Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 23.10.2023 16:02:46
Уникальный программный ключ:
8db180d1a3f02ac9e60521a5672742735c18b1d6

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение
высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ

Декан факультета
информационных технологий
/Д. Г. Демидов/

«28» 04 2022 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Проблемно-ориентированные языки»

Направление подготовки
09.03.03 Прикладная информатика
Образовательная программа (профиль подготовки)
«Большие и открытые данные»

Квалификация (степень) выпускника
бакалавр

Форма обучения
очная

Москва 2022

Программа дисциплины «Проблемно-ориентированные языки» составлена в соответствии с требованиями ФГОС ВО и учебным планом по направлению **09.03.03 «Прикладная информатика»** и профилю подготовки **«Большие и открытые данные».**

Программу составил 

_____/А.В. Осипов/

Программа дисциплины утверждена на заседании кафедры «Прикладная информатика»
«____» августа 2022 г. протокол № _____

Заведующий кафедрой
доцент, к.э.н.  /С.В.Суворов/

Программа согласована с руководителем образовательной программы по направлению подготовки **09.03.03 «Прикладная информатика»** по профилю подготовки **«Большие и открытые данные».**


_____ /С.В.Суворов/
«____» августа 2022 г.

Программа утверждена на заседании учебно-методической комиссии факультета Информационных технологий

Председатель комиссии  / Д. Г. Демидов/

«____»_____2022 г. Протокол:

1. Цели освоения дисциплины

К **основным целям** освоения дисциплины «Проблемно-ориентированные языки» следует отнести:

– формирование знаний о современных принципах построения, методах и средствах работы с проблемно-ориентированными языками на примере программирования прикладных задач на языке Java;

– подготовка студентов к деятельности в соответствии с квалификационной характеристикой бакалавра по направлению, в том числе формирование умений по выявлению необходимых усовершенствований и разработке нового программного обеспечения.

К **основным задачам** освоения дисциплины «Проблемно-ориентированные языки» следует отнести:

– освоение методологии, анализа и выбора принципов и методов проблемно-ориентированного программирования на примере программирования прикладных задач на языке программирования высокого уровня.

2. Место дисциплины в структуре ООП бакалавриата

Дисциплина «Проблемно-ориентированные языки» относится к числу профессиональных учебных дисциплин базовой части базового цикла (Б1) основной образовательной программы бакалавриата. Дисциплина «Проблемно-ориентированные языки» взаимосвязана логически и содержательно-методически со следующими дисциплинами и практиками ООП:

В базовой части базового цикла (Б1):

- теоретические основы информатики;
- программирование;
- Веб-технологии.

3. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы

В результате освоения дисциплины (модуля) у обучающихся формируются следующие компетенции и должны быть достигнуты следующие результаты обучения как этап формирования соответствующих компетенций:

Код компетенции	В результате освоения образовательной программы обучающийся должен обладать	Перечень планируемых результатов обучения по дисциплине
ОПК-3	способностью к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	**знать:** • общие принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям. **уметь:** • разрабатывать отдельные компоненты и дополнения алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям. **владеть:** • методами и средствами создания алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.
ПК-7	способностью к разработке и применению	**знать:** • теоретические и практические подходы к разработке и применению

	алгоритмических и программных решений в области системного и прикладного программного обеспечения.	алгоритмических и программных решений в области системного и прикладного программного обеспечения. **уметь:** • разрабатывать и применять алгоритмические и программные решения в области системного и прикладного программного обеспечения. **владеть:** • методами разработки и применения алгоритмических и программных решений в области системного и прикладного программного обеспечения.

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 4 зачетные единицы, т.е. 144 академических часов (из них 72 часов – самостоятельная работа студентов).

Разделы дисциплины «Проблемно-ориентированные языки» изучаются на первом курсе. На первом курсе в **первом** семестре выделяется 4 зачетные единицы, т.е. 144 академических часов (из них 72 часов – самостоятельная работа студентов).

Первый семестр: лекции – 1 час в неделю (18 часов), лабораторные работы – 2 часа в неделю (54 часов), форма контроля – экзамен.

Содержание разделов дисциплины «Проблемно-ориентированные языки» отражено ниже. Структура и содержание разделов дисциплины «Проблемно-ориентированные языки» по срокам и видам работы отражены в приложении 1.

Содержание разделов дисциплины

Первый семестр

Введение

Предмет, задачи и содержание дисциплины. Структура курса, его место и роль в подготовке специалиста, связь с другими дисциплинами. Примеры проблемно-ориентированных языков. Их назначение и особенности. Проблемно-ориентированные надстройки современных языков программирования. Язык Java как пример проблемно-ориентированного языка. Создание оконного интерфейса приложений Java с помощью GUI Builder IDE NetBeans. Обработка событий.

Приложение автоматизации сети ресторанов как пример разработки программы работы с файлами и каталогами с оконным интерфейсом в IDE NetBeans.

Многопоточное программирование

Многопоточное программирование в языке Java. Методы класса Thread. Главный поток. Создание дочернего потока. Запуск и остановка потока. Ожидание завершения потока. Синхронизация потоков в языке Java. Определение всех выполняющихся потоков.

Сериализация и десериализация

Сериализация и десериализация. Основные понятия и принципы. Примеры на языке Java. Приложение автоматизации сети ресторанов как пример применения механизма сериализации и десериализации классов Java при создании сетевых приложений.

Модель клиент-сервер и ее варианты

Модель клиент-сервер и ее варианты. Протоколы сетевого взаимодействия. IP-адрес и порт. Назначение, основные способы применения в Java-приложениях. Сокеты. Назначение, основные способы применения в Java-приложениях. Основные классы Java для сетевого программирования. Назначение, основные способы применения. Сетевые клиенты. Подключение к серверу. Нахождение IP- адреса и доменных имен. Обработка сетевых ошибок. Чтение Web-страницы по протоколу HTTP. Сетевые серверы. Описание сервера на языке Java и получение им запроса от клиента. Обработка сервером средствами языка Java нескольких клиентских запросов. Возвращение сервером примитивного типа данных. Возвращение сервером объекта. Обслуживание сервером HTTP запроса средствами языка Java.

Создание сетевых приложений на языке программирования Java с использованием протоколов UDP и TCP

Этапы реализации обмена данными по протоколу UDP средствами языка Java. Особенности создания сетевых приложений с использованием протокола UDP. Класс Java DatagramPacket. Назначение, основные способы применения. Класс Java DatagramSocket. Назначение, основные способы применения. Создание сервера UDP средствами языка Java. Назначение, основные способы его создания и применения. Создание клиента UDP на Java. Назначение, основные способы его создания и применения. Программа сетевого взаимодействия по протоколу UDP как пример разработки сетевого приложения на основе протокола UDP в IDE

NetBeans. Этапы установления TCP соединения между двумя Java-приложениями. Класс Java Socket. Назначение, основные способы применения в распределенных системах. Класс Java ServerSocket. Назначение, основные способы применения в распределенных системах. Создание сервера TCP средствами языка Java. Назначение, основные способы его создания и применения. Создание клиента TCP средствами языка Java. Назначение, основные способы его создания и применения. Программа чата как пример разработки сетевого приложения на основе протокола TCP в IDE NetBeans.

Архитектура RMI

Краткая характеристика архитектуры RMI. Ее уровни, пакеты. Регистр RMI. Уровень удаленной ссылки в архитектуре RMI. Транспортный уровень архитектуры RMI. Этапы создания и выполнения приложения RMI на языке Java. Пакеты RMI. Их назначение, способ применения и краткая характеристика. Описание удаленного интерфейса в RMI средствами Java. Обращение к удаленным объектам RMI. Реализация удаленного интерфейса в RMI средствами Java. Создание сервера RMI средствами Java. Создание клиента RMI средствами Java. Создание удаленного сервисного класса для сервера RMI на языке Java. Выполняемые им операции. Политика безопасности RMI. Компоненты приложения RMI и их применение в сетевых приложениях. Пример разработки RMI приложения в IDE NetBeans.

XML документы

XML документы и их применение в Java приложениях. Синтаксический анализ XML документов средствами языка Java. Проверка достоверности XML документов средствами языка Java. Синтаксический анализ XML документов древовидными (DOM) анализаторами. Синтаксический анализ XML документов потоковыми (SAX) анализаторами. Сравнительная характеристика DOM и SAX анализаторов. Формирование XML документов. Запись XML документов. Преобразование XML документов средствами XLST. Приложение чтения курса валют с сайта ЦБ России как пример разработки программы чтения и анализа XML документа в IDE NetBeans.

Работа с базами данных с помощью языка программирования Java

Технология JDBC. Подключение к базам данных через JDBC в языке Java. Отправка запросов через JDBC. Извлечение результатов запроса к базе данных в языке Java. Приложение автоматизации сети ресторанов как пример разработки программы доступа к базам данных при помощи технологии JDBC в IDE NetBeans.

5. Образовательные технологии

Методика преподавания дисциплины «Проблемно-ориентированные языки» и реализация компетентностного подхода в изложении и восприятии материала предусматривает использование следующих активных и интерактивных форм проведения групповых, индивидуальных, аудиторных занятий в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков обучающихся:

– подготовка к выполнению лабораторных работ в лабораториях вуза;

– обсуждение и защита рефератов по дисциплине;

– подготовка, представление и обсуждение презентаций на семинарских занятиях;

– организация и проведение текущего контроля знаний студентов в форме бланкового тестирования;

– проведение интерактивных занятий по процедуре подготовки к интернет-тестированию*;*

– использование интерактивных форм текущего контроля в форме аудиторного и внеаудиторного интернет-тестирования;

– проведение мастер-классов экспертов и специалистов по программированию.

Удельный вес занятий, проводимых в интерактивных формах, определен главной целью образовательной программы, особенностью контингента обучающихся и содержанием дисциплины «Проблемно-ориентированные языки» и в целом по дисциплине составляет 50% аудиторных занятий. Занятия лекционного типа составляют 25% от объема аудиторных занятий.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

В процессе обучения используются следующие оценочные формы самостоятельной работы студентов, оценочные средства текущего контроля успеваемости и промежуточных аттестаций:

<u>В первом семестре</u>

- реферат по теме занятий (индивидуально для каждого обучающегося);
- подготовка к выполнению лабораторных работ и их защита.

Оценочные средства текущего контроля успеваемости включают контрольные вопросы и задания в форме бланкового и (или) компьютерного

тестирования, для контроля освоения обучающимися разделов дисциплины, защита рефератов.

Образцы тестовых заданий, контрольных вопросов и заданий для проведения текущего контроля приведены в приложении 2.

6.1. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю)

6.1.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы.

В результате освоения дисциплины (модуля) формируются следующие компетенции:

Код компетенции	В результате освоения образовательной программы обучающийся должен обладать
ОПК-3	способностью к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.
ПК-7	способностью к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения.

В процессе освоения образовательной программы данные компетенции, в том числе их отдельные компоненты, формируются поэтапно в ходе освоения обучающимися дисциплин (модулей), практик в соответствии с учебным планом и календарным графиком учебного процесса.

6.1.2. Описание показателей и критериев оценивания компетенций, формируемых по итогам освоения дисциплины (модуля), описание шкал оценивания

Показателем оценивания компетенций на различных этапах их формирования является достижение обучающимися планируемых результатов обучения по дисциплине (модулю).

ОПК-3 - способностью к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям				
Показатель	**Критерии оценивания**			
	2	**3**	**4**	**5**
знать: - общие принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	Обучающийся демонстрирует полное отсутствие или недостаточное соответствие следующих знаний: общие принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	Обучающийся демонстрирует неполное соответствие следующих знаний: общие принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям. Допускаются значительные ошибки, проявляется недостаточность знаний по ряду показателей, обучающийся испытывает значительные затруднения при оперировании	Обучающийся демонстрирует частичное соответствие следующих знаний: общие принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	Обучающийся демонстрирует полное соответствие следующих знаний: общие принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.

		знаниями при их переносе на новые ситуации.		
уметь: - разрабатывать отдельные компоненты и дополнения алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	Обучающийся не умеет или в недостаточной степени умеет разрабатывать отдельные компоненты и дополнения алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	Обучающийся демонстрирует неполное соответствие следующих умений: разрабатывать отдельные компоненты и дополнения алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям. Допускаются значительные ошибки, проявляется недостаточность умений, по ряду показателей, обучающийся испытывает значительные	Обучающийся демонстрирует частичное соответствие следующих умений: разрабатывать отдельные компоненты и дополнения алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям. Умения освоены, но допускаются незначительные ошибки, неточности, затруднения при операциях по работе с программным обеспечением, переносе умений на	Обучающийся демонстрирует полное соответствие следующих умений: разрабатывать отдельные компоненты и дополнения алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям. Свободно оперирует приобретенными умениями, применяет их в ситуациях повышенной сложности.

		затруднения при оперировании умениями при их переносе на новые ситуации.	новые, нестандартные ситуации.	
владеть: методами и средствами создания алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	Обучающийся не владеет или в недостаточной степени владеет методами и средствами создания алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	Обучающийся владеет методами и средствами создания алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям в неполном объеме, допускаются значительные ошибки, проявляется недостаточность владения навыками по ряду показателей, Обучающийся испытывает значительные затруднения при применении навыков в новых ситуациях.	Обучающийся частично владеет методами и средствами создания алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям, навыки освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе умений на новые, нестандартные ситуации.	Обучающийся в полном объеме владеет методами и средствами создания алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям, свободно применяет полученные навыки в ситуациях повышенной сложности.

ПК-7 – способностью к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения				
знать: теоретические и практические подходы к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения.	Обучающийся демонстрирует полное отсутствие или недостаточное соответствие следующих знаний: теоретические и практические подходы к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения.	Обучающийся демонстрирует неполное соответствие следующих знаний: теоретические и практические подходы к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения.	Обучающийся демонстрирует частичное соответствие следующих знаний: теоретические и практические подходы к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях.	Обучающийся демонстрирует полное соответствие следующих знаний: теоретические и практические подходы к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения, свободно оперирует приобретенными знаниями.
уметь: разрабатывать и применять алгоритмические и программные решения в области системного и прикладного программного обеспечения.	Обучающийся не умеет или в недостаточной степени умеет разрабатывать и применять алгоритмические и программные решения в области системного и прикладного программного обеспечения.	Обучающийся демонстрирует неполное соответствие следующих умений: разрабатывать и применять алгоритмические и программные решения в области системного и прикладного программного обеспечения. Допускаются значительные ошибки, проявляется недостаточность умений, по ряду показателей, обучающийся испытывает значительные затруднения при оперировании	Обучающийся демонстрирует частичное соответствие следующих умений: разрабатывать и применять алгоритмические и программные решения в области системного и прикладного программного обеспечения. Умения освоены, но допускаются незначительные ошибки, неточности, затруднения при их демонстрации на практике, переносе умений на новые, нестандартные ситуации.	Обучающийся демонстрирует полное соответствие следующих умений: разрабатывать и применять алгоритмические и программные решения в области системного и прикладного программного обеспечения. Свободно оперирует приобретенными умениями, применяет их в ситуациях повышенной сложности.

		умениями при их переносе на новые ситуации.		
владеть: - методами разработки и применения алгоритмических и программных решений в области системного и прикладного программного обеспечения.	Обучающийся не владеет или в недостаточной степени владеет методами разработки и применения алгоритмических и программных решений в области системного и прикладного программного обеспечения.	Обучающийся владеет методами разработки и применения алгоритмических и программных решений в области системного и прикладного программного обеспечения, допускаются значительные ошибки, проявляется недостаточность владения навыками по ряду показателей, Обучающийся испытывает значительные затруднения при применении навыков в новых ситуациях.	Обучающийся частично владеет методами разработки и применения алгоритмических и программных решений в области системного и прикладного программного обеспечения, навыки освоены, но допускаются незначительные ошибки, неточности, затруднения при их демонстрации на практике, переносе умений на новые, нестандартные ситуации.	Обучающийся в полном объеме владеет методами разработки и применения алгоритмических и программных решений в области системного и прикладного программного обеспечения, свободно применяет полученные навыки в ситуациях повышенной сложности.

Шкалы оценивания результатов промежуточной аттестации и их описание:

Форма промежуточной аттестации: экзамен

Промежуточная аттестация обучающихся в форме экзамена проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по данной дисциплине (модулю), при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине (модулю) проводится преподавателем, ведущим занятия по дисциплине (модулю) методом экспертной оценки. По итогам промежуточной аттестации по дисциплине (модулю) выставляется оценка «отлично», «хорошо», «удовлетворительно» или «неудовлетворительно».

К промежуточной аттестации допускаются только студенты, выполнившие все виды учебной работы, предусмотренные рабочей программой по дисциплине «Проблемно-ориентированные языки»: выполнили лабораторные работы, выступили с докладом по теме реферата.

Шкала оценивания	**Описание**
Отлично	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Хорошо	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует неполное, правильное соответствие знаний, умений, навыков приведенным в таблицах показателей, либо если при этом были допущены 2-3 несущественные ошибки.
Удовлетворительно	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, в котором освещена основная, наиболее важная часть материала, но при этом допущена одна значительная ошибка или неточность.
Неудовлетворительно	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

Фонды оценочных средств представлены в приложении 2 к рабочей программе.

7. Учебно-методическое и информационное обеспечение дисциплины.

а) основная литература:

1) Васильев, А.Н. Самоучитель Java с примерами и программами [Электронный ресурс] : учеб. пособие / А.Н. Васильев. — Электрон. дан. — Санкт-Петербург : Наука и Техника, 2017. — 368 с. — Режим доступа: https://e.lanbook.com/book/101548. — Загл. с экрана.
2) Вязовик, Н.А. Программирование на Java [Электронный ресурс] : учеб. пособие — Электрон. дан. — Москва : , 2016. — 603 с. — Режим доступа: https://e.lanbook.com/book/100405. — Загл. с экрана.

б) дополнительная литература:

1) Риз, Р. Обработка естественного языка на Java [Электронный ресурс] : учеб. пособие — Электрон. дан. — Москва : ДМК Пресс, 2016. — 264 с. — Режим доступа: https://e.lanbook.com/book/93272. — Загл. с экрана.

в) программное обеспечение и интернет-ресурсы:

1. Язык программирования Java версии 8.1.
2. Интегрированная среда разработки IDE NetBeans 8.1.

8. Материально-техническое обеспечение дисциплины

Компьютерные классы кафедры «Прикладная информатика», оснащенные компьютерами с установленным программным обеспечением в соответствии с п. 7 рабочей программы

9. Методические рекомендации для самостоятельной работы студентов

Обучение по дисциплине «Проблемно-ориентированные языки» предполагает изучение курса на аудиторных занятиях и самостоятельной работы обучающихся. Виды самостоятельной работы обучающегося, порядок и сроки ее выполнения:

- подготовка к лекциям, лабораторным работам и практическим занятиям с использованием конспекта лекций, материалов лабораторных занятий и учебно-методического и информационного обеспечения дисциплины (в течение 2-го семестра в соответствии с расписанием занятий);
- оформление отчетов по выполненным лабораторным работам и теоретическая подготовка к их сдаче (в течение 2-го семестра в соответствии с расписанием занятий).

Перечень вопросов для проведения текущего контроля и промежуточной аттестации – в соответствии с тематикой разделов дисциплины.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, которая является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;

– разъясняет учебные элементы, трудные для понимания;

– систематизирует учебный материал;

– ориентирует в учебном процессе.

Подготовка к занятиям лекционного типа заключается в следующем:

– внимательном чтении материала предыдущей лекции;

– предварительное ознакомление с темой предстоящей лекции (по тематическому плану, по информации лектора);

– ознакомление с учебным материалом по учебнику и учебным пособиям;

– уяснение места изучаемой темы в своей профессиональной подготовке;

– подготовке возможных вопросов, которые студент задаст лектору на лекции.

Подготовка к написанию реферата исходит из того, что реферат – это самостоятельная учебно-исследовательская работа обучающегося, где автор раскрывает суть исследуемой проблемы, приводит различные точки зрения, а также собственные взгляды на нее. Содержание материала должно быть логичным, изложение материала носит проблемно-поисковый характер.

Этапы работы над рефератом.

1. Формулирование темы. Тема должна быть не только актуальной по своему значению, но и оригинальной, интересной по содержанию.

2. Подбор и изучение основных источников по теме.

3. Составление библиографии.

4. Обработка и систематизация информации.

5. Разработка плана реферата.

6. Оформление реферата в виде презентации.

7. Публичное выступление с результатами исследования на занятии.

Содержание работы должно отражать.

1. Знание современного состояния проблемы.

2. Обоснование выбранной темы.

3. Использование известных результатов и фактов.

4. Полноту цитируемой литературы, ссылки на работы ученых, занимающихся данной проблемой.

5. Актуальность поставленной проблемы.

6. Материал, подтверждающий научное, либо практическое значение в настоящее время.

Типовая структура реферата.

1. Титульный лист.

2. План (простой или развернутый).

3. Введение.

4. Основная часть.

5. Заключение.

6. Список литературы.

Реферат может рассматриваться как одна из форм контрольных работ. Объем реферата не должен превышать 10 страниц.

Представление реферата осуществляется в форме доклада с предъявлением презентации.

Приложение 1 к
рабочей программе

Структура и содержание дисциплины «Проблемно-ориентированные языки» по направлению подготовки 09.03.03 «Прикладная информатика» (бакалавр)

n/n	Раздел	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов, и трудоемкость в часах					Виды самостоятельной работы студентов					Формы аттестации	
				Л	П/С	Лаб	СРС	КСР	К.Р.	К.П.	РГР	Реферат	К/р	Э	З
	Второй семестр														
1.1	**Введение. Создание оконного интерфейса приложений Java с помощью GUI Builder IDE NetBeans.**	2	1	2											
1.2	*Лабораторная работа «Приложение автоматизации сети ресторанов как пример разработки программы работы с файлами и каталогами с оконным интерфейсом в IDE NetBeans».*	2	1,2			6	12								
1.3	**Многопоточное программирование.**	2	2	2											
1.4	*Лабораторная работа «*Приложение автоматизации сети ресторанов как пример многопоточного приложения».	2	3,4			6	12								
1.5	**Сериализация и**	2	3	2											

	десериализация.														
1.6	*Лабораторная работа* «Приложение автоматизации сети ресторанов как пример применения механизма сериализации и десериализации классов Java при создании сетевых приложений».	2	5,6			7	12								
1.7	**Модель клиент-сервер и ее варианты. Выдача задания на реферат.**	2	4	2			3					+			
1.8	*Лабораторная работа* «Программа сетевого взаимодействия по протоколу UDP как пример разработки сетевого приложения на основе протокола UDP в IDE NetBeans».	2	7,8			7	6					+			
1.9	**Программа чата как пример разработки сетевого приложения на основе протокола TCP в IDE NetBeans.**	2	5	2								+			
1.10	*Лабораторная работа* «Программа чата как пример разработки сетевого приложения на основе протокола TCP в IDE NetBeans».	2	9,10			7	9					+			
1.11	**Архитектура RMI.**.	2	6	2								+			
1.12	Лабораторная работа «Пример разработки RMI приложения в IDE NetBeans».	2	11,12			7	6					+			

1.13	**XML документы.**	2	7	2								+			
1.14	*Лабораторная работа* «Приложение чтения курса валют с сайта ЦБ России как пример разработки программы чтения и анализа XML документа в IDE NetBeans».	2	13,14			7	6					+			
1.15	**Работа с базами данных с помощью языка программирования Java.**	2	8	4								+			
1.16	*Лабораторная работа* «Приложение автоматизации сети ресторанов как пример разработки программы доступа к базам данных при помощи технологии JDBC в IDE NetBeans».	2	15,16			7	6					+			
	Форма аттестации		19-21												З
	Всего часов по дисциплине в первом семестре			18		54	72					+			

Приложение 2 к
рабочей программе

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение
высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Направление подготовки: 09.03.03 «Прикладная информатика»

ОП (профиль): «Большие и открытые данные»

Вид профессиональной деятельности: (В соответствии с ФГОС ВО)

Форма обучения
Очная

Кафедра: Прикладной информатики

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ПО ДИСЦИПЛИНЕ

«ПРОБЛЕМНО-ОРИЕНТИРОВАННЫЕ ЯЗЫКИ»

Состав: 1. Паспорт фонда оценочных средств

2. Описание оценочных средств:

2.1 Показатели уровня сформированности компетенций.
2.2 Пример тем рефератов.
2.3 Пример комплекта разноуровневых задач и заданий.
2.4 Пример вопросов устного опроса.

Москва, 2022 год

Паспорт фонда оценочных средств

Фонд оценочных средств включает в себя:

1. Показатели уровня сформированности компетенций.
2. Пример тем рефератов.
3. Пример комплекта разноуровневых задач и заданий.
4. Пример вопросов устного опроса.

ПОКАЗАТЕЛЬ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Проблемно-ориентированные языки					
ФГОС ВО 09.03.03 «Прикладная информатика»					
В процессе освоения данной дисциплины студент формирует и демонстрирует следующие **профессиональные компетенции**:					
КОМПЕТЕНЦИИ		**Перечень компонентов**	**Технология формирования компетенций**	**Форма оценочного средства**	**Степени уровней освоения компетенций**
ИН-ДЕКС	**ФОРМУЛИРОВ-КА**				
ОПК-3	***способностью*** *к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз*	**Знать:** - общие принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	лекции, самостоятельная работа, лабораторные работы.	УО, РЗЗ, Р	**Базовый уровень** - студент способен разрабатывать и применять алгоритмические и программные решения в области системного и прикладного программирования, математических, информационных и имитационных моделей, создавать информационные ресурсы глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям в стандартных учебных ситуациях. **Повышенный уровень** - студент способен разрабатывать и применять алгоритмические и программные решения в области системного и

	данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	**Уметь:** - разрабатывать отдельные компоненты и дополнения алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям. **Владеть*:*** - методами и средствами создания алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей,			прикладного программирования, математических, информационных и имитационных моделей, создавать информационные ресурсы глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям в практической деятельности.

		информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.			
ПК-7	***способностью*** *к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения.*	**Знать:** - теоретические и практические подходы к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения. **Уметь:** - разрабатывать и применять алгоритмические и программные решения в области системного и прикладного программного обеспечения. **Владеть*:*** - методами разработки и применения алгоритмических и программных решений в области системного и прикладного программного обеспечения.	лекции, самостоятельная работа, лабораторные работы.	УО, РЗЗ, Р	**Базовый уровень** - студент способен разрабатывать и применять алгоритмические и программные решения в области системного и прикладного программного обеспечения в стандартных учебных ситуациях. **Повышенный уровень** - студент способен разрабатывать и применять алгоритмические и программные решения в области системного и прикладного программного обеспечения в практической деятельности.

Пример тем рефератов

1. Создание оконного интерфейса приложений Java с помощью GUI Builder IDE NetBeans.
2. Многопоточное программирование в языке Java.
3. Модель клиент-сервер и ее варианты.
4. Протоколы сетевого взаимодействия.
5. Краткая характеристика архитектуры RMI.
6. XML документы и их применение в Java приложениях.
7. Технология JDBC. Подключение к базам данных через JDBC в языке Java.
8. Основные классы Java для сетевого программирования.
9. Сериализация и десериализация.
10. Анализ XML документов средствами языка Java

Пример комплекта разноуровневых задач и заданий

Требуется написать и отладить визуальное приложение автоматизации платежей клиентами сети ресторанов.

a) Клиенты сети ресторанов могут завести именную карту клиента с предоставлением им в любом из ресторанов сети следующих услуг: получение справки о владельце и балансе карты, пополнение карты, оплаты услуг с помощью карты (списывание денег), расчет и применение скидки при оплате услуг, обслуживание в долг и иные услуги по усмотрению разработчика.

При написании визуального приложения предусмотреть:

b) обработку запросов обслуживания карт по кодам операций с переменным объемом уточняющей информации;

c) передачей по сети как примитивных типов данных, так и транспортных классов с использованием их сериализации и десериализации;

d) хранение данных на сервере в файлах;

e) хранение данных на сервере реляционной базе данных (рекомендуется использовать базу данных PostgreSQL.

Пример вопросов устного опроса

1. Главный поток. Создание дочернего потока. Запуск и остановка потока. Ожидание завершения потока. Примеры на языке Java.
2. Синхронизация потоков в языке Java. Определение всех выполняющихся потоков. Примеры.
3. IP адрес и порт. Назначение, основные способы применения в Java-приложениях. Примеры.
4. Сокеты. Назначение, основные способы применения в Java-приложениях. Примеры.
5. Сетевые клиенты. Подключение к серверу. Примеры на языке Java.
6. Нахождение IP-адреса и доменных имен. Обработка сетевых ошибок. Примеры на языке Java.
7. Чтение Web-страницы по протоколу HTTP. Примеры на языке Java.
8. Сетевые серверы. Описание сервера на языке Java и получение им запроса от клиента. Примеры.
9. Обработка сервером средствами языка Java нескольких клиентских запросов. Возвращение сервером примитивного типа данных. Возвращение сервером объекта. Примеры.
10. Обслуживание сервером HTTP запроса средствами языка Java. Примеры.
11. Этапы реализации обмена данными по протоколу UDP средствами языка Java. Примеры.
12. Особенности создания сетевых приложений с использованием протокола UDP. Примеры.
13. Класс Java DatagramPacket. Назначение, основные способы применения. Примеры.
14. Класс Java DatagramSocket. Назначение, основные способы применения. Примеры.
15. Создание сервера UDP средствами языка Java. Назначение, основные способы его создания и применения. Примеры.
16. Создание клиента UDP на Java. Назначение, основные способы его создания и применения. Примеры.
17. Этапы установления TCP соединения между двумя Java-приложениями. Примеры.
18. Класс Java Socket. Назначение, основные способы применения в распределенных системах. Примеры.

19. Класс Java ServerSocket. Назначение, основные способы применения в распределенных системах. Примеры.
20. Создание сервера TCP средствами языка Java. Назначение, основные способы его создания и применения. Примеры.
21. Создание клиента TCP средствами языка Java. Назначение, основные способы его создания и применения. Примеры.
22. Уровень удаленной ссылки в архитектуре RMI. Примеры.
23. Транспортный уровень архитектуры RMI. Примеры.
24. Этапы создания и выполнения приложения RMI на языке Java. Примеры.
25. Пакеты RMI. Их назначение, способ применения и краткая характеристика. Примеры.
26. Описание удаленного интерфейса в RMI средствами Java. Примеры.
27. Обращение к удаленным объектам (RMI). Примеры.
28. Реализация удаленного интерфейса в RMI средствами Java. Примеры.
29. Создание сервера RMI средствами Java. Примеры.
30. Создание клиента RMI средствами Java. Примеры.
31. Создание удаленного сервисного класса для сервера RMI на языке Java. Выполняемые им операции. Примеры.
32. Политика безопасности RMI. Примеры.
33. Компоненты приложения RMI и их применение в сетевых приложениях. Примеры.
34. Синтаксический анализ XML документов средствами языка Java. Примеры.
35. Проверка достоверности XML документов средствами языка Java. Примеры.
36. Синтаксический анализ XML документов древовидными (DOM) анализаторами. Примеры.
37. Синтаксический анализ XML документов потоковыми (SAX) анализаторами. Примеры.
38. Сравнительная характеристика DOM и SAX анализаторов. Примеры.
39. Формирование XML документов. Примеры.
40. Запись XML документов. Примеры.
41. Преобразование XML документов средствами XLST. Примеры.
42. Извлечение результатов запроса к базе данных в языке Java. Примеры.
43. Приложение автоматизации сети ресторанов как пример разработки программы доступа к базам данных при помощи технологии JDBC в IDE NetBeans.