера JDBC (несколько вариантов)?</p> <ol style="list-style-type: none"> 1) Драйверы протокола UDP 1.09 2) JDBC-NET (Сетевые драйвера на чистом Java) 3) Мост JDBC-ODBC 4) Драйверы родного протокола СУБД
17	<p>Какой интерфейс пакета <code>java.sql</code> используется, когда запрос к источнику данных является обращением к хранимой процедуре?</p> <ol style="list-style-type: none"> 1) <code>Statement</code> 2) <code>PreparedStatement</code> 3) <code>StoredStatement</code> 4) <code>CallableStatement</code>
18	<p>Каким методом можно получить информацию о колонках <code>ResultSet</code>?</p> <ol style="list-style-type: none"> 1) <code>getColumnData()</code> 2) <code>getColumnType()</code> 3) <code>getMetaData()</code> 4) <code>getCommonColumnType()</code>
19	<p>Сколько уровней изоляции транзакций определено в <code>java.sql.Connection</code>?</p> <ol style="list-style-type: none"> 1) 6 2) 5 3) 3 4) 2
20	<p>Как известно, некоторые СУБД не поддерживают хранимые процедуры и внешние соединения, также может различаться формат хранения дат. Каким образом JDBC решает проблему несоответствия языку SQL (несколько вариантов)?</p> <ol style="list-style-type: none"> 1) С помощью "escape" последовательностей 2) С помощью интерфейса <code>DatabaseMetaData</code> 3) С помощью компонента <code>DriverManager</code> 4) Через инвалидацию сессий
21	<p>Что из себя представляет тип <code>TIMESTAMP</code>?</p> <ol style="list-style-type: none"> 1) тип <code>TIME</code> + тип <code>STAMP</code> 2) тип <code>DATE</code> + тип <code>TIME</code> + тип <code>STAMP</code> 3) тип <code>DATE</code> + тип <code>TIME</code> + миллисекунды 4) тип <code>DATE</code> + тип <code>TIME</code> + наносекунды
22	<p>Сколько входных параметров могут иметь экземпляры интерфейса <code>PreparedStatement</code> (несколько вариантов)?</p> <ol style="list-style-type: none"> 1) 0 2) 1 3) >1 4) максимум 8

Вариант 2

№	вопрос
1	Компоненты <code>Stateful Session EJB</code> работают под большой нагрузкой. Для повышения

	<p>масштабируемости было принято решение использовать механизм пассивизации/активизации. Какое единственное высказывание верно?</p> <p>1) Метод, проаннотированный @PrePassivate, будет вызван однозначно чаще, чем метод, проаннотированный @PostActivate</p> <p>2) Метод, проаннотированный @PostPassivate, будет вызван однозначно чаще, чем метод, проаннотированный @PreActivate</p> <p>3) Методы, проаннотированные @PostPassivate и @PreActivate, будут вызваны строго равное число раз</p> <p>4) Ничего из перечисленного не верно</p>
2	<p>Приложение успешно собирается, но не деплоится на сервере. Как можно изменить следующий код для исправления ошибки (несколько вариантов)?</p> <pre>@Named @SessionScope public class MyBackingBean { //..... }</pre> <p>1) @Named @SessionScope</p> <pre>public class MyBackingBean implements Serializable { //..... }</pre> <p>2) @Named @SessionScope</p> <pre>public static final class MyBackingBean { //..... }</pre> <p>3) @Named @SessionScope</p> <pre>public final class MyBackingBean { //..... }</pre> <p>4) @Named @RequestScope</p> <pre>public class MyBackingBean { //..... }</pre>
3	<p>Разработчик хочет уничтожить экземпляр (bean instance) Stateless Session EJB. Он может выполнить:</p> <p>(1) Метод, проаннотированный @PreDestroy</p> <p>(2) Метод, проаннотированный @Remove:</p> <p>1) Только (2)</p> <p>2) (1), (2)</p> <p>3) (2), (1)</p> <p>4) (1) и (2) — неверные методы для этого</p>
4	<p>Какое утверждение о entity manager является верным?</p> <p>1) Entity manager, полученный путём внедрения ресурса в stateful session бин может использовать resource-local EntityTransaction для управления транзакциями</p> <p>2) Entity manager внедрённый в сессионные бины может использовать как JTA так и resource-local контроль транзакций</p> <p>3) Entity manager управляемый контейнером должен использовать JTA</p> <p>4) Entity manager, созданный вызовом метода EntityManagerFactory.createEntityManager всегда использует контроль транзакций JTA</p>
5	<p>Выберите два правильных высказывания с точки зрения разработчика EJB:</p> <p>1) EJB не должен передавать другим компонентам приложения ссылку на себя в виде ссылки this</p>

	<p>2) Класс реализации EJB не должен наследоваться от других java-классов</p> <p>3) Компонентам EJB следует использовать статические переменные для обмена данными между экземплярами (bean instances)</p> <p>4) Компоненты EJB не должны создавать новые потоки выполнения (threads)</p>
6	<p>Дан следующий код:</p> <pre> 10. @Stateless 11. @RunAs("X") 12. public class SecureBean01 implements Secure01 { 13. @EJB Secure02 secure02; 23. @RolesAllowed("A") 24. public void methodA() { 25. secure02.methodB(); 26. } 10. @Stateless 11. public void SecureBean02 implements Secure02 { 23. @RolesAllowed("A") 24. public void methodB() { </pre> <p>Пользователь с ролью "A" вызывает метод Secure01.methodA. Учтите, что НЕТ никаких других метаданных, относящихся к безопасности, каким будет ожидаемый результат?</p> <ol style="list-style-type: none"> 1) Исключительная ситуация на строке 25 2) Исключительная ситуация на строке 13 3) methodA не может быть вызван данным пользователем 4) Код будет выполнен без ошибок
7	<p>Выберите одно верное высказывание о JPA Entities:</p> <ol style="list-style-type: none"> 1) Первичный ключ может быть как примитивного типа, так и классом 2) Класс обязательно должен иметь public конструктор без параметров 3) JPA и Entity Manager не требуют наличия EJB-контейнера 4) Класс Entity должен быть сериализуемым (serializable)
8	<p>Для каких из перечисленных аннотаций может быть указан элемент optional (несколько вариантов)?</p> <ol style="list-style-type: none"> 1) @Basic 2) @Column 3) @OneToOne 4) @ManyToOne
9	<p>Какое единственное высказывание из приведенных верно?</p> <ol style="list-style-type: none"> 1) MDB может поддерживать сессионное состояние клиента 2) Клиент может вызвать метод, проаннотированный как @Remove, для уничтожения экземпляра MDB 3) Клиент не может получить прямую локальную ссылку на MDB 4) Клиент может получить прямую удаленную ссылку на MDB
10	<p>Java Persistence API определяет интерфейс Query. Какие два утверждения о методе Query.executeUpdate являются верными?</p> <ol style="list-style-type: none"> 1) Все управляемые сущности (managed entity objects), соответствующие обновленным записям в базе, синхронизируют свои данные в соответствии с обновлением 2) Произойдет PersistenceException, если не была обновлена ни одна сущность (entity) 3) Он должен всегда выполняться в границах транзакции 4) Произойдет IllegalStateException, если метод был вызван на объекте Query, созданном с указанием запроса SELECT
11	<p>Укажите правильные варианты регистрации некоего класса драйвера JDBC org.mydb.DriverClient (несколько вариантов):</p> <ol style="list-style-type: none"> 1) DriverManager.register("org.mydb.DriverClient") 2) Driver.register("org.mydb.DriverClient") 3) Class.forName("org.mydb.DriverClient"); 4) System.setProperty("jdbc.drivers", "org.mydb.DriverClient")

12	<p>Какие преимущества использования PreparedStatement по сравнению со Statement (несколько вариантов)?</p> <ol style="list-style-type: none"> 1) При использовании PreparedStatement запрос на стороне СУБД прекомпилируется, строится оптимальный план выполнения запроса, что может быть использовано многократно 2) Автоматическая защита от SQL-инъекций 3) Строка SQL задается однажды и далее многократно может быть использована 4) Легче контролировать соответствие типов параметров запроса типам SQL
13	<p>Для чего используется JDBC (несколько вариантов)?</p> <ol style="list-style-type: none"> 1) Для установки соединения с БД 2) Для отправки запросов к БД 3) Для обработки результатов 4) Для увеличения производительности ODBC драйверов
14	<p>Метод executeUpdate возвращает число измененных строк в БД, как выдумаете, что вернет этот метод если в SQL запросе использовались операции CREATE TABLE, DROP TABLE?</p> <ol style="list-style-type: none"> 1) Всегда будет возвращать 0 2) Я точно знаю, это -1. 3) executeUpdate не используется с DDL операторами 4) Для CREATE TABLE это 0, для DROP TABLE - число строк в таблице
15	<p>Выберите типы параметров, относящиеся к типам PreparedStatement (несколько вариантов):</p> <ol style="list-style-type: none"> 1) FIFO параметры 2) LIFO параметры 3) IN параметры 4) OUT параметры
16	<p>Объект какого класса инкапсулирует логическое соединение с БД?</p> <ol style="list-style-type: none"> 1) CallableStatement 2) Connection 3) Statement 4) DriverManager
17	<p>У нас есть пролистываемая таблица результатов в объекте ResultSet. Каким методом этого объекта можно передвинуть курсор на 6-ю строку?</p> <ol style="list-style-type: none"> 1) relative(6) 2) next(6) 3) absolute(6) 4) Не один из перечисленных
18	<p>Объекты каких классов позволяют загрузить и зарегистрировать необходимый JDBC-драйвер и получить соединение с базой данных или получить доступ к БД через пространство имен (несколько вариантов)?</p> <ol style="list-style-type: none"> 1) javax.sql.DataSource 2) java.sql.Statement 3) java.sql.Connection 4) java.sql.DriverManager
19	<p>Какая (какие) из следующих констант при создании объекта класса Statement определяет, что курсор можно двигать вперед-назад по объекту ResultSet (1-ый аргумент)?</p> <ol style="list-style-type: none"> 1) ResultSet.TYPE_SCROLL_INSENSITIVE 2) ResultSet.TYPE_FORWARD_ONLY 3) ResultSet.TYPE_SCROLL_SENSITIVE 4) Нет верного ответа
20	<p>Какой метод интерфейса Statement необходимо использовать при выполнении SQL-оператора SELECT, который возвращает объект ResultSet?</p> <ol style="list-style-type: none"> 1) execute() 2) executeQuery() 3) executeUpdate()

	4) executeSelect()
21	Родителем класса java.sql.Timestamp является: 1) java.util.Date 2) java.sql.Time 3) java.util.Time 4) java.util.Date
22	Какой интерфейс из пакета java.sql должен реализовывать каждый драйвер JDBC? 1) ResultSet 2) DriverPropertyInfo 3) Driver 4) DriverManager

Критерии оценки тестовых заданий, устных опросов:

1. Полнота знаний теоретического контролируемого материала.
2. Количество правильных ответов.

Тестовое задание / опрос считается сданным, если студент правильно ответил на 60 процентов от общего числа вопросов.

Критерии оценивания	Оценка
Студент ответил на 90 % (и более) вопросов	Отлично
Студент ответил на 70-89 % вопросов	Хорошо
Студент ответил на 60-69 % вопросов	Удовлетворительно
Студент ответил менее чем на 59 % вопросов	Неудовлетворительно

Сумма баллов по разделам дисциплины складывается из оценок, полученных обучающимся в течение семестра по всем формам текущего контроля. Каждая форма контроля оценивается баллом в интервале от 0 до 10.

Промежуточная аттестация осуществляется в форме зачета.

При заочной форме обучения в качестве оценочного средства аттестации раздела используется также контрольная работа.

Контрольная работа – средство проверки умений применять полученные знания для выполнения заданий по теме или разделу. Выполняется по индивидуальному заданию, представленному в методических указаниях для выполнения контрольных работ по дисциплине.

Критерии оценки контрольной работы

Максимальное количество баллов, начисляемое за контрольную работу, составляет 20 баллов по системе ECTS.

Расшифровка уровня знаний, соответствующего баллам ECTS

Оценка (ECTS)	Сумма баллов	Требования к знаниям на устном зачёте
«Зачтено» – А – Е	12 - 20	Оценка «Зачтено» выставляется студенту, если он выполнил не менее 60% заданий контрольной работы; верно ответил на вопросы преподавателя
«Не зачтено» – F	менее 12	Оценка «Не зачтено» выставляется студенту, если он выполнил менее 60% заданий контрольной работы; затрудняется с ответами на вопросы преподавателя

Перечень вопросов для подготовки к зачету

1. Соединение пользователей с ресурсами.
2. Прозрачность.
3. Открытость.
4. Масштабируемость.
5. Мультипроцессоры.
6. Гомогенные мультимедийные компьютерные системы.

7. Гетерогенные мультикомпьютерные системы.
8. Распределенные операционные системы.
9. Сетевые операционные системы.
10. Программное обеспечение промежуточного уровня.
11. Клиенты и серверы.
12. Разделение приложений по уровням.
13. Варианты архитектуры клиент-сервер.
14. Низкоуровневые протоколы.
15. Транспортные протоколы.
16. Протоколы верхнего уровня.
17. Базовые операции RPC.
18. Передача параметров.
19. Расширенные модели RPC.
20. DCE RPC.
21. Распределенные объекты.
22. Привязка клиента к объекту.
23. Статическое и динамическое удаленное обращение к методам.
24. Передача параметров.
25. Удаленные объекты DCE.
26. Java RMI.
27. Сохранность и синхронность во взаимодействиях.
28. Нерезидентная связь на основе сообщений.
29. Сохранная связь на основе сообщений.
30. IBM MQSeries.
31. Поддержка непрерывных сред.
32. Поток данных и качество обслуживания.
33. Синхронизация потоков данных.

Критерии оценки зачета

Шкалы оценки образовательных достижений

Сумма баллов	Оценка по 4-х балльной шкале	Зачет	Оценка (ECTS)	
90-100	отлично	Зачтено	A	
85-89	хорошо		B	
75-84			C	
70-74			D	
65-69	удовлетворительно	не зачтено	E	
60-64			F	
Ниже 60	неудовлетворительно			

Баллы (итоговой рейтинговой оценки)	Баллы (зачет)	Требования к знаниям
100-90	Зачтено 24 – 40 баллов	теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному
85 - 89		теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом в основном сформированы, все предусмотренные программой обучения учебные задания выполнены, качество выполнения большинства из них оценено числом баллов, близким к максимальному.

75 - 84		теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками.
65 - 74		теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки.
60-64		теоретическое содержание курса освоено частично, некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения учебные задания не выполнены, либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному
Ниже 60	не зачтено 0-23 баллов	очень слабые знания, недостаточные для понимания курса, имеется большое количество основных ошибок и недочетов

Учебно-методическое и информационное обеспечение учебной дисциплины

Основная литература:

1. Логинова, Ф. С. Программирование и программное обеспечение информационных технологий : учебное пособие / Ф. С. Логинова. — Санкт-Петербург : ИЭО СПбУТУиЭ, 2012. — 474 с. <https://e.lanbook.com/reader/book/64048/#2>

Дополнительная литература

2. Дэвис, К. Шаблоны проектирования для облачной среды : руководство / К. Дэвис ; перевод с английского Д. А. Беликова.. — Москва : ДМК Пресс, 2020. — 388 с. <https://e.lanbook.com/reader/book/140593/#5>

3. Свердлов, С. З. Языки программирования и методы трансляции: учебное пособие / С. З. Свердлов. — 2-е изд., испр. — Санкт-Петербург : Лань, 2019. — 564 с. <https://e.lanbook.com/book/173116>

Программное обеспечение и Интернет-ресурсы:

1. Электронно-библиотечная система «Лань».
2. Электронно-библиотечная система «Юрайт»
3. Российская национальная библиотека - <http://www.nlr.ru/>
4. Поисковая система - <http://www.rambler.ru>.
5. Поисковая система - <http://www.yandex.ru>.
6. Гарант - <http://base.garant.ru/>.
7. Интернет-Университет Информационных Технологий -<http://www.intuit.ru>

Для проведения практических занятий и выполнения самостоятельной работы используются учебные компьютерные классы с выходом в Интернет и лицензионным программным обеспечением.

Материально-техническое обеспечение учебной дисциплины

Лекционные занятия проводятся в аудиториях: учебная мебель, учебная доска, комплект мультимедийного оборудования

Практические занятия проводятся в компьютерных классах: учебная мебель, учебная доска, комплект мультимедийного оборудования, персональные компьютеры.

Для самостоятельной работы обучающихся имеется: читальный зал с выходом в сеть Интернет: Учебная мебель, комплект мультимедийного оборудования, персональные компьютеры, МФУ.

Учебно-методические рекомендации для студентов

1. Указания для прослушивания лекций

Перед началом занятий внимательно ознакомиться с учебным планом проведения лекций и списком рекомендованной литературы.

Перед посещением очередной лекции освежить в памяти основные концепции пройденного ранее материала. Подготовить при необходимости вопросы преподавателю. Не надо опасаться, что вопросы могут быть простыми.

На лекции основное внимание следует уделять не формулам и математическим выкладкам, а содержанию изучаемых вопросов, определениям и постановкам задач.

В процессе изучения лекционного курса необходимо по возможности часто возвращаться к основным понятиям и методам решения задач (здесь возможен выборочный контроль знаний студентов).

Желательно использовать конспекты лекций, в которых используется принятая преподавателем система обозначений.

Для более подробного изучения курса следует работать с рекомендованными литературными источниками и вновь появляющимися источниками.

2. Указания для участия в практических занятиях

Перед посещением уяснить тему практического занятия и самостоятельно изучить теоретические вопросы.

В конце практического занятия при необходимости выяснить у преподавателя неясные вопросы.

Основные результаты выполнения работы необходимо распечатать.

3. Указания для выполнения самостоятельной работы

Получить у преподавателя задание и список рекомендованной литературы. Изучение теоретических вопросов следует проводить по возможности самостоятельно, но при затруднениях обращаться к преподавателю.

Подготовить письменный отчет о проделанной работе.

При выполнении заданий по усмотрению преподавателя работа может быть оценена без письменного отчета на основе ответов на контрольные вопросы, при условии активной самостоятельной работы.

Подготовить ответы на контрольные вопросы.

Методические рекомендации для преподавателей

1. Указания для проведения лекций

На первой вводной лекции сделать общий обзор содержания курса и отметить новые методы и подходы к решению задач, рассматриваемых в курсе, довести до студентов требования кафедры, ответить на вопросы.

При подготовке к лекционным занятиям необходимо продумать план его проведения, содержание вступительной, основной и заключительной части лекции, ознакомиться с новинками учебной и методической литературы, публикациями периодической печати по теме лекционного занятия. Уточнить план проведения семинарского занятия по теме лекции. Перед изложением текущего лекционного материала напомнить об основных итогах, достигнутых на предыдущих лекциях. С этой целью задать несколько вопросов аудитории и осуществить выборочный контроль знания студентов.

В ходе лекционного занятия преподаватель должен назвать тему, учебные вопросы, ознакомить студентов с перечнем основной и дополнительной литературы по теме занятия. Раскрывая содержание учебных вопросов, акцентировать внимание студентов на основных категориях, явлениях и процессах, особенностях их протекания. Раскрывать сущность и содержание различных точек зрения и научных подходов к объяснению тех или иных явлений и процессов.

Следует аргументировано обосновать собственную позицию по спорным теоретическим вопросам. Приводить примеры. Задавать по ходу изложения лекционного материала риторические вопросы и самому давать на них ответ. Это способствует активизации мыслительной деятельности студентов, повышению их внимания и интереса к материалу лекции, ее содержанию. Преподаватель должен руководить работой студентов по конспектированию лекционного материала, подчеркивать необходимость отражения в конспектах основных положений изучаемой темы, особо выделяя, категориальный аппарат. В заключительной части лекции необходимо сформулировать

общие выводы по теме, раскрывающие содержание всех вопросов, поставленных в лекции. Объявить план очередного практического занятия, дать краткие рекомендации по подготовке студентов к семинару. Определить место и время консультации студентам, пожелавшим выступить на занятии с докладами.

На последней лекции уделить время для обзора наиболее важных положений, рассмотренных в курсе.

2. Указания для проведения практических занятий

Четко обозначить тему занятия.

Обсудить основные понятия, связанные с темой.

В процессе решения задач вести дискуссию со студентами о правильности применения теоретических знаний.

Отмечать студентов, наиболее активно участвующих в решении задач и дискуссиях.

В конце занятия задать аудитории несколько контрольных вопросов.

3. Указания по контролю самостоятельной работы студентов

По усмотрению преподавателя задание на самостоятельную работу может быть индивидуальным или фронтальным.

При использовании индивидуальных заданий требовать от студента письменный отчет о проделанной работе.

При применении фронтальных заданий вести коллективные обсуждения со студентами основных теоретических положений.

С целью контроля качества выполнения самостоятельной работы требовать индивидуальные отчеты (допустимо вместо письменного отчета применять индивидуальные контрольные вопросы).

Программа составлена в соответствии с требованиями ОС НИЯУ МИФИ по направлению 09.03.02 Информационные системы и технологии

Рабочую программу составил ст. преподаватель Р.В. Ефремов

Рецензент: к.т.н., доцент И.А. Штырова

Программа одобрена на заседании УМКН «Информационные системы и технологии».

Председатель учебно-методической комиссии О.В. Виштак