

Подписано электронной подписью:
Вержицкий Данил Григорьевич
Должность: Директор КГПИ КемГУ
Дата и время: 2025-04-23 00:00:00
471086fad29a3b30e244c728abc3661ab35c9d50210dcf0e75e03a5b6fdf6436

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Кузбасский гуманитарно-педагогический институт
федерального государственного бюджетного образовательного учреждения
высшего образования
«Кемеровский государственный университет»
Факультет информатики, математики и экономики

УТВЕРЖДАЮ

Декан

_____ А. В. Фомина
«30» января 2025 г.

Рабочая программа дисциплины

К.М.05.03 Технологии разработки программного обеспечения для научных исследований

Направление подготовки

01.04.02 Прикладная математика и информатика

Направленность (профиль) подготовки

Математическое моделирование

Программа
магистратуры

Квалификация выпускника
магистр

Форма обучения
очная

Год набора 2025

Новокузнецк 2025

Оглавление

1 Цель дисциплины.	3
Формируемые компетенции, индикаторы достижения компетенций, знания, умения, навыки	3
Место дисциплины	3
2 Объём и трудоёмкость дисциплины по видам учебных занятий. Формы промежуточной аттестации.	3
3. Учебно-тематический план и содержание дисциплины.	4
3.1 Учебно-тематический план	4
4 Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации.	5
5 Материально-техническое, программное и учебно-методическое обеспечение дисциплины.	5
5.1 Учебная литература	5
5.2 Материально-техническое и программное обеспечение дисциплины.	6
5.3 Современные профессиональные базы данных и информационные справочные системы.	6
6 Иные сведения и (или) материалы.	7
6.1.Примерные темы письменных работ	7
6.2. Примерные вопросы и задания / задачи для промежуточной аттестации .	7
6.3. Использование интерактивной формы обучения.	10

1 Цель дисциплины.

В результате освоения данной дисциплины у обучающегося должны быть сформированы компетенции основной профессиональной образовательной программы магистратуры (далее - ОПОП):

ПК-1. Способен проводить научно-исследовательские и опытно-конструкторские разработки при исследовании самостоятельных тем

Формируемые компетенции, индикаторы достижения компетенций, знания, умения, навыки

Таблица 1 – Индикаторы достижения компетенций, формируемые дисциплиной

Код и название компетенции	Индикаторы достижения компетенции по ОПОП	Знания, умения, навыки (ЗУВ), формируемые дисциплиной
ПК-1 Способен проводить научно-исследовательские и опытно-конструкторские разработки при исследовании самостоятельных тем	ПК 1.1. Осуществляет разработку планов и методических программ проведения исследований, организует проведение исследования. ПК 1.3 Оценивает качество формализации и алгоритмизации поставленных задач ПК 1.4. Оценивает качество и эффективности программного кода. Принимает решения по его изменению. ПК 1.7 Проводит разработку и оценку качества технической документации	Знать: – Основы разработки и проектирования программного кода для научных исследований; – Техники проверки работоспособности программного обеспечения для научных исследований; – Техники и способы интеграции программных модулей и компонентов программного обеспечения для научных исследований; – Основы руководства разработкой технической документации. Уметь: – решать прикладные задачи проверки работоспособности программного обеспечения; – применять методы интеграции модулей и компонентов программного обеспечения; – составлять техническую документацию.. Владеть навыками: – организации процесса разработки программного кода; – тестирования и проверки работоспособности программного обеспечения; – интеграции модулей и компонентов программного обеспечения; составления технической документации.

Место дисциплины

Дисциплина включена в модуль «Научно-исследовательская работа в области математического моделирования» ОПОП ВО, обязательная часть. Дисциплина осваивается на 2 курсе в 3 семестре.

2 Объём и трудоёмкость дисциплины по видам учебных занятий.

Формы промежуточной аттестации.

Таблица 2 – Объем и трудоемкость дисциплины по видам учебных занятий

Общая трудоемкость и виды учебной работы по дисциплине, проводимые в разных формах	Объём часов по формам обучения		
	ОФО	ОЗФО	ЗФО
1 Общая трудоемкость дисциплины	108		
2 Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)	32		

Аудиторная работа (всего):	32		
в том числе:			
лекции	16		
практические занятия, семинары			
практикумы			
лабораторные работы	16		
в интерактивной форме			
в электронной форме			
Внеаудиторная работа (всего):			
в том числе, индивидуальная работа обучающихся с преподавателем			
подготовка курсовой работы /контактная работа			
групповая, индивидуальная консультация и иные виды учебной деятельности, предусматривающие групповую или индивидуальную работу обучающихся с преподавателем)			
творческая работа (эссе)			
3 Самостоятельная работа обучающихся (всего)	76		
4 Промежуточная аттестация обучающегося и объём часов, выделенный на промежуточную аттестацию: – зачет	-		

3. Учебно-тематический план и содержание дисциплины.

3.1 Учебно-тематический план

Таблица 3 - Учебно-тематический план очной формы обучения

№ недели п/п	Разделы и темы дисциплины по занятиям	Общая трудоемко сть (всего час.)	Трудоемкость занятий (час.)			Формы текущего контроля и промежуточной аттестации успеваемости
			Аудиторн. занятия		СРС	
			лекц.	лаб.		
1-2	1. Введение. Модели жизненного цикла программного обеспечения. Зрелость процессов разработки ПО. ИТ-решения по управлению жизненным циклом ПО.	16	2	-	14	Устный опрос
3-4	2. Методология проектирования программных продуктов для научных исследований	20	2	4	14	Лабораторное задание№1
5-8	3. Технологии разработки	24	4	4	16	
5-6	3.1 Гибкая методология разработки программного обеспечения Принципы гибкой разработки. Значение гибкой разработки .	12	2	2	8	
7-8	3.2 Функционально-объектная декомпозиция	12	2	2	8	
9-12	4. Средства информационной поддержки	24	4	4	16	Лабораторное задание№2
9-10	4.1 MicrosoftVisualStudioTeamFoundationServer(TFS). Развертывание Team Foundation Server.	12	2	2	8	
11-12	4.2 Шаблоны командных проектов. Среда визуального программирования "Алгозит"	12	2	2	8	

№ недели п/п	Разделы и темы дисциплины по занятиям	Общая трудоёмко сть (всего час.)	Трудоемкость занятий (час.)			Формы текущего контроля и промежуточной аттестации успеваемости
			Аудиторн. занятия		СРС	
			лекц.	лаб.		
13-16	5. Внедрение и сопровождение программных продуктов	24	4	4	16	
13-14	5.1 Управление жизненным циклом приложений. Архитектурное проектирование.	12	2	2	8	
15-16	5.2 Разработка приложения в среде "Алгозит". Тестирование приложения	12	2	2	8	
	Промежуточная аттестация	-				Зачет
	Всего 3 семестр:	108	16	16	76	-

4 Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации.

Для положительной оценки по результатам освоения дисциплины обучающемуся необходимо выполнить все установленные виды учебной работы. Оценка результатов работы обучающегося в баллах (по видам) приведена в таблице 7.

Таблица 7 - Шкала и показатели оценивания результатов учебной работы обучающихся по видам в балльно-рейтинговой системе (БРС)

Учебная работа (виды)	Сумма баллов	Виды и результаты учебной работы	Оценка в аттестации (шкала и показатели оценивания)	Баллы
Текущая учебная работа в семестре (выполнение заданий)	80	Лекционные занятия (конспект) (8 занятий)	4 балла посещение 1 занятия	8 –16
		Лабораторные работы (2 работа)	За одну лабораторную работу 16 баллов (пороговое значение) 24 баллов 32 баллов (максимальное значение)	32–64
Итого по текущей работе в семестре				40 –80
Промежуточная аттестация (зачет)	20	Теоретический вопрос №1	5 баллов (пороговое значение) 10 баллов (максимальное значение)	5 - 10
		Практическое задание	5 баллов (пороговое значение) 10 баллов (максимальное значение)	5 - 10
Итого по промежуточной аттестации (зачету)				10-20 б.
Суммарная оценка по дисциплине: Сумма баллов текущей и промежуточной аттестации				51 – 100 б.

5 Материально-техническое, программное и учебно-методическое обеспечение дисциплины.

5.1 Учебная литература

Основная учебная литература

1. Кон, М. Agile. Оценка и планирование проектов: Практическое руководство / Кон М. - М.: Альпина Паблишер, 2018. - 418 с.: ISBN 978-5-9614-6947-9. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1003486> . – Режим доступа: по подписке.

Дополнительная учебная литература

1. Чукич, И. Функциональное программирование на C++ : учебное пособие / И. Чукич

; перевод с английского В. Ю. Винника, А. Н. Киселева. — Москва : ДМК Пресс, 2020. — 360 с. — ISBN 978-5-97060-781-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/140597>. — Режим доступа: для авториз. пользователей.

2. Пай, П. Реактивное программирование на C++ / П. Пай, П. Абрахам ; перевод с английского В. Ю. Винника. — Москва : ДМК Пресс, 2019. — 324 с. — ISBN 978-5-97060-778-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/131698>. — Режим доступа: для авториз. пользователей.

3. Гвоздева, Т. В. Проектирование информационных систем. Стандартизация : учебное пособие / Т. В. Гвоздева, Б. А. Баллод. — Санкт-Петербург : Лань, 2019. — 252 с. — ISBN 978-5-8114-3517-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/reader/book/115515/#1>. — Режим доступа: для авториз. пользователей.

4. Резова, Н. Л. Технология программирования : учебное пособие / Н. Л. Резова, Г. Ш. Шкаберина. — Красноярск : СибГУ им. академика М. Ф. Решетнёва, 2019. — 94 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/147448>. — Режим доступа: для авториз. пользователей.

5.2 Материально-техническое и программное обеспечение дисциплины.

Учебные занятия по дисциплине проводятся в учебных аудиториях КГПИ КемГУ:

410 Учебная аудитория (мультимедийная) для проведения: - занятий лекционного типа; - групповых и индивидуальных консультаций; - текущего контроля и промежуточной аттестации; Специализированная (учебная) мебель: доска меловая, кафедра, моноблоки аудиторные. Оборудование: стационарное - компьютер, экран, проектор. Используемое программное обеспечение: MSWindows, LibreOffice (свободно распространяемое ПО), Яндекс.Браузер (отечественное свободно распространяемое ПО). Интернет с обеспечением доступа в ЭИОС.	654079, Кемеровская область, г. Новокузнецк, пр-кт Metallurgov, д. 19
508 Лаборатория компьютерного моделирования Учебная аудитория (мультимедийная) для проведения: - занятий лабораторного типа; - групповых и индивидуальных консультаций; - самостоятельной работы; - текущего контроля и промежуточной аттестации. Специализированная (учебная) мебель: доска меловая, кафедра, столы, стулья. Оборудование для презентации учебного материала: стационарное - компьютер преподавателя, проектор, экран. Лабораторное оборудование: стационарное – компьютеры для обучающихся (18 шт.). Используемое программное обеспечение: MSWindows, LibreOffice (свободно распространяемое ПО), FoxitReader (свободно распространяемое ПО), Firefox 14 (свободно распространяемое ПО), Яндекс.Браузер (отечественное свободно распространяемое ПО), Opera 12 (свободно распространяемое ПО), Microsoft Visual Studio, Интерпретатор "Ядро" (отечественное ПО лицензионный договор №1 от 16.06.2020 г. до 16.06.2025 г.); Среда функционально-объектного программирования "Алгозит" (Отечественное ПО лицензионный договор №2 от 16.06.2020 г. до 16.06.2025 г.). Интернет с обеспечением доступа в ЭИОС.	654079, Кемеровская область, г. Новокузнецк, пр-кт Metallurgov, д. 19

5.3 Современные профессиональные базы данных и информационные справочные системы.

Перечень СПБД и ИСС по дисциплине

Общероссийский математический портал (информационная система) - <http://www.mathnet.ru/>

Научная электронная библиотека eLIBRARY.RU – крупнейший российский информационный портал в области науки, технологии, медицины и образования, содержащий рефераты и полные тексты - www.elibrary.ru

Science Direct содержит более 1500 журналов издательства Elsevier, среди них издания по математике и информатике. <https://www.sciencedirect.com>

Информационная система «Единое окно доступа к информационным ресурсам» - <http://window.edu.ru/catalog/>

Базы данных и аналитические публикации на портале «Университетская информационная система Россия» - <https://uisrussia.msu.ru/>

Крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки - <https://github.com/>

Новые информационные технологии и программы - Сайт о свободном программном обеспечении и новых информационных технологиях - <http://pro-spo.ru/>

CITForum.ru - on-line библиотека свободно доступных материалов по информационным технологиям на русском языке - <http://citforum.ru>

6 Иные сведения и (или) материалы.

6.1.Примерные темы письменных работ

6.1.1. Отчет по лабораторному заданию №1 «Работа с карточками»

В малых группах разыграть итерацию процесса разработки мнимого или реального проекта с использованием «доски» и «карточек».

6.1.2. Отчет по лабораторному заданию №1«Средства информационной поддержки»

Разработать вычислительную функционально-объектную программу в среде визуального программирования «Алгозит» согласно варианту:

- 1) факториала числа,
- 2) корня квадратного,
- 3) синуса,
- 4) косинуса,
- 5) тангенса,
- 6) котангенса,
- 7) логарифма,
- 8) экспоненты.

Работу следует выполнять в малых группах, следуя принципам «экстремального программирования»

6.2. Примерные вопросы и задания / задачи для промежуточной аттестации

Примерные теоретические вопросы и практические задания к экзамену

Раздел 1. Введение

Примерные теоретические вопросы

1. Критерии качества программного средства. Многоуровневая модель качества ПО. Оценочные характеристики качества программного продукта
2. Жизненный цикл программного продукта, фазы жизненного цикла. Этапы классического жизненного цикла, их содержание.
3. Фаза разработки, этапы процесса разработки. Стратегии конструирования ПО: линейная, инкрементная, эволюционная.
4. Стандарт ISO/IEC 12207-95: основные определения – система, модель жизненного

цикла, квалификационные требования. Основные процессы, их содержание, работы и задачи процесса разработки.

5. Стандарт ISO/IEC 15504 (SPICE): оценка возможностей разработчика. Связь этого стандарта с моделью зрелости предприятия SEI CMM.

6. Прогностические модели процесса разработки: каскадная, RAD, спиральная.

7. Адаптивные модели процесса разработки: экстремальное программирование, Scrum.

Примерные практические задания

1. Опишите любым способом алгоритм юнит-теста для методов сохранения, обновления, удаления объекта доступа к данным.

Раздел 2. Методология проектирования программных продуктов для научных исследований

Примерные теоретические вопросы

8. Руководство программным проектом. Предварительные оценки проекта. Системный анализ и анализ требований. Анализ рисков. Планирование процесса разработки. Типовая структура распределения работ.

9. Контроль процесса разработки. Размерно- и функционально-ориентированные метрики. Метрические характеристики объектно-ориентированных систем.

10. Структурный и объектно-ориентированный подходы к разработке ПО. Их сравнительный анализ. Сущность объектного подхода к разработке программных средств.

11. Анализ предметной области: цели и задачи. Модели предметной области. Формальные определения. Классификация моделей. Методология IDEF0, синтаксис IDEF0-моделей.

12. Диаграммы потоков данных (DFD-диаграммы) и диаграммы потоков работ (IDEF3-диаграммы), их использование при моделировании предметной области.

Примерные практические задания

1. Сформулируйте цель (обобщенно) и задачи (подробно) разработки веб-сайта, позволяющего вычислять производные произвольных выражений. Сортируйте задачи по приоритетам.

2. Проведите объектную декомпозицию сущностей ядра веб-сайта, позволяющего вычислять производные произвольных выражений.

Раздел 3. Технологии разработки

Примерные теоретические вопросы

13. Объектно-ориентированный анализ предметной области. Методика определения границ системы и ключевых абстракций. Пример проведения анализа. Функциональные и не-функциональные требования к системе.

14. Функциональные требования к системе. Способ их представления в виде UML-диаграммы. Пример диаграммы с использованием отношений «расширяет» и «включает». Понятие прецедента и сценария.

15. Концептуальная модель системы: концептуальные классы, системные события и системные операции. Способ их представления в виде UML-диаграмм. Пример концептуального описания прецедента.

16. Диаграммы взаимодействия как элементы концептуальной модели. Синтаксис диаграмм взаимодействия.

Примерные практические задания

1. Постройте UML диаграмму классов, решающих задачу интегрирования функции двух переменных на прямоугольной области.

2. Постройте функционально-объектную схему, решающую задачу интегрирования функции двух переменных на прямоугольной области.

Раздел 4. Средства информационной поддержки

Примерные теоретические вопросы

17. Проектирование программных средств. Цели и задачи этапа проектирования. Поня-

тие модели проектирования, ее отличия от концептуальной модели. Стадии проектирования, их краткая характеристика.

18. Задачи, решаемые на стадии эскизного проектирования. Понятие архитектуры ПС. Проблема выбора архитектуры. Влияние архитектуры на качественные характеристики ПС.

19. Понятие модуля и модульного программирования. Преимущества модульного подхода к разработке ПО. Модули как средство физического структурирования ПО. Свойства модулей.

20. Задачи, решаемые на стадии детального проектирования. Цели и задачи проектирования пользовательского интерфейса.

21. Понятие шаблона. Классификация шаблонов. Стандарт описания шаблонов.

22. Идентификация методов программных классов. Диаграммы классов, способы отображения отношений ассоциации и зависимости. Пример диаграммы классов.

Раздел 5. Внедрение и сопровождение программных продуктов

23. Тестирование и отладка программного средства. Стадии тестирования и их характеристика. Основные принципы тестирования. Тесты и тестовые наборы. Понятие тестового покрытия.

24. Отладочное тестирование. Соотношение структурного и функционального подходов. Примеры реализации.

25. Интеграционное тестирование. Виды интеграционного тестирования. Критерии полноты тестовых наборов. Регрессионное тестирование. Критерии завершения отладочного тестирования.

26. Системное тестирование. Виды системного тестирования. Критерии полноты тестовых наборов.

27. Особенности объектно-ориентированного тестирования. Расширение области применения тестирования. Критерии тестирования моделей. Тестирование классов. Тестирование кластеров и потоковое тестирование.

28. Понятие автоматизированного тестирования. Достоинства и недостатки автоматизированного тестирования. Средства автоматизированного тестирования.

29. Утилита модульного тестирования NUnit. Средства описания тестов. Утверждения, параметры утверждений.

30. Понятие версии программного продукта и системы контроля версий. Модели версионирования, их сравнение.

31. Понятие сборки, манифест сборки. Сборка приложения, системы автоматизации сборки.

32. Утилита NAnt, файл сборки и его структура. Цели, зависимость целей, описание целей.

33. Документирование процесса разработки. Типы документов управления.

34. Документирование программного продукта. Документация сопровождения, ее назначение и состав. Пользовательская документация, ее назначение и состав.

Примерные практические задания

1. Используя шаблон MVC опишите архитектуру простейшей игры (например Змейка).

Раздел 5. Внедрение и сопровождение программных продуктов

23. Тестирование и отладка программного средства. Стадии тестирования и их характеристика. Основные принципы тестирования. Тесты и тестовые наборы. Понятие тестового покрытия.

24. Отладочное тестирование. Соотношение структурного и функционального подходов. Примеры реализации.

25. Интеграционное тестирование. Виды интеграционного тестирования. Критерии полноты тестовых наборов. Регрессионное тестирование. Критерии завершения отладочного тестирования.

26. Системное тестирование. Виды системного тестирования. Критерии полноты тестовых наборов.

27. Особенности объектно-ориентированного тестирования. Расширение области применения тестирования. Критерии тестирования моделей. Тестирование классов. Тестирование кластеров и потоковое тестирование.

28. Понятие автоматизированного тестирования. Достоинства и недостатки автоматизированного тестирования. Средства автоматизированного тестирования.

29. Утилита модульного тестирования NUnit. Средства описания тестов. Утверждения, параметры утверждений.

30. Понятие версии программного продукта и системы контроля версий. Модели версионирования, их сравнение.

31. Понятие сборки, манифест сборки. Сборка приложения, системы автоматизации сборки.

32. Утилита NAnt, файл сборки и его структура. Цели, зависимость целей, описание целей.

33. Документирование процесса разработки. Типы документов управления.

34. Документирование программного продукта. Документация сопровождения, ее назначение и состав. Пользовательская документация, ее назначение и состав.

Примерные практические задания

1. Опишите алгоритмы юнит-тестов для класса модели игры Тетрисы.

2. Опишите алгоритмы юнит-тестов для класса модели игры Змейка.

6.3. Использование интерактивной формы обучения.

Все лабораторные занятия по дисциплине проводятся с использованием активных и интерактивных форм обучения (выполнение заданий в малых группах, разработка прикладных ПС для решения конкретной практической задачи и т.д.).

Основной целью интерактивной методики является привитие навыков самостоятельного поиска ответов и обучения через взаимодействие. При этом основной упор сделан на взаимодействие обучаемых друг с другом. Через интерактивное взаимодействие студенты получают практические навыки установления профессиональных контактов в соответствии с потребностями совместной деятельности, включая обмен информацией и выработку единой стратегии взаимодействия.

Составитель (и): Ульянов А.Д., доцент

(фамилия, инициалы и должность преподавателя (ей))