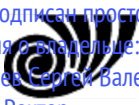


Документ подписан простой электронной подписью Информация о владельце: ФИО: Таскаев Сергей Валерьевич Должность: Ректор	 МИНОБРАЗОВАНИЯ И НАУКИ РОССИИ Федеральное государственное бюджетное образовательное учреждение высшего образования «Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)	
Дата подписания: 08.06.2025 15:37:05 Уникальный программный ключ: 04c19ed8b0961506cb77a486b9a878806522525	Рабочая программа дисциплины "Программирование на языке Java (научный семинар)" по направлению подготовки (специальности) 02.03.02 "Фундаментальная информатика и информационные технологии" направленности (профилю) Прикладное программирование и системы искусственного интеллекта ФГБОУ ВО «ЧелГУ»	стр. 1

Рабочая программа дисциплины (модуля)*
Программирование на языке Java (научный семинар)

Направление подготовки (специальность)

02.03.02 Фундаментальная информатика и информационные технологии

Направленность (профиль)

Прикладное программирование и системы искусственного интеллекта

Присваиваемая квалификация (степень)

бакалавр

Форма обучения

очная

Год(ы) набора 2026

*Рабочая программа дисциплины (модуля) адаптирована для инклюзивного обучения инвалидов и лиц с ограниченными возможностями здоровья

Челябинск 2026 г.



Содержание

1. Цели освоения дисциплины
2. Место дисциплины в структуре ОПОП
3. Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля)
4. Объем дисциплины (модуля)
5. Структура и содержание дисциплины (модуля)
6. Фонд оценочных средств
 - 6.1. Перечень видов оценочных средств
 - 6.2. Типовые контрольные задания и иные материалы для текущей аттестации
 - 6.3. Типовые контрольные вопросы и задания для промежуточной аттестации
 - 6.4. Критерии оценивания
7. Учебно-методическое и информационное обеспечение дисциплины (модуля)
 - 7.1. Рекомендуемая литература
 - 7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"
 - 7.3. Перечень информационных технологий
8. Материально-техническое обеспечение дисциплины (модуля)
9. Методические указания для обучающихся по освоению дисциплины (модуля)
10. Специальные условия освоения дисциплины обучающимися с инвалидностью и ограниченными возможностями здоровья



1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является:

- освоение технологии создания программного обеспечения на платформе Java.

Основными задачами изучения дисциплины являются:

- обеспечение студентов знаниями о методах и приемах разработки программного обеспечения на платформе Java,

- освоение на практике технологии разработки, тестирования и развертывания программного обеспечения на платформе Java.

Результаты изучения дисциплины направлены на достижение следующих индикаторов:

УК-1.1. Выполняет поиск информации, определяет критерии системного анализа поставленных задач.

УК-1.2. Использует критический анализ, систематизацию и обобщение информации для решения поставленных задач.

ПК-2.1. Обладает знаниями о методах и средствах сборки модулей и компонент программного обеспечения, о разработке процедур для развертывания программного обеспечения, миграции и преобразования данных, о создании программных интерфейсов; о методах и механизмах оценки и анализа функционирования средств и систем информационных технологий; о международных и профессиональных стандартах информационных технологий, о современных парадигмах и методологиях, инструментальных и вычислительных средствах.

ПК-2.2. Демонстрирует умения: применять методы и средства сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов; проводить проверку и оценку работоспособности программного продукта.

ПК-2.3. Имеет практический опыт (навыки): сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов; оценки работоспособности программного продукта.

ПК-3.1. Обладает знаниями о методах и средствах проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов.

ПК-3.2. Демонстрирует умения: разрабатывать требования к программному продукту, применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов.

ПК-3.3. Имеет практический опыт (навыки):

проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Цикл (раздел) ОПОП: К.М.01.ДВ.03.01

2.1 Требования к предварительной подготовке обучающегося:

Информатика

Алгоритмы и структуры данных

Объектно-ориентированное программирование на языке Java

2.2 Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:

Производственная практика (преддипломная практика)

Подготовка к процедуре защиты и защита выпускной квалификационной работы

Разработка мобильных приложений

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

УК-1: Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач

Знать:

методы и способы поиска информации, определения критериев системного анализа поставленных задач в области программирования на языке Java.

Уметь:



Рабочая программа дисциплины "Программирование на языке Java (научный семинар)" по направлению подготовки (специальности) 02.03.02 "Фундаментальная информатика и информационные технологии" направленности (профилю) Прикладное программирование и системы искусственного интеллекта ФГБОУ ВО «ЧелГУ»

стр. 4

выполнять поиск информации, определять критерии системного анализа поставленных задач в области программирования на языке Java.

Владеть:

навыком критического анализа, систематизации и обобщения информации применительно к программированию на языке Java.

ПК-2: Способность к осуществлению интеграции программных модулей и компонент и проверки работоспособности программного продукта на основе международных и профессиональных стандартов информационных технологий, современных парадигм и методологий, инструментальных и вычислительных средств, методов и механизмов оценки и анализа функционирования средств и систем информационных технологий

Знать:

о методах и средствах сборки модулей и компонент программ на языке Java; синтаксис языка, основные возможности библиотек.

Уметь:

проектировать иерархию классов, читать и писать программный код на языке Java; проводить проверку и оценку работоспособности программ.

Владеть:

навыками программирования на языке Java.

ПК-3: Способность к разработке требований и проектированию программного обеспечения на основе применения базовых математических знаний и информационных технологий при решении проектно-технических и прикладных задач

Знать:

методы и приемы разработки программного обеспечения на платформе Java

Уметь:

создавать простые приложения на основе современной ИТ-платформы с использованием библиотеки классов данной платформы и разработкой собственных классов.

Владеть:

технологией разработки, тестирования и развертывания программного обеспечения на платформе Java.

В результате освоения дисциплины обучающийся должен

3.1	Знать:
3.1.1	- методы и способы поиска информации, определения критериев системного анализа поставленных задач в области программирования на языке Java;
3.1.2	- синтаксис языка, основные возможности библиотек;
3.1.3	- методы и приемы разработки программного обеспечения на платформе Java.
3.2	Уметь:
3.2.1	- выполнять поиск информации, определять критерии системного анализа поставленных задач в области программирования на языке Java;
3.2.2	- проектировать иерархию классов, читать и писать программный код на языке Java;
3.2.3	- создавать простые приложения на основе современной ИТ-платформы с использованием библиотеки классов данной платформы и разработкой собственных классов.
3.3	Владеть:
3.3.1	- критического анализа, систематизации и обобщения информации применительно к программированию на языке Java;
3.3.2	- программирования на языке Java;
3.3.3	- разработки, тестирования и развертывания программного обеспечения на платформе Java.



4. ОБЪЕМ ДИСЦИПЛИНЫ (МОДУЛЯ)

Общая трудоемкость	5 ЗЕТ	
Часов по учебному плану	180	Виды контроля в семестрах: экзамены 8
в том числе		
аудиторные занятия	40	
самостоятельная работа	100,7	
часов на контроль	36	
контактная работа: 43,3		
ИКР: 3,3		

5. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Литература
	Раздел 1. Технология Java и ее применение.			
1.1	Технология Java и ее применение. Базовые сведения о Java. Характеристика. Особенности. Виртуальная машина Java. Платформа Java. Версии Java: Java SE, Java EE, Java ME, Android SKD. Обзор сред разработки. Синтаксис. Типы данных. Объектно-ориентированное программирование на Java. /Лек/	8	6	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2
1.2	Технология Java и ее применение. Базовые сведения о Java. Характеристика. Особенности. Виртуальная машина Java. Платформа Java. Версии Java: Java SE, Java EE, Java ME, Android SKD. Обзор сред разработки. Синтаксис. Типы данных. Объектно-ориентированное программирование на Java. /Ср/	8	30	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1
1.3	Объектно-ориентированное программирование на Java. /Лаб/	8	6	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2
	Раздел 2. Принципы объектно-ориентированного дизайна приложений			
2.1	Принципы объектно-ориентированного дизайна приложений. Обработка исключений. Коллекции. Базовый ввод-вывод. Модель обработки событий в JAVA. Графический интерфейс пользователя. Персистентность данных в объектно-ориентированных приложениях. Организация взаимодействия с базами данных. Многопоточное программирование в Java. Сетевое программирование. Средства пакета java.net. Лямбда функции и функциональное программирование на Java. /Лек/	8	10	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2
2.2	Функциональное программирование на Java. /Лаб/	8	6	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2
2.3	Принципы объектно-ориентированного дизайна приложений. Обработка исключений. Коллекции. Базовый ввод-вывод. Модель обработки событий в JAVA. Графический интерфейс пользователя. Персистентность данных в объектно-ориентированных приложениях. Организация взаимодействия с базами данных. Многопоточное программирование в Java. Сетевое программирование. Средства пакета java.net. Лямбда функции и функциональное программирование на Java. /Ср/	8	34,9	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1
	Раздел 3. Разработка веб-приложений и веб-сервисов на платформе Java			



Рабочая программа дисциплины "Программирование на языке Java (научный семинар)" по направлению подготовки (специальности) 02.03.02 "Фундаментальная информатика и информационные технологии" направленности (профилю) Прикладное программирование и системы искусственного интеллекта ФГБОУ ВО «ЧелГУ»

стр. 6

3.1	Разработка веб-приложений и веб-сервисов на платформе Java. Спецификация Java. Организация сборки проекта на Java. Архитектура приложения уровня предприятия. Servlet API. Разработка пользовательского интерфейса при помощи JSF. Технология ORM. Обзор EJB. Работа с Web Services. /Лек/	8	4	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2
3.2	Разработка веб-приложений и веб-сервисов на платформе Java /Лаб/	8	8	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2
3.3	Разработка веб-приложений и веб-сервисов на платформе Java. Спецификация Java. Организация сборки проекта на Java. Архитектура приложения уровня предприятия. Servlet API. Разработка пользовательского интерфейса при помощи JSF. Технология ORM. Обзор EJB. Работа с Web Services. /Ср/	8	35,8	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1
3.4	/Экзамен/	8	36	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2
Раздел 4. Иная контактная работа				
4.1	Индивидуальные консультации, текущий контроль /ИКР/	8	3,3	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2

6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

6.1. Перечень видов оценочных средств

Комплект задач для лабораторных работ
Проектное задание
Вопросы для экзамена
Практические задания для экзамена.

6.2. Типовые контрольные задания и иные материалы для текущей аттестации

Примеры задач для лабораторных работ

1. Объектно-ориентированное программирование на Java.

1.1 Класс Phone.

а) Создайте класс Phone, который содержит переменные number, model и weight.

б) Создайте три экземпляра этого класса.

в) Выведите на консоль значения их переменных.

г) Добавить в класс Phone методы: receiveCall, имеет один параметр – имя звонящего. Выводит на консоль сообщение "Звонит {name}". getNumber – возвращает номер телефона. Вызвать эти методы для каждого из объектов.

д) Добавить конструктор в класс Phone, который принимает на вход три параметра для инициализации переменных класса - number, model и weight.

е) Добавить конструктор, который принимает на вход два параметра для инициализации переменных класса - number, model.

ж) Добавить конструктор без параметров.

з) Вызвать из конструктора с тремя параметрами конструктор с двумя.

и) Добавьте перегруженный метод receiveCall, который принимает два параметра - имя звонящего и номер телефона звонящего. Вызвать этот метод.

к) Создать метод sendMessage с аргументами переменной длины. Данный метод принимает на вход номера телефонов, которым будет отправлено сообщение. Метод выводит на консоль номера этих телефонов.

л) Изменить класс Phone в соответствии с концепцией JavaBean. Смотрите решение задачи в видео 1, видео 2, видео 3, видео 4, видео 5.

1.2 Создать класс Person, который содержит:

а) поля fullName, age.

б) методы move() и talk(), в которых просто вывести на консоль сообщение - "Такой-то Person говорит".

в) Добавьте два конструктора - Person() и Person(fullName, age).

1.3 Создайте два объекта этого класса. Один объект инициализируется конструктором Person(), другой - Person(fullName, age).

Создать класс "Матрица". Класс должен иметь следующие поля:

1) двумерный массив вещественных чисел;

2) количество строк и столбцов в матрице.

Класс должен иметь следующие методы:



- 1) сложение с другой матрицей;
 - 2) умножение на число;
 - 3) вывод на печать;
 - 4) умножение матриц - по желанию.
- 1.4 Определить класс Reader, хранящий такую информацию о пользователе библиотеки: ФИО, номер читательского билета, факультет, дата рождения, телефон. Методы takeBook(), returnBook().
Разработать программу, в которой создается массив объектов данного класса. Перегрузить методы takeBook(), returnBook():
- takeBook, который будет принимать количество взятых книг. Выводит на консоль сообщение "Петров В. В. взял 3 книги".
 - takeBook, который будет принимать переменное количество названий книг. Выводит на консоль сообщение "Петров В. В. взял книги: Приключения, Словарь, Энциклопедия".
 - takeBook, который будет принимать переменное количество объектов класса Book (создать новый класс, содержащий имя и автора книги). Выводит на консоль сообщение "Петров В. В. взял книги: Приключения, Словарь, Энциклопедия".
- Аналогичным образом перегрузить метод returnBook(). Выводит на консоль сообщение "Петров В. В. вернул книги: Приключения, Словарь, Энциклопедия". Или "Петров В. В. вернул 3 книги". Решение
- 1.5 Даны два целых числа A и B. Выведите все числа от A до B включительно, в порядке возрастания, если $A < B$, или в порядке убывания в противном случае. Использовать рекурсию.
- 1.6 Создайте пример наследования, реализуйте класс Student и класс Aspirant, аспирант отличается от студента наличием некой научной работы.
- а) Класс Student содержит переменные: String firstName, lastName, group. А также, double averageMark, содержащую среднюю оценку.
 - б) Создать переменную типа Student, которая ссылается на объект типа Aspirant.
 - в) Создать метод getScholarship() для класса Student, который возвращает сумму стипендии. Если средняя оценка студента равна 5, то сумма 100 грн, иначе 80. Переопределить этот метод в классе Aspirant. Если средняя оценка аспиранта равна 5, то сумма 200 грн, иначе 180.
 - г) Создать массив типа Student, содержащий объекты класса Student и Aspirant. Вызвать метод getScholarship() для каждого элемента массива. Смотрите решение задачи в видео 1, видео 2.

2. Функциональное программирование на Java.

- 2.1 Напишите программу для вычисления приближенного значения корня уравнения $\cos x = x$. Вычисления проводите методом бисекции (последовательного деления промежутка, содержащего корень функции, пополам). Предложите программы на языке Java, написанные в традиционном императивном и функциональном стилях.
- 2.2 Напишите функцию, которая в массиве неотрицательных целых чисел находит максимальное значение суммы заданного количества соседних элементов. Например, требуется в массиве из 10 000 чисел найти 10 рядом стоящих чисел с максимальной суммой. Функция должна быть написана в функциональном стиле программирования.
- 2.2 Напишите программу для вычисления приближенного значения числа e по формуле для разложения e^x в ряд Тейлора. Предложите программы на языке Java, написанные в традиционном императивном и функциональном стилях.

3. Разработка веб-приложений и веб-сервисов на платформе Java

Типовое задание для проекта

1. Разработать программу для выполнения арифметической операции над двумя числами. Код должен позволить диалоговыми средствами последовательно ввести первое число, затем - второе число, затем - арифметическую операцию. Программа должна анализировать введенные данные на корректность, запрашивать у пользователя подтверждение на продолжение работы, если данные не корректны и завершать работу в случае отсутствия подтверждения. В результирующем диалоговом окне вывести результат вычисления введенной арифметической операции над введенными числами.
2. Дано целое число $n = 100$, а также первый член $a = 154$ и разность $d = 3$ арифметической прогрессии. Сформировать и вывести массив размера n , содержащий n первых членов данной прогрессии: $a, a+d, a+2d, a+3d, \dots$

6.3. Типовые контрольные вопросы и задания для промежуточной аттестации

Перечень типовых вопросов к экзамену:

1. История появления и развития платформы Java
2. Структура современной платформы Java
3. Возможности предоставляемые современной платформой Java



4. Языки программирования платформы Java. Обзор
5. Средства разработки под платформу Java. Обзор
6. Приложение Java. Разработка и развертывание
7. Возможности технологий Java
8. Основные инструменты платформы Java
9. Архитектура JVM
10. Пакетная организация классов. Объявление и доступность
11. Пакетная организация классов. Выполнение
12. Пакетная организация классов. Указание зависимых классов
13. Язык Java. История создания, этапы и логика развития.
14. Язык Java. Принципы лежащие в его основе.
15. Структура и синтаксис примитивной консольной программы Java
16. Соглашения о наименованиях
17. Документирование кода. Аннотации javadoc
18. Базовый Java. Система встроенных типов. Типобезопасность.
19. Базовый Java. Ссылочные типы и типы значений.
20. Базовый Java. Примитивные встроенные типы данных.
21. Базовый Java. Базовые управляющие конструкции. Синтаксис. Примеры использования.
22. Базовый Java. Базовые управляющие конструкции. Циклы. Рекомендации по проектированию.
23. Базовый Java. Аннотации.
22. Базовый Java. Ссылочные типы. Массивы. Разновидности. Синтаксис. Примеры использования.
23. Базовый Java. Ссылочные типы. Строки. Синтаксис. Примеры использования.
24. Базовый Java. Консольный ввод вывод.
25. Базовый Java. Базовые типы данных. Иерархия.
26. Базовый Java. Ссылочные типы. Классы - оболочки.
27. Базовый Java. Ссылочные типы. Класс. Общие характеристики.
28. Базовый Java. Ссылочные типы. Класс. Структура.
29. Базовый Java. Ссылочные типы. Класс. Поля (класса, экземпляра), константы.
30. Базовый Java. Ссылочные типы. Класс. Ограничение доступа.
31. Базовый Java. Ссылочные типы. Класс. Конструкторы.
32. Базовый Java. Ссылочные типы. Класс. Методы.
33. Базовый Java. Ссылочные типы. Класс. Статические методы и поля.
34. Базовый Java. Ссылочные типы. Класс. Модификатор final.
35. Базовый Java. Ссылочные типы. Класс. Абстрактные методы.
36. Базовый Java. Ссылочные типы. Класс. Модификаторы native и synchronized.
37. Базовый Java. Ссылочные типы. Класс. Логические блоки.
38. Базовый Java. Ссылочные типы. Класс. Перегрузка методов.
39. Базовый Java. Ссылочные типы. Класс. Методы с переменным числом параметров.
40. Базовый Java. Ссылочные типы. Класс. Рекомендации по проектированию.
41. Базовый Java. Ссылочные типы. Перечисления.
42. Базовый Java. Ссылочные типы. Интерфейсы.
43. Базовый Java. Наследование и полиморфизм. Общее понятие.
44. Базовый Java. Наследование и полиморфизм. Позднее связывание.
45. Базовый Java. Наследование и полиморфизм. Ссылки super и this.
46. Базовый Java. Наследование и полиморфизм. Классы и методы final.
47. Базовый Java. Наследование и полиморфизм. Абстрактные классы.
48. Базовый Java. Наследование и полиморфизм. Класс Object.
49. Базовый Java. Наследование и полиморфизм. Клонирование.
50. Базовый Java. Наследование и полиморфизм. Рекомендации по проектированию иерархии классов.
51. Параметризация. Общее понятие.
52. Параметризация. Параметризация классов.
53. Параметризация. Джокеры (метасимволы).
54. Параметризация. Параметризация методов.
55. Параметризация. Параметризация интерфейсов.
56. Исключения. Общее понятие.
57. Исключения. Механизм обработки исключительных ситуаций.
58. Исключения. Иерархия исключений.
59. Исключения. Иерархия исключений. Ошибки.
60. Исключения. Проверяемые исключения.
61. Исключения. Непроверяемые исключения.



- 62. Исключения. Продвижение возбужденного исключения.
- 63. Исключения. Способы обработки исключений. По месту возбуждения (пример)
- 65. Исключения. Способы обработки исключений. Обработка нескольких исключений
- 66. Исключения. Обработка исключений. Блок finally
- 67. Исключения. Возбуждение исключений пользователем. Оператор throw
- 68. Исключения. Обработка исключений и наследование
- 69. Исключения. Рекомендации по обработке исключений
- 70. Механизм утверждений (assertions).
- 71. Коллекции. Общие определения
- 72. Коллекции. Списки
- 73. Коллекции. Метасимвол в коллекциях
- 74. Коллекции. Интерфейс ListIterator
- 75. Коллекции. Интерфейс Comparator
- 76. Коллекции. Класс LinkedList и интерфейс Queue
- 77. Коллекции. Интерфейс Deque и класс ArrayDeque
- 78. Коллекции. Множества
- 79. Коллекции. Карты отображений
- 80. Коллекции. Унаследованные коллекции
- 81. Коллекции. Алгоритмы класса Collections

Пример билетов на экзамен

Билет 1

- 1. Базовый Java. Ссылочные типы. Класс. Поля (класса, экземпляра), константы.
- 2. Параметризация. Параметризация классов.

Задача

Найти количество всех прямоугольных параллелепипедов, объём которых равен заданному натуральному числу v ($1 \leq v \leq 100$)

и стороны которых выражены натуральными числами. При этом решения, которые получаются перестановкой размеров рёбер параллелепипеда считать разными.

Ввод

В первой строке ввода содержится число v

Вывод

Количество всех прямоугольных параллелепипедов.

Пример ввода

9

Пример вывода

6

Билет 20

- 1. Архитектура JVM
- 2. Базовый Java. Консольный ввод вывод.

Задача

На интервале $[2;n]$

найти натуральное число с максимальной суммой делителей.

Ввод

В первой строке ввода содержится число n ($2 \leq n \leq 1000$)

Вывод

В первой строке – максимальная сумма делителей.

Пример ввода

9

Пример вывода

15

(Максимальная сумма делителей у числа 8 $1+2+4+8=15$)

6.4. Критерии оценивания

Схема подведения итогов промежуточной аттестации в форме экзамена



Баллы, полученные за отдельные задания экзамена (тестирование, теоретический вопрос, практическое задание) суммируются.

Виды заданий	Баллы
Теоретический вопрос1	0-30
Теоретический вопрос2	0-30
Практическое задание	0-40
Всего: 0-100	

0-49 баллов – неудовлетворительно;
50-69 баллов - удовлетворительно;
70-89 баллов - хорошо;
90-100 баллов - отлично.

При подведении итогов промежуточной аттестации учитываются результаты текущей аттестации.

Виды заданий	Баллы
Лабораторный практикум	0-50
Проектное задание	0-50

0-49 баллов - неудовлетворительно (2);
50-69 баллов - удовлетворительно (3);
70-89 баллов - хорошо (4);
90-100 баллов - отлично (5).

Полученные за текущую аттестацию баллы могут либо повысить, либо понизить отметку за экзамен за счет вычисления среднего балла за экзамен и текущую аттестацию.

0-49 баллов - неудовлетворительно (2);
50-69 баллов - удовлетворительно (3);
70-89 баллов - хорошо (4);
90-100 баллов - отлично (5).

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

7.1. Рекомендуемая литература

7.1.1. Основная литература

	Авторы,	Заглавие	Издательство,	Ресурс
Л1.1	Государев И. Б.	Введение в веб-разработку на языке JavaScript: учебное пособие (https://e.lanbook.com/book/206588)	Санкт-Петербург : Лань, 2022	ЭБС
Л1.2	Бобырь М.В.	Программирование на языке Java. Практический курс: учебное пособие (https://znanium.ru/catalog/document?id=460819)	Москва : ООО "Научно-издательский центр ИНФРА-М", 2025	ЭБС
Л1.3	Спилкэ Л.	Java устранение проблем (https://e.lanbook.com/book/455345)	Москва : ДМК Пресс, 2023	ЭБС
Л1.4	Федоричев Л. А., Букунова О. В.	Реализация многопоточности в языке Java: учебное пособие для вузов (https://e.lanbook.com/book/457502)	Санкт-Петербург : Лань, 2025	ЭБС

7.1.2. Дополнительная литература

	Авторы,	Заглавие	Издательство,	Ресурс
Л2.1	Дэвид Х.	Java EE 7 и сервер приложений GlassFish 4 (http://e.lanbook.com/books/element.php?pl1_id=69962)	Москва : ДМК Пресс, 2016	ЭБС
Л2.2	Коузен К.	Современный Java: рецепты программирования (https://e.lanbook.com/book/116121)	Москва : ДМК Пресс, 2018	ЭБС

7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	Школа программиста : сайт / С. Н. Беляев, ККДП. – Красноярск, 2003-. – URL: http://acmp.ru/ , свободный. – Текст : электронный. http://acmp.ru/
----	---



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)

Рабочая программа дисциплины "Программирование на языке Java (научный семинар)" по направлению подготовки (специальности) 02.03.02 "Фундаментальная информатика и информационные технологии" направленности (профилю) Прикладное программирование и системы искусственного интеллекта ФГБОУ ВО «ЧелГУ»

стр. 11

7.3 Перечень информационных технологий

7.3.1 Программное обеспечение

LMS Moodle

Java

OpenOffice

Open Project

7.3.2 Профессиональные базы данных и информационно-справочные системы

Электронный каталог научной библиотеки ЧелГУ : база данных / Челяб. гос. ун-т. – Челябинск, 1992. – Текст : электронный.

eLIBRARY.RU : научная электронная библиотека [научной периодики на русском языке]. – Москва, [1999-]. – Доступ к полным текстам после регистрации из сети ЧелГУ. – URL: <http://elibrary.ru/defaultx.asp>. – Текст : электронный.

Moodle : система дистанционного обучения : [база данных] / Челяб. гос. ун-т. – Челябинск, [б.г.]. – Доступ из сети ЧелГУ или, после регистрации из сети ун-та, из любой точки, имеющей доступ в интернет. – URL: <http://moodle.uio.csu.ru/login/index.php>. – Текст : электронный.

Научная библиотека Челябинского государственного университета : [сайт] / Челяб. гос. ун-т. – Челябинск, [2001-]. – URL: <http://www.lib.csu.ru/>, свободный. – Загл. с экрана. – Текст : электронный.

Интернет университет информационных технологий. – Электрон. дан. – URL: <http://www.intuit.ru/>. – Текст : электронный.

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Для реализации дисциплины используются учебные аудитории для проведения занятий лекционного типа, лабораторных занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы.

Учебные аудитории укомплектованы специализированной мебелью.

Для проведения занятий лекционного типа используется переносное и / или стационарное мультимедийное оборудование (экран, ноутбук или десктоп, проектор). Для обеспечения тематической иллюстрации занятий лекционного типа в образовательном процессе используются цифровые образовательные ресурсы (мультимедийные презентации по всем темам программы).

Для проведения практических работ и самостоятельной работы используется компьютерный класс с выходом в Интернет и доступом в электронную информационно-образовательную среду университета, с установленным программным обеспечением, указанным в п.7.3.1.

Для самостоятельной работы обучающихся используется также читальный зал научной библиотеки ЧелГУ (первый корпус ЧелГУ) с доступом к различной справочной литературе, энциклопедиям, библиографическим и полнотекстовым базам данных, информационным Интернет-ресурсам.

9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

При изучении данной дисциплины используются лекционные занятия и самостоятельная работа студента. На лекционных занятиях преподаватель излагает основное содержание тем программы. Проработку теоретического материала желательно проводить как после каждого занятия, так и по завершению темы с использованием рекомендованной литературы и пособий. Это позволит связать воедино полученные сведения и составить цельную картину.

На лекционных занятиях рассматриваются основные методы и приемы создания и отладки параллельных приложений. Крайне желательно приступать к выполнению домашних заданий после усвоения архитектуры и базовых понятий отрабатываемых технологий и следовать рекомендациям по их решению. Задания следует выполнять в том порядке, в каком они предлагаются для решения. Рекомендуется перед каждым практическим занятием выполнить домашнее задание, что позволит лучше усвоить предыдущий материал, и изучить лекционный материал по предстоящей теме. Студенту следует быть активным участником занятия. Умение обосновывать свою точку зрения, найти компромиссное решение в этически выдержанной дискуссии не только важно для лучшего усвоения материала, но и ценится в реальной жизни.

Важным моментом при изучении любой дисциплины является организация самостоятельной работы. При освоении материала не следует стремиться к механическому запоминанию приведенных определений, формулировок и положений, если требования к прямо не указывают на это. Вполне эффективной может оказаться попытка понять суть явления, выработать свое отношение к нему, опираясь на материал, содержащийся в рекомендованной



литературе. Сказанное особенно эффективно, когда речь идет о таких требованиях, как «понимает» или «имеет представление». Напротив, если требование к деятельности «должен уметь», то рекомендуется поупражняться в соответствующем виде деятельности. Все это имеет непосредственное отношение к подготовке к практическим занятиям.

При проведении лабораторных работ студентам необходимо обращать внимание на тот теоретический (лекционный) материал, который используется в конкретном задании.

При обучении лиц с ограниченными возможностями здоровья электронное обучение, дистанционные образовательные технологии предусматривают возможность приема-передачи информации в доступных для них формах.

Реализация дисциплины с применением электронного обучения, дистанционных образовательных технологий (далее – ЭО, ДОТ) осуществляется на основании «Положения о реализации основных и дополнительных образовательных программ с применением электронного обучения и дистанционных образовательных технологий в федеральном государственном бюджетном образовательном учреждении высшего образования «Челябинский государственный университет», «Положения о порядке зачета обучающимися по основным профессиональным образовательным программам высшего образования в ФГБОУ ВО «ЧелГУ» результатов освоения в организациях, осуществляющих образовательную деятельность, учебных предметов, курсов, дисциплин (модулей), практик, дополнительных образовательных программ» посредством электронной информационно-образовательной среды ФГБОУ ВО «ЧелГУ». В исключительных случаях (форс-мажор и т.п.) при реализации образовательной деятельности с применением ЭО, ДОТ могут применять компоненты, не входящие в перечень электронной информационно-образовательной среды.

10. СПЕЦИАЛЬНЫЕ УСЛОВИЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ ОБУЧАЮЩИМИСЯ С ИНВАЛИДНОСТЬЮ И ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Освоение дисциплины инвалидами и лицами с ограниченными возможностями здоровья осуществляется с использованием специальных технических средств и информационных технологий, предоставляемых Ресурсным учебно-методическим центром по обучению инвалидов и лиц с ограниченными возможностями здоровья ЧелГУ по запросу обучающегося (мобильные специальные технические средства для лиц с нарушениями зрения и с нарушением слуха, ассистивные информационные технологии).

При необходимости для обучающихся с нарушениями зрения на рабочих местах для проведения практических или лабораторных занятий устанавливается специальное программное обеспечение (программа речевой навигации, речевые синтезаторы, экранные лупы).

В учебные аудитории обеспечивается беспрепятственный доступ для обучающихся с инвалидностью и с ограниченными возможностями здоровья. В каждой аудитории, где обучаются инвалиды и лица с ограниченными возможностями здоровья, предусматривается соответствующее количество мест для обучающихся с учетом нарушений их здоровья.

Для освоения дисциплины инвалидам и лицам с ограниченными возможностями здоровья предоставляется доступ к печатным источникам, имеющимся в научной библиотеке ЧелГУ, с помощью специальных технических средств; доступ с помощью специальных технических и программных средств к электронным источникам, представленным в форме электронного документа в фонде научной библиотеки ЧелГУ или электронно-библиотечных системах.

Учебно-методические материалы для обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья предоставляются в формах, адаптированных к ограничениям их здоровья и особенностям восприятия информации.

Для инвалидов и лиц с ограниченными возможностями здоровья освоение дисциплины может быть частично или полностью осуществлено с использованием дистанционных образовательных технологий.

При проведении промежуточной аттестации по дисциплине обучающимся с инвалидностью и с ограниченными возможностями здоровья обеспечивается по их заявлению предоставление в доступной форме в зависимости от их индивидуальных особенностей инструкции о порядке проведения промежуточной аттестации, оценочных средств и возможности ответов на задания (письменно на бумаге, набор ответов на компьютере, письменно шрифтом Брайля, с использованием услуг ассистента, устно).

При проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование предоставленных ЧелГУ или собственных технических средств, необходимых им в связи с их индивидуальными особенностями. При необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на задания, процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

