Министерство науки и высшего образования Российской Федерации

Муромский институт (филиал)

федерального государственного бюджетного образовательного учреждения высшего образования

«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
(МИ ВлГУ)

Кафедра *ПИн*

«УТВЕРЖДАЮ»
Заместитель директора по УР
_____Д.Е. Андрианов
19.05.2026

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Научно-исследовательская работа студента

Направление подготовки *09.04.04 Программная инженерия*

Профиль подготовки *Технологии разработки интеллектуальных систем*

Семестр	Трудоемкость, час./зач. ед.	Лекции, час.	Практические занятия, час.	Лабораторные работы, час.	Консультация, час.	Контроль, час.	Всего (контактная работа), час.	СРС, час.	Форма промежуточного контроля (экз., зач., зач. с оц.)
1	252 / 7	16	32		1,6	0,25	49,85	202,15	Зач. с оц.
2	180 / 5	16	32		1,6	0,25	49,85	130,15	Зач. с оц.
3	252 / 7	16	32		1,6	2,25	51,85	200,15	Зач. с оц.
Итого	684 / 19	48	96		4,8	2,75	151,55	532,45	

Муром, 2026 г.

1. Цель освоения дисциплины

Целью научно-исследовательской работы является использование теоретических знаний, полученных студентами при изучении учебных дисциплин направления 09.04.04 "Программная инженерия", для осуществления самостоятельной научной-исследовательской деятельности на основе приобретения и закрепления практических навыков и компетенций.

Научно-исследовательская работа в семестре выполняется студентом-магистрантом под руководством научного руководителя. Направление научно исследовательских работ магистранта определяется в соответствии с магистерской программой и темой магистерской диссертации.

Задачами НИР является:

- обеспечение становления профессионального научно-исследовательского мышления магистрантов, формирование у них четкого представления об основных профессиональных задачах, способах их решения;
- формирование умений использовать современные технологии сбора информации, обработки и интерпретации полученных экспериментальных и эмпирических данных, владение современными методами исследований;
- формирование готовности проектировать и реализовывать в образовательной практике новое содержание учебных программ, использовать инновационные образовательные технологии;
- обеспечение готовности к профессиональному самосовершенствованию, развитию инновационного мышления и творческого потенциала, профессионального мастерства;
- самостоятельное формулирование и решение задач, возникающих в ходе научно-исследовательской и педагогической деятельности и требующих углубленных профессиональных знаний;
- проведение библиографической работы с привлечением современных информационных технологий.

2. Место дисциплины в структуре ОПОП ВО

Изучение дисциплины "Научно-исследовательская работа" базируется на изучении "Анализ больших данных", "Современные алгоритмы обработки данных".

3. Планируемые результаты обучения по дисциплине

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции	Результаты обучения по дисциплине	
УК-6 Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки	УК-6.1 Оценивает свои ресурсы и их пределы (личностные, ситуативные, временные), оптимально их использует для успешного выполнения порученного задания	методами оценки ресурсов и их пределов, оптимально их использовать для успешного выполнения порученного задания (УК-6.1)	тест,практическое задание
ОПК-2 Способен разрабатывать оригинальные алгоритмы и программные средства, в том	ОПК-2.2 Разрабатывает алгоритмы и проводит исследования в целях совершенствования эффективности работы программного обеспечения	алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения	тест,практическое задание

числе с использованием современных интеллектуальных технологий, для решения профессиональных задач		профессиональных задач (ОПК-2.2)	
	ОПК-2.3 Разрабатывает и обеспечивает программную реализацию эффективных алгоритмов решения сформулированных задач с использованием современных языков программирования	реализует алгоритмы для решения практических и научно-исследовательских задач с использованием современных языков программирования (ОПК-2.3)	
ОПК-6 Способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности	ОПК-6.1 Осуществляет информационный поиск и использует новые знания в своей предметной области	самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности (ОПК-6.1)	тест,практическое задание
ПК-7 Владение методами научных исследований и навыками их проведения	ПК-7.1 Демонстрирует знания принципов проведения научных исследований	владеть методами научных исследований и навыками их проведения (ПК-7.1)	тест,практическое задание

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 19 зачетных единиц, 684 часа.

4.1. Форма обучения: очная

Уровень базового образования: высшее.

Срок обучения 2г.

4.1.1. Структура дисциплины

№ п\п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Основы научных исследований	1	10	26						72	Устный опрос
2	Методы исследований в программной инженерии	1	6	6						130,15	Практическая работа
Всего за семестр		252	16	32				1,6	0,25	202,15	Зач. с оц.
3	Методы исследований в программной инженерии	2	6	18						60	Практическая работа
4	Инструменты и технологии для исследований	2	10	14						70,15	Практическая работа
Всего за семестр		180	16	32				1,6	0,25	130,15	Зач. с оц.
5	Инструменты и технологии для исследований	3	2	8						60	Практическая работа
6	Прикладные исследования в программной инженерии	3	14	24						140,15	Практическая работа
Всего за семестр		252	16	32			+	1,6	2,25	200,15	Зач. с оц.
Итого		684	48	96				4,8	2,75	532,45	

4.1.2. Содержание дисциплины

4.1.2.1. Перечень лекций

Семестр 1

Раздел 1. Основы научных исследований

Лекция 1.

Введение в научно-исследовательскую деятельность. Методология научного исследования (2 часа).

Лекция 2.

Выбор и формулировка темы исследования. Научный руководитель и работа в научном коллективе (2 часа).

Лекция 3.

Этика научных исследований. Поиск и анализ научных источников литературы (2 часа).

Лекция 4.

Систематические обзоры и мета-анализ литературы. Библиографические менеджеры. Написание научного текста (2 часа).

Лекция 5.

Экспериментальные исследования. Качественные и количественные методы (2 часа).

Раздел 2. Методы исследований в программной инженерии

Лекция 6.

Исследования в области разработки ПО (2 часа).

Лекция 7.

Исследования в машинном обучении (2 часа).

Лекция 8.

Эмпирическая программная инженерия (2 часа).

Семестр 2

Раздел 3. Методы исследований в программной инженерии

Лекция 9.

Статистическая обработка данных (2 часа).

Лекция 10.

Визуализация данных в исследованиях (2 часа).

Лекция 11.

Анализ больших данных в программной инженерии (2 часа).

Раздел 4. Инструменты и технологии для исследований

Лекция 12.

Подготовка научной статьи (2 часа).

Лекция 13.

Рецензирование и доработка статьи (2 часа).

Лекция 14.

Публикация в Scopus/WoS (2 часа).

Лекция 15.

Подготовка презентации исследования (2 часа).

Лекция 16.

Подготовка и участие в конференциях (2 часа).

Семестр 3

Раздел 5. Инструменты и технологии для исследований

Лекция 17.

LaTeX для научной публикации (2 часа).

Раздел 6. Прикладные исследования в программной инженерии

Лекция 18.

Jupyter Notebook для исследований (2 часа).

Лекция 19.

Управление кодом в научных проектах (2 часа).

Лекция 20.

Автоматизация исследований (2 часа).

Лекция 21.

Исследования в DevOps и SRE, исследования в HCI (2 часа).

Лекция 22.

Исследования в тестировании ПО. Исследования в кибербезопасности. Исследование в распределенных системах (2 часа).

Лекция 23.

Написание магистерской диссертации и подготовка к защите (2 часа).

Лекция 24.

Научная карьера и ее этапы (2 часа).

4.1.2.2. Перечень практических занятий

Семестр 1

Раздел 1. Основы научных исследований

Практическое занятие 1

Анализ современного состояния проблем предметной области. Формулировка гипотезы, целей и задач исследования (2 часа).

Практическое занятие 2

Разработка плана научного исследования. Сравнение методологий научного исследования (2 часа).

Практическое занятие 3

Разбор кейсов научного плагиата и способы их избежания (2 часа).

Практическое занятие 4

Поиск статей Scopus и WoS по тематике исследования. Анализ цитируемости статьи (2 часа).

Практическое занятие 5

Написание аннотации к научной статье (2 часа).

Практическое занятие 6

Критический разбор статьи (2 часа).

Практическое занятие 7

Проведение обзора литературы (2 часа).

Практическое занятие 8

Планирование A/B-теста для сравнения двух алгоритмов (2 часа).

Практическое занятие 9

Проведение нагрузочного тестирования (2 часа).

Практическое занятие 10

Анализ воспроизводимости исследования (2 часа).

Практическое занятие 11

Разработка метрик ПО (2 часа).

Практическое занятие 12

Сравнение алгоритмов сортировки (2 часа).

Практическое занятие 13

Статистический анализ данных набора данных (2 часа).

Раздел 2. Методы исследований в программной инженерии

Практическое занятие 14

Визуализация результатов исследования (2 часа).

Практическое занятие 15

Построение корреляционной матрицы и ее анализ (2 часа).

Практическое занятие 16

Кластерный анализ данных. Анализ временных рядов (2 часа).

Семестр 2

Раздел 3. Методы исследований в программной инженерии

Практическое занятие 17

Автоматизация сбора данных (2 часа).

Практическое занятие 18

Реализация разработанных в ходе исследования алгоритмов (2 часа).

Практическое занятие 19

Оптимизация кода для исследования. Сравнение библиотек машинного обучения (2 часа).

Практическое занятие 20

Разработка скрипта для обработки логов (2 часа).

Практическое занятие 21

Создание репозитория для научного проекта. Ведение истории изменений (2 часа).

Практическое занятие 22

Настройка CI/CD для автоматического тестирования (2 часа).

Практическое занятие 23

Анализ истории Git-репозитория (2 часа).

Практическое занятие 24

Совместная работа над научным проектом (2 часа).

Практическое занятие 25

Подготовка статьи в LaTex (2 часа).

Раздел 4. Инструменты и технологии для исследований

Практическое занятие 26

Оформление библиографии в BibTeX (2 часа).

Практическое занятие 27

Написание тезисов конференции. Редактирование научного текста. Перевод тезисов на английский язык (2 часа).

Практическое занятие 28

Анализ инцидентов: разбор логов. Сравнение методов мониторинга. Исследование отказоустойчивости системы (2 часа).

Практическое занятие 29

Оптимизация времени сборки проекта (2 часа).

Практическое занятие 30

Анализ эффективности алертов (2 часа).

Практическое занятие 31

Сравнений фреймворков автотестов (2 часа).

Практическое занятие 32

Генерация тестовых данных. Мутационное тестирование (2 часа).

Семестр 3

Раздел 5. Инструменты и технологии для исследований

Практическое занятие 33

Фаззинг API (2 часа).

Практическое занятие 34

Исследование эффективности тест-кейсов (2 часа).

Практическое занятие 35

Анализ уязвимостей в коде (2 часа).

Практическое занятие 36

Атака на веб-приложения (2 часа).

Раздел 6. Прикладные исследования в программной инженерии

Практическое занятие 37

Исследование методов шифрования (2 часа).

Практическое занятие 38

Анализ сетевого трафика (2 часа).

Практическое занятие 39

Разрабокта PoC-эксплоита (2 часа).

Практическое занятие 40

Сравнение моделей машинного обучения. Анализ предвзятости алгоритмов (2 часа).

Практическое занятие 41

Интерпретация ML-моделей (2 часа).

Практическое занятие 42

Генерация синтетических данных. Оптимизация гиперпараметров (2 часа).

Практическое занятие 43

Разработка презентации исследования. Симуляция защиты исследования (2 часа).

Практическое занятие 44

Анализ рецензий на статью (2 часа).

Практическое занятие 45

Подготовка демонстрации проекта. Сравнение IDE для научной работы (2 часа).

Практическое занятие 46

Исследование энергоэффективности алгоритмов (2 часа).

Практическое занятие 47

Анализ кода open-source проектов (2 часа).

Практическое занятие 48

Проектирование мини-курса по теме исследования. Разработка мини-курса по теме исследования (2 часа).

4.1.2.3. Перечень лабораторных работ

Не планируется.

4.1.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы

Перечень тем, вынесенных на самостоятельное изучение:

1. Анализ современных тенденций в DevOps.
2. Сравнение алгоритмов машинного обучения для классификации текста.
3. Исследование методов оптимизации SQL-запросов.
4. Разработка метрик оценки качества кода.
5. Анализ уязвимостей в веб-приложениях.
6. Исследование методов тестирования блокчейн-приложений.
7. Сравнение фреймворков для автоматизации тестирования.
8. Анализ производительности микросервисной архитектуры.
9. Разработка системы мониторинга для облачных сервисов.
10. Исследование методов борьбы с техническим долгом.
11. Анализ алгоритмов сжатия данных.
12. Сравнение методов обработки естественного языка (NLP).
13. Исследование методов анонимизации данных.
14. Оптимизация работы с большими данными в реальном времени.
15. Анализ алгоритмов распределенного хранения данных.
16. Разработка рекомендательной системы на основе ML.
17. Исследование методов детектирования аномалий в данных.
18. Сравнение методов шифрования в современных приложениях.
19. Анализ методов повышения отказоустойчивости систем.
20. Исследование энергоэффективности алгоритмов.
21. Разработка системы логирования и анализа ошибок.
22. Сравнение методов балансировки нагрузки в веб-приложениях.
23. Анализ методов улучшения UX/UI в мобильных приложениях.
24. Исследование методов ускорения загрузки веб-страниц.
25. Разработка системы автоматического ревью кода.
26. Сравнение подходов к контейнеризации приложений.

27. Анализ методов безопасного хранения паролей.
28. Исследование алгоритмов поиска в графах.
29. Разработка системы предсказания сбоев в ПО.
30. Сравнение методов обработки потоковых данных.
31. Анализ методов защиты от DDoS-атак.
32. Исследование алгоритмов распознавания изображений.
33. Разработка системы автоматического документирования кода.
34. Сравнение методов развертывания ML-моделей.
35. Анализ методов кэширования в распределенных системах.
36. Исследование методов улучшения производительности баз данных.
37. Разработка системы анализа тональности текста.
38. Сравнение методов синхронизации в многопоточных приложениях.
39. Анализ методов обнаружения утечек памяти.
40. Исследование алгоритмов генерации тестовых данных.
41. Разработка системы автоматического исправления кода.
42. Сравнение методов аутентификации в веб-приложениях.
43. Анализ методов оптимизации работы с памятью.
44. Исследование алгоритмов сжатия изображений.
45. Разработка системы анализа производительности API.
46. Сравнение методов обработки событий в реальном времени.
47. Анализ методов защиты от SQL-инъекций.
48. Исследование алгоритмов маршрутизации в сетях.
49. Разработка системы предсказания нагрузки на сервер.
50. Сравнение методов развертывания приложений в облаке.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

4.1.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР

Не планируется.

4.1.2.6. Примерный перечень тем курсовых работ (проектов)

1. Разработка и анализ алгоритма оптимизации запросов в базах данных.
2. Сравнительный анализ фреймворков для автоматизированного тестирования веб-приложений.
3. Разработка системы мониторинга производительности микросервисной архитектуры.
4. Исследование методов обнаружения уязвимостей в исходном коде.
5. Оптимизация работы с большими данными.
6. Разработка рекомендательной системы на основе машинного обучения.
7. Анализ методов повышения отказоустойчивости распределенных систем.
8. Сравнение алгоритмов сжатия данных для различных типов файлов.
9. Разработка системы автоматического документирования кода.
10. Исследование методов балансировки нагрузки в облачных сервисах.
11. Разработка чат-бота с использованием NLP и нейронных сетей.
12. Анализ методов защиты веб-приложений от DDoS-атак.
13. Оптимизация времени сборки и развертывания CI/CD-пайплайнов.
14. Разработка системы анализа тональности текста в социальных сетях.
15. Исследование методов анонимизации данных в аналитических системах.
16. Разработка инструмента для визуализации архитектуры ПО.
17. Сравнение методов контейнеризации приложений.
18. Исследование алгоритмов маршрутизации в IoT-устройствах.
19. Разработка системы предсказания нагрузки на сервер с использованием ML.
20. Анализ методов улучшения UX/UI в мобильных приложениях.

5. Образовательные технологии

В процессе изучения дисциплины применяется контактная технология преподавания (за исключением самостоятельно изучаемых студентами вопросов). При проведении практических работ применяется имитационный или симуляционный подход. Шаги решения задач студентам демонстрируются при помощи мультимедийной техники. В дальнейшем студенты самостоятельно решают аналогичные задания.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

Фонды оценочных материалов (средств) приведены в приложении.

7. Учебно-методическое и информационное обеспечение дисциплины.

7.1. Основная учебно-методическая литература по дисциплине

1. Шуваев А.В. Программная инженерия : учебное пособие для магистрантов направления подготовки 09.04.02 – Информационные системы и технологии / Шуваев А.В.. — Ставрополь : Ветеран, 2020. — 84 с. - https://www.iprbookshop.ru/121736.html
2. Бондаренко И.С. Научно-исследовательская работа : методические указания к подготовке материалов для участия в конференц-неделе / Бондаренко И.С., Темкин И.О.. — Москва : Издательский Дом МИСиС, 2018. — 40 с. - https://www.iprbookshop.ru/92584.html

7.2. Дополнительная учебно-методическая литература по дисциплине

1. Шишикин, В. Г. Научно-исследовательская и практическая работа студентов : учебно-методическое пособие / В. Г. Шишикин, Е. В. Никитенко. — Новосибирск : Новосибирский государственный технический университет, 2019. - https://www.iprbookshop.ru/98773.html

7.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

В образовательном процессе используются информационные технологии, реализованные на основе информационно-образовательного портала института (www.mivlgu.ru/iop), и инфокоммуникационной сети института:

- предоставление учебно-методических материалов в электронном виде;
- взаимодействие участников образовательного процесса через локальную сеть института и Интернет;
- предоставление сведений о результатах учебной деятельности в электронном личном кабинете обучающегося.

Информационные справочные системы:

- электронная библиотечная системы "IPRBooks" (http://www.iprbookshop.ru/);

Программное обеспечение:

LibreOffice (Mozilla Public License v2.0)

Diagram Designer (Свободное программное обеспечение)

PostgreSQL (Лицензия PostgreSQL)

7.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

iprbookshop.ru

mivlgu.ru/iop

8. Материально-техническое обеспечение дисциплины

Лаборатория программирования и баз данных

12 шт. компьютеров Intel Core i5-10150 3,70 GHz/ 16Gb(DDr4)/ SSD-150Gb / Haff 23,8'; проектор ACER P1100 DLP Projector EMEA; экран проекционный настенный DRAPPER Apex STAR; маршрутизатор Gigabit Switch TEG-S16S; плоттер HP Design Jet T610. Маркерная доска. Доступ к сети Интернет.

9. Методические указания по освоению дисциплины

Для успешного освоения теоретического материала обучающийся: знакомится со списком рекомендуемой основной и дополнительной литературы; уточняет у преподавателя, каким дополнительным пособиям следует отдать предпочтение; ведет конспект лекций и прорабатывает лекционный материал, пользуясь как конспектом, так и учебными пособиями.

о выполнения практических работ обучающийся изучает соответствующий раздел теории. Перед занятием студент знакомится с описанием заданий для выполнения работы, внимательно изучает содержание и порядок проведения практической работы. Практическая работа проводятся в компьютерном классе. Обучающиеся выполняют индивидуальную задачу компьютерного моделирования в соответствии с заданием на практическую работу. Полученные результаты исследований сводятся в отчет и защищаются по традиционной методике в классе на следующем практическом занятии. Необходимый теоретический материал, индивидуальное задание, шаги выполнения практической работы и требование к отчету приведены в методических указаниях, размещенных на информационно-образовательном портале института.

Самостоятельная работа оказывает важное влияние на формирование личности будущего специалиста, она планируется обучающимся самостоятельно. Каждый обучающийся самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием дисциплины. Он выполняет внеаудиторную работу и изучение разделов, выносимых на самостоятельную работу, по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Курсовая работа выполняется в соответствии с методическими указаниями на курсовую работу. Обучающийся выбирает одну из указанных в перечне тем курсовых работ, исходя из своих интересов, наличия соответствующих литературных и иных источников. В ходе выполнения курсовой работы преподаватель проводит консультации обучающегося. На заключительном этапе обучающийся оформляет пояснительную записку к курсовой работе и выполняет ее защиту в присутствии комиссии из преподавателей кафедры.

Форма заключительного контроля при промежуточной аттестации – зачет с оценкой. Для проведения промежуточной аттестации по дисциплине разработаны фонд оценочных средств и балльно-рейтинговая система оценки учебной деятельности студентов. Оценка по дисциплине выставляется в информационной системе и носит интегрированный характер, учитывающий результаты оценивания участия студентов в аудиторных занятиях, качества и своевременности выполнения заданий в ходе изучения дисциплины и промежуточной аттестации.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению *09.04.04 Программная инженерия* и профилю подготовки *Технологии разработки интеллектуальных систем*

Рабочую программу составил *к.т.н. Белякова А.С.*_____

Программа рассмотрена и одобрена на заседании кафедры *ПИн*

протокол № 28 от 05.05.2026 года.

Заведующий кафедрой *ПИн* _____*Кульков Я.Ю.*

(Подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии факультета

протокол № 9 от 12.05.2026 года.

Председатель комиссии ФИТР _____ *Кутарова Е.И.*

(Подпись) (Ф.И.О.)

Приложение

Фонд оценочных материалов (средств) по дисциплине

Научно-исследовательская работа студента

1. Оценочные материалы для проведения текущего контроля успеваемости по дисциплине

1. Какой метод НЕ относится к эмпирическим исследованиям в программной инженерии?

A) A/B-тестирование
B) Нагрузочное тестирование
C) Систематический обзор литературы
D) Анализ репозиториев GitHub
E) Опрос разработчиков

2. Какой инструмент используется для автоматического тестирования воспроизводимости исследования?

A) Jupyter Notebook
B) Docker
C) LaTeX
D) Zotero
E) Prometheus

3. Какой показатель НЕ используется для оценки научных публикаций?

A) Импакт-фактор
B) Индекс Хирша (h-index)
C) Количество звезд на GitHub
D) Число цитирований
E) Quartile журнала

4. Какой метод применяется для анализа уязвимостей в исходном коде?

A) Fuzzing
B) Метод Монте-Карло
C) Линейная регрессия
D) Кросс-валидация
E) SWOT-анализ

5. Какой раздел научной статьи описывает проведенные эксперименты?

A) Введение (Introduction)
B) Методология (Methodology)
C) Результаты (Results)
D) Обсуждение (Discussion)
E) Заключение (Conclusion)

Общее распределение баллов текущего контроля по видам учебных работ для студентов

Рейтинг-контроль 1	устный опрос, 1 тест	до 20 баллов
Рейтинг-контроль 2	устный опрос, 1 тест	до 20 баллов
Рейтинг-контроль 3	устный опрос, 1 тест	до 40 баллов
Посещение занятий студентом	контроль посещаемости	до 16 баллов

Дополнительные баллы (бонусы)	за своевременную защиту всех практических работ	4
Выполнение семестрового плана самостоятельной работы	нет	0

2. Промежуточная аттестация по дисциплине

Перечень вопросов к экзамену / зачету / зачету с оценкой.
Перечень практических задач / заданий к экзамену / зачету / зачету с оценкой (при наличии)

1. Какой из перечисленных этапов НЕ входит в стандартный цикл научного исследования?

A) Постановка проблемы

B) Сбор данных

C) Написание кода

D) Анализ результатов

E) Публикация выводов

2. Какой метод чаще всего используется для проверки статистической значимости результатов?

A) Метод Монте-Карло

B) t-критерий Стьюдента

C) SWOT-анализ

D) Регрессионный анализ

E) Факторный анализ

3. Какой инструмент НЕ используется для визуализации научных данных?

A) Matplotlib

B) Tableau

C) LaTeX

D) Seaborn

E) Plotly

4. Какой раздел научной статьи содержит сравнение с работами других авторов?

A) Аннотация

B) Введение

C) Методология

D) Обзор литературы

E) Заключение

5. Какой тип исследования НЕ подходит для изучения причинно-следственных связей?
A) Эксперимент

B) Наблюдение

C) Опрос

D) Кейс-стади

E) Мета-анализ

6. Какой показатель оценивает влияние научного журнала?
A) Индекс Хирша

B) Импакт-фактор

C) Индекс цитирования

D) Коэффициент корреляции

E) Уровень достоверности

7. Какой метод НЕ используется для анализа качества кода?
A) Статический анализ

B) Рефакторинг

C) Мутационное тестирование

D) A/B-тестирование

E) Метрики сложности

8. Какой подход применяется для автоматического поиска уязвимостей в коде?
A) Фаззинг

B) Кросс-валидация

C) Кластерный анализ

D) Регрессионное тестирование

E) Моделирование

9. Какой раздел статьи описывает ограничения исследования?
A) Аннотация

B) Введение

C) Методология

D) Результаты

E) Обсуждение

10. Какой метод НЕ относится к машинному обучению?
A) Метод k-ближайших соседей

B) Дерево решений

C) Линейная регрессия

D) Факторный анализ

E) Нейронные сети

11. Какой инструмент используется для управления библиографией?
A) Docker

B) Zotero

C) Jupyter

D) Prometheus

E) Git

12. Какой метод применяется для анализа производительности алгоритмов?
A) Анализ временной сложности

B) Регрессионный анализ

C) Факторный анализ

D) Кластерный анализ

E) SWOT-анализ

13. Какой тип статьи содержит обзор существующих исследований?
A) Оригинальная статья

B) Обзорная статья

C) Короткое сообщение

D) Технический отчет

E) Мета-анализ

14. Какой метод НЕ используется для сбора данных в качественных исследованиях?
A) Интервью

B) Наблюдение

C) Эксперимент

D) Опрос

E) Анализ документов

15. Какой показатель оценивает научную продуктивность исследователя?
A) Импакт-фактор

B) Индекс Хирша

C) Уровень достоверности

D) Коэффициент корреляции

E) Количество публикаций

16. Какой метод применяется для тестирования алгоритмов машинного обучения?
A) Кросс-валидация

B) Фаззинг

C) Регрессионное тестирование

D) Нагрузочное тестирование

E) Модульное тестирование

17. Какой раздел научной статьи содержит интерпретацию результатов?
A) Аннотация

B) Введение

C) Методология

D) Результаты

E) Обсуждение

18. Какой метод НЕ используется для анализа данных?
A) Кластерный анализ

B) Регрессионный анализ

C) Факторный анализ

D) SWOT-анализ

E) Корреляционный анализ

19. Какой инструмент используется для воспроизводимости исследований?
A) Docker

B) Zotero

C) LaTeX

D) GitHub

E) Jupyter

20. Какой метод применяется для оптимизации гиперпараметров моделей ML?
A) GridSearch

B) Фаззинг

C) Регрессионное тестирование

D) Нагрузочное тестирование

E) Модульное тестирование

21. Какой раздел статьи содержит краткое изложение исследования?
A) Аннотация

B) Введение

C) Методология

D) Результаты

E) Обсуждение

22. Какой метод НЕ используется для анализа производительности ПО?
A) Профилирование

B) Нагрузочное тестирование

C) Регрессионное тестирование

D) Анализ временной сложности

E) Бенчмаркинг

23. Какой инструмент используется для написания научных статей?
A) LaTeX

B) Docker

C) GitHub

D) Jupyter

E) Prometheus

24. Какой метод применяется для анализа социальных сетей?
A) Сетевой анализ

B) Регрессионный анализ

C) Факторный анализ

D) SWOT-анализ

E) Корреляционный анализ

25. Какой раздел статьи содержит предложения для будущих исследований?
A) Аннотация

B) Введение

C) Методология

D) Результаты

E) Заключение

Методические материалы, характеризующие процедуры оценивания

На основе типовых заданий программным комплексом информационно-образовательного портала МИ ВлГУ формируются в автоматическом режиме тестовые задания для студентов: три вопроса из блока 1, три вопроса из блока 2 и задача из блока 3. Программный комплекс формирует индивидуальные задания для каждого зарегистрированного в системе студента и устанавливает время прохождения тестирования. Результатом тестирования является процент правильных ответов, с учетом индивидуального семестрового рейтинга студента формируется экзаменационная оценка.

Максимальная сумма баллов, набираемая студентом по дисциплине равна 100.

Оценка в баллах	Оценка по шкале	Обоснование	***Уровень сформированности компетенций***
Более 80	«Отлично»	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	***Высокий уровень***
66-80	«Хорошо»	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	***Продвинутый уровень***

		минимальным числом баллов, некоторые виды заданий выполнены с ошибками	
50-65	«Удовлетворительно»	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	***Пороговый уровень***
Менее 50	«Неудовлетворительно»	Содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки	***Компетенции не сформированы***

3. Задания в тестовой форме по дисциплине

Примеры заданий:

1. Какой метод НЕ используется для оценки сложности алгоритмов?
A) O-нотация (Big-O)
B) Анализ временных затрат
C) Кластерный анализ
D) Теоретико-графовые методы
E) Профилирование кода

2. Какой инструмент НЕ применяется для управления зависимостями в Python?
A) pip
B) conda
C) npm
D) poetry
E) virtualenv

3. Какой принцип ООП НЕ является базовым?
A) Инкапсуляция
B) Полиморфизм
C) Наследование
D) Абстракция
E) Рефакторинг

4. Какой протокол НЕ используется в веб-разработке?
A) HTTP
B) WebSocket
C) MQTT
D) FTP
E) SMTP

5. Какой тип тестирования НЕ относится к автоматизированному?
A) Модульное
B) Интеграционное
C) Регрессионное

D) Юзабилити-тестирование
E) Нагрузочное

6. Какой язык НЕ является строго типизированным?
A) Java
B) C++
C) Python
D) Haskell
E) Rust

7. Какой инструмент НЕ используется для контроля версий?
A) Git
B) Mercurial
C) Subversion
D) Jira
E) Bitbucket

8. Какой подход НЕ относится к методологиям разработки ПО?
A) Agile
B) Waterfall
C) DevOps
D) Scrum
E) Singleton

9. Какой термин НЕ связан с базами данных?
A) ORM
B) NoSQL
C) ACID
D) REST
E) Индекс

10. Какой алгоритм НЕ используется для сортировки данных?
A) Быстрая сортировка (QuickSort)
B) Сортировка пузырьком (BubbleSort)
C) Поиск в ширину (BFS)
D) Сортировка слиянием (MergeSort)
E) Пирамидальная сортировка (HeapSort)

Полный перечень тестовых заданий с указанием правильных ответов, размещен в банке вопросов на информационно-образовательном портале института по ссылке https://www.mivlgu.ru/iop/course/view.php?id=3933

Оценка рассчитывается как процент правильно выполненных тестовых заданий из их общего числа.