

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Нижегородский государственный университет
им. Н.И. Лобачевского»**

Арзамасский филиал ННГУ - Факультет естественных и математических наук

УТВЕРЖДЕНО

решением президиума Ученого совета ННГУ

протокол № 1 от 16.01.2024 г.

Рабочая программа дисциплины

Инструментальные средства разработки программного обеспечения

Уровень высшего образования

Бакалавриат

Направление подготовки / специальность

09.03.03 - Прикладная информатика

Направленность образовательной программы

Системное и прикладное программирование

Форма обучения

очная, очно-заочная

г. Арзамас

2024 год начала подготовки

1. Место дисциплины в структуре ОПОП

Дисциплина Б1.В.20 Инструментальные средства разработки программного обеспечения относится к части, формируемой участниками образовательных отношений образовательной программы.

2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства	
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	Для текущего контроля успеваемости	Для промежуточной аттестации
ПК-10: Способен осуществлять локальную модернизацию системы, адаптировать бизнес- процессы организации к возможностям ИС (ИИС)	<p>ПК-10.1: Демонстрирует знание методологических основ документирования бизнес- процессов.</p> <p>ПК-10.2: Демонстрирует умение организовать и поддерживать репозиторий ИС, хранящий информацию о сопровождении системы в процессе ее жизненного цикла.</p> <p>ПК-10.3: Имеет практический опыт документирования бизнес- процессов и адаптации их к возможностям конкретной ИС.</p>	<p>ПК-10.1:</p> <p>знать основные подходы к разработке программного обеспечения, основные парадигмы программирования, их достоинства и недостатки методы проектирования программных систем, понятие класса и объекта, основные способы взаимодействия классов.</p> <p>уметь использовать языковые конструкции для разработки классов и объектов, разграничения доступа к данным проводить анализ предметной области с целью выделения ее основных понятий, свойств и характеристик классов владеть базовыми навыками разработки, методами анализа предметной области, навыками проведения анализа взаимосвязей между ее основными компонентами</p> <p>ПК-10.2:</p> <p>Знать формальные методы и модели описания структуры информационных систем;</p> <p>Уметь проводить системный анализ прикладной области с целью оптимального выбора архитектуры системы</p> <p>Владеть навыками</p>	<p>Задания</p> <p>Практическое задание</p> <p>Реферат</p> <p>Тест</p>	<p>Зачёт:</p> <p>Контрольные вопросы</p>

		<p>организации и поддержки репозитория ИС</p> <p>ПК-10.3:</p> <p>Знать основы документирования бизнес-процессов и адаптацию их к возможностям конкретной ИС</p> <p>Уметь проводить документирование бизнес-процессов и адаптацию их к возможностям конкретной ИС</p> <p>Владеть навыками документирования бизнес-процессов и адаптацию их к возможностям конкретной ИС</p>		
<p>ПК-8: Способен разрабатывать лингвистическое, информационное и программное обеспечение ИС (ИИС) и сопровождающую его документацию</p>	<p>ПК-8.1: Демонстрирует знание современных языков и систем программирования, формализмов описания знаний на концептуальном и инфологическом уровнях, требований к технической документации на все виды обеспечения ИС (ИИС).</p> <p>ПК-8.2: Применяет современные языки и системы программирования, формализмы описания знаний на концептуальном и инфологическом уровнях при разработке лингвистического, информационного и программного обеспечения ИИС и сопровождающей его документации.</p> <p>ПК-8.3: Имеет практический опыт разработки лингвистического, информационного и программного обеспечения конкретной ИС (ИИС) и сопровождающей ее документации.</p>	<p>ПК-8.1:</p> <p>Знать модели процесса разработки программного обеспечения; основные принципы процесса разработки программного обеспечения;</p> <p>Уметь работать с современными системами программирования, включая объектно-ориентированные.</p> <p>Владеть основными методологиями процессов разработки программного обеспечения,</p> <p>ПК-8.2:</p> <p>Знать основные подходы к интегрированию программных модулей; основные методы и средства эффект явной разработки; основы верификации и аттестации программного обеспечения; концепции и реализации программных процессов;</p> <p>Уметь пользоваться глобальными компьютерными сетями, каталогами ресурсов, поисковые системы</p> <p>Владеть методами для получения кода с заданной функциональностью и</p>	<p>Задания</p> <p>Практическое задание</p> <p>Реферат</p> <p>Тест</p>	<p>Зачёт:</p> <p>Контрольные вопросы</p>

		<p>степенью качества</p> <p>ПК-8.3:</p> <p>Знать особенности осуществления разработки лингвистического, информационного и программного обеспечения конкретной ИС</p> <p>Уметь разрабатывать программное обеспечение ИС и сопровождающую его документацию</p> <p>Владеть способностью осуществлять разработку лингвистического, информационного и программного обеспечения конкретной ИС (ИИС) и сопровождающей его документации.</p>		
--	--	--	--	--

3. Структура и содержание дисциплины

3.1 Трудоемкость дисциплины

	очная	очно-заочная
Общая трудоемкость, з.е.	3	3
Часов по учебному плану	108	108
в том числе		
аудиторные занятия (контактная работа):		
- занятия лекционного типа	14	8
- занятия семинарского типа (практические занятия / лабораторные работы)	28	18
- КСР	1	1
самостоятельная работа	65	81
Промежуточная аттестация	0 Зачёт	0 Зачёт

3.2. Содержание дисциплины

(структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий)

Наименование разделов и тем дисциплины	Всего (часы)	в том числе			Самостоятельная работа обучающегося, часы
		Контактная работа (работа во взаимодействии с преподавателем), часы из них			
		Занятия лекционного	Занятия семинарского	Всего	

			типа		типа (практические занятия/лаборат орные работы), часы					
	о ф о	о з ф о	о ф о	о з ф о	о ф о	о з ф о	о ф о	о з ф о	о ф о	о з ф о
Тема 1. Инструментальные средства управления проектом	11	17	2	2	4	4	6	6	5	11
Тема 2. Инструментальные средства проектирования предметной области	16	17	2	2	4	4	6	6	10	11
Тема 3. Инструментальные средства проектирования и анализа требований к программному обеспечению	16	17	2	2	4	4	6	6	10	11
Тема 4. Средства разработки программного обеспечения	16	18	2	2	4	4	6	6	10	12
Тема 5. Визуальные средства разработки приложений	16	14	2	0	4	2	6	2	10	12
Тема 6. Инструментальные средства тестирования приложений	16	12	2	0	4	0	6	0	10	12
Тема 7. Инструментальные средства документирования приложений	16	12	2	0	4	0	6	0	10	12
Аттестация	0	0								
КСР	1	1						1	1	
Итого	108	108	14	8	28	18	43	27	65	81

Содержание разделов и тем дисциплины

Тема 1. Инструментальные средства управления проектом

Visual Studio: Обзор и основные функции.

Visual Studio Code: Преимущества и возможности.

JetBrains Rider: Особенности и преимущества.

Resharper: Возможности для оптимизации кода.

MSBuild: Процесс сборки проектов.

NDepend: Анализ и оптимизация кода.

StyleCop: Проверка стиля кода.

Git: Система контроля версий для разработчиков.

Team Foundation Server: Управление версиями для команд.

Тема 2. Инструментальные средства проектирования предметной области

Инструменты для создания диаграмм классов: UML Class Diagrams, Microsoft Visio.

Инструменты для моделирования баз данных: Microsoft Visio, ERwin Data Modeler.

Инструменты для бизнес-моделирования: Microsoft Visio, Enterprise Architect.

Инструменты для тестирования программного обеспечения: TestLink, Quality Center.

Инструменты для документирования: Microsoft Word, Adobe Robotic.

Инструменты для контроля версий: Git, Subversion.

Инструменты для автоматизации сборки: MSBuild, Jenkins.

Инструменты для статического анализа кода: ReSharper, SonarQube.

Тема 3. Инструментальные средства проектирования и анализа требований к программному обеспечению

UML (Unified Modeling Language) - стандарт для создания диаграмм в проектах разработки программного обеспечения.

Enterprise Architect - инструмент для моделирования бизнес-процессов и анализа требований.

IBM Rational Rose - средство для визуального моделирования и анализа требований.

Borland Together - система управления проектами и требованиями.

DOORS - система управления требованиями и конфигурациями.

SpecFlow - инструмент для создания сценариев тестирования и анализа требований на основе языка программирования C#.

FitNesse - инструмент для автоматизации тестирования и анализа функциональных требований.

jqAssistant - инструмент для анализа и визуализации требований на основе JSON.

FuseTest - средство для автоматизированного тестирования и анализа нефункциональных требований.

ReQtest - инструмент для управления требованиями и тестирования на основе гибких методологий разработки.

Тема 4. Средства разработки программного обеспечения

Microsoft Visual Studio - интегрированная среда разработки (IDE) для создания приложений на различных языках программирования.

Microsoft Visual Studio Code - редактор кода с поддержкой различных языков программирования и возможностью интеграции с системами контроля версий.

JetBrains ReSharper - плагин для Visual Studio, который улучшает качество кода, автоматизирует рефакторинг и предлагает инструменты для навигации и поиска ошибок.

JetBrains Rider - кросс-платформенная IDE для разработки приложений на .NET Framework и .NET Core.

Eclipse - свободная интегрированная среда для разработки на различных языках, включая C#.

MonoDevelop - свободная IDE с открытым исходным кодом для разработки приложений на разных платформах.

SharpDevelop - еще одна свободная IDE для создания приложений с использованием C#.

Microsoft Expression Blend - программа для создания пользовательских интерфейсов и анимаций.

Тема 5. Визуальные средства разработки приложений

Xamarin - платформа для разработки кросс-платформенных приложений с использованием .NET Framework.

Unity - игровой движок для создания двух- и трехмерных игр.

Unreal Engine - еще один игровой движок, используемый для создания игр и интерактивных приложений.

Expression Blend - инструмент для разработки пользовательских интерфейсов.

Qt - кросс-платформенный инструмент для разработки графических и мультимедийных приложений.

SharpVectors - библиотека векторной графики.

AForge.NET - пакет для обработки изображений, распознавания образов, обработки звука и других задач.

Тема 6. Инструментальные средства тестирования приложений

JUnit - фреймворк для модульного тестирования.

NUnit - альтернатива JUnit.

MSTest - встроенный в .NET Framework фреймворк.

TestNG - фреймворк, позволяющий писать тестовые сценарии.

C# Unit Testing - примеры написания юнит-тестов на языке C#.

FitNesse - фреймворк функционального тестирования.

Specflow - фреймворк для написания сценариев с использованием Behavior Driven Development (BDD).

MsTest - средство тестирования, встроенное в Visual Studio.

Selenium - фреймворк для функционального и интеграционного тестирования веб-приложений.

Allure - инструмент для визуализации результатов тестирования.

Тема 7. Инструментальные средства документирования приложений

Sandcastle - инструмент для генерации документации по API.

Doxygen - генератор документации для языков программирования C++, Java и ряда других.

NDoc - инструмент для документирования кода на языке Nemerle.

SandCastle Help File Builder - инструмент для создания файлов справки.

Swagger - платформа для описания и документирования API.

Apideck - онлайн-сервис для документирования API.

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа обучающихся включает в себя подготовку к контрольным вопросам и заданиям для текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведенным в п. 5.

Для обеспечения самостоятельной работы обучающихся используются:

- электронный курс "Инструментальные средства разработки программного обеспечения" (Не нашел).

Иные учебно-методические материалы: Учебно-методические документы, регламентирующие самостоятельную работу

адреса доступа к документам:

<https://arz.unn.ru/sveden/document/>

https://arz.unn.ru/pdf/Metod_all_all.pdf

5. Фонд оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю)

5.1 Типовые задания, необходимые для оценки результатов обучения при проведении текущего контроля успеваемости с указанием критериев их оценивания:

5.1.1 Типовые задания (оценочное средство - Задания) для оценки сформированности компетенции ПК-10:

1. Специальное программное обеспечение
2. Инструментальная система технологии программирования
3. Четыре класса компьютерной поддержки инструментальных систем технологий программирования
4. Комплексность
5. Ориентированность на коллективную разработку
6. Технологическая определенность
7. Интегрированность
8. Компоненты инструментальных систем технологий программирования
9. База данных разработки
10. Инструментарий
11. Интерфейсы
12. Общая архитектура инструментальных систем технологий программирования
13. Инструментальная система поддержки проекта
14. Языково-зависимая инструментальная система
15. Пользовательский интерфейс
16. Схема организации взаимодействия компьютера и пользователя
17. Процедурно-ориентированный подход к разработке интерфейсов
18. Объектно-ориентированный подход к разработке интерфейсов
19. Типы интерфейсов
20. Интерфейс-меню
21. Интерфейсы со свободной навигацией
22. Критерии оценки интерфейса пользователем

5.1.2 Типовые задания (оценочное средство - Задания) для оценки сформированности компетенции ПК-8:

1. Необходимые инструментальные средства разработки программ
2. Часто используемые инструментальные средства разработки программ
3. Специализированные инструментальные средства разработки программ
4. Интегрированные среды разработки
5. Средства разработки программного обеспечения
6. Определение «разработка программ»
7. Три этапа разработки программ
8. Средства проектирования приложений
9. Средства реализации программного кода
10. Средства тестирования программ
11. Классы инструментальных средств разработки программ
12. Четыре категории инструментальных программ, применяемые при проектировании экспертных систем
13. Оболочки экспертных систем
14. Языки программирования высокого уровня
15. Среда программирования, поддерживающая несколько парадигм
16. Дополнительные модули
17. Языки инженерии знаний
18. Язык логического программирования Prolog
19. Язык функционального программирования Lisp
20. Средства автоматизации разработки экспертных систем
21. Общее программное обеспечение

Критерии оценивания (оценочное средство - Задания)

Оценка	Критерии оценивания
отлично	Ответ полный и правильный на основании изученной теории; материал изложен в необходимой логической последовательности, грамотный научный язык; ответ самостоятельный
хорошо	Ответ полный и правильный на основании изученной теории; материал изложен в необходимой логической последовательности при этом допущены две-три несущественные ошибки, исправленные по требованию преподавателя
удовлетворительно	Ответ полный, но при этом допущена существенная ошибка или неполный, несвязный ответ
неудовлетворительно	Ответ обнаруживает непонимание студентом основного содержания учебного материала или допущены существенные ошибки, которые не могут быть исправлены при наводящих вопросах преподавателя

5.1.3 Типовые задания (оценочное средство - Практическое задание) для оценки сформированности компетенции ПК-10:

1. Инструментальные средства поддержки процесса управления требованиями.
2. Инструментальные средства поддержки процесса разработки проекта.
3. Инструментальные средства реализации кода.
4. Инструментальные средства тестирования.
5. Инструментальные средства поддержки процесса управления конфигурациями.
6. Структурные карты.

5.1.4 Типовые задания (оценочное средство - Практическое задание) для оценки сформированности компетенции ПК-8:

1. Анализ предметной области
2. Разработка UML диаграмм
3. Работа с инструментальными средствами, поддерживающими методологию объектно-ориентированного моделирования.
4. Работа с CASE – средствами проектирования программного обеспечения
5. Работа с CASE – средствами кодирования программного обеспечения
6. Работа с CASE – средствами тестирования программного обеспечения

Критерии оценивания (оценочное средство - Практическое задание)

Оценка	Критерии оценивания
зачтено	Ответ полный и правильный на основании изученной теории; теоретический материал и решение поставленных задач изложены в необходимой логической последовательности, грамотный научный язык; ответ самостоятельный. Могут быть допущены две-три не существенные ошибки, исправленные по требованию преподавателя
не зачтено	Ответ обнаруживает непонимание студентом основного содержания учебного материала или допущены существенные ошибки, которые не могут быть исправлены при наводящих вопросах преподавателя

5.1.5 Типовые задания (оценочное средство - Реферат) для оценки сформированности компетенции ПК-10:

1. Определение, характеристики и основные элементы процессного подхода.
2. Процессный подход к моделированию деятельности.
3. Общие сведения, функциональное назначение методологии ARIS.
4. Архитектура ARIS - пять типов представлений, отражающих основные аспекты деятельности организации.
5. Базовая модель ARIS - этапы описания бизнес-процессов.
6. Виды моделей методологии ARIS - основные принципы построения, структура, свойства, составляющие элементы.
7. Использование методологии ARIS в различных областях деятельности.

5.1.6 Типовые задания (оценочное средство - Реферат) для оценки сформированности компетенции ПК-8:

1. Инструментальные средства поддержки процесса управления требованиями.
2. Инструментальные средства поддержки процесса разработки проекта.
3. Инструментальные средства реализации кода.
4. Инструментальные средства тестирования.
5. Инструментальные средства поддержки процесса управления конфигурациями.
6. Семейство стандартов моделирования IDEF. Структурные карты. Диаграммы переходов состояний.
7. Инструментальные средства поддержки методологий функционального моделирования.

Критерии оценивания (оценочное средство - Реферат)

Оценка	Критерии оценивания
отлично	реферативная работа полностью раскрывает основные вопросы теоретического материала. Студент приводит информацию из первоисточников и изданий периодической печати, приводит практические примеры, в докладе отвечает на дополнительные вопросы преподавателя и студентов
хорошо	реферативная работа частично раскрывает основные вопросы теоретического материала. Студент приводит информацию из первоисточников, отвечает на дополнительные вопросы преподавателя и студентов (при докладе), но при этом дает не четкие ответы, без достаточно их аргументации
удовлетворительно	реферативная работа в общих чертах раскрывает основные вопросы теоретического материала. Студент приводит информацию только из учебников. При ответах на дополнительные вопросы в докладе путается в ответах, не может дать понятный и аргументированный ответ
неудовлетворительно	ставится за рефераты, в которых нет информации о проблематике работы и ее месте в контексте других работ по исследуемой теме

5.1.7 Типовые задания (оценочное средство - Тест) для оценки сформированности компетенции ПК-10:

11. Устройство компьютера, специально предназначенное для поддержки разработки программного средства, называется -...

- 1) аппаратным инструментом
- 2) программным инструментом
- 3) программной средой
- 4) инструментарий технологии программирования

12. Редакторы обеспечивают...

- 1) конструирование тех или иных программных продуктов и документов на различных этапах жизненного цикла
- 2) автоматически приводить документы к другой форме представления или переводить документ одного вида к документу другого вида

3) статическую обработку документов, осуществляя различные виды их контроля, выявление определенных их свойств и накопление статистических данных, либо динамический анализ программ

4) выполнять на компьютере описание процессов или отдельных их частей, представленных в виде, отличном от машинного кода

13. Инструменты, поддерживающие процесс выполнения программ, обеспечивают...

1) конструирование тех или иных программных продуктов и документов на различных этапах жизненного цикла

2) автоматический привод документов к другой форме представления или перевод документа одного вида к документу другого вида

3) возможность выполнять на компьютере описание процессов или отдельных их частей, представленных в виде, отличном от машинного кода

4) статическую обработку документов, осуществляя различные виды их контроля, выявление определенных их свойств и накопление статистических данных, либо динамический анализ программ

14. Инструментальная система технологии программирования предназначена для...

1) поддержки всех процессов разработки и сопровождения в течение всего жизненного цикла ПС и ориентирована на коллективную разработку больших программных систем с длительным жизненным циклом

2) автоматического перевода документов к другой форме представления или перевода документа одного вида к документу другого вида

3) поддержки ранних этапов разработки программ и автоматической генерации программ по спецификациям

4) поддержки процессов программирования (кодирования), тестирования и отладки программ

15. Рабочее место компьютерной технологии предназначено для...

1) конструирования тех или иных программных продуктов и документов на различных этапах жизненного цикла

2) автоматического перевода документов к другой форме представления или перевода документа одного вида к документу другого вида

3) поддержки ранних этапов разработки программ и автоматической генерации программ по спецификациям

4) поддержки процессов программирования (кодирования), тестирования и отладки программ

16. Инструментальные среды программирования содержат

1) редактор, анализатор и компилятор

2) редактор, интерпретатор и компилятор

3) интерпретатор, компилятор, преобразователь

4) редактор и интерпретатор

17. Для поддержки разработки программного продукта на разных языках

программирования (например, текстовый редактор, редактор связей или интерпретатор языка целевого компьютера) используют...

1) среду программирования общего назначения

2) языково-ориентированную среду программирования

3) интерпретирующую среду программирования

4) прикладную среду программирования

18. На рисунке представлена классификация

1) инструментальной системы технологии программирования

- 2) инструментальной среды программирования
- 3) рабочего места компьютерной технологии
- 4) языков программирования

19. При использовании компьютерных технологий для разработки ПП жизненный цикл ПП представлен следующей цепочкой:

- 1) прототипирование – кодогенерация – комплексная отладка и тестирование – аттестация, применение, сопровождение
- 2) прототипирование – разработка спецификаций – автоматизированный контроль спецификаций – кодогенерация – комплексная отладка и тестирование – аттестация, применение, сопровождение
- 3) разработка спецификаций – автоматизированный контроль спецификаций – кодогенерация – комплексная отладка и тестирование – аттестация, применение, сопровождение
- 4) прототипирование – разработка спецификаций – кодогенерация – аттестация, применение, сопровождение

20. Основными чертами инструментальной системы технологии программирования являются...

- 1) массовость, дискретность, результативность, определенность, понятность
- 2) комплексность, ориентированность на коллективную разработку, технологическая определенность, интегрированность
- 3) актуальность, непротиворечивость, полнота
- 4) комплексность, актуальность, интегрированность, массовость, понятность

5.1.8 Типовые задания (оценочное средство - Тест) для оценки сформированности компетенции ПК-8:

1. Программное средство, предназначенное для поддержки разработки других программ, называется -...

- А) аппаратным инструментом
- Б) программным инструментом
- В) программной средой
- Г) инструментарий технологии программирования

2. Анализаторы обеспечивают...

- 1) конструирование тех или иных программных продуктов и документов на различных этапах жизненного цикла
- 2) автоматически приводить документы к другой форме представления или переводить документ одного вида к документу другого вида
- 3) статическую обработку документов, осуществляя различные виды их контроля, выявление определенных их свойств и накопление статистических данных, либо динамический анализ программ
- 4) выполнять на компьютере описание процессов или отдельных их частей, представленных в виде, отличном от машинного кода

3. Преобразователи обеспечивают...

- 1) конструирование тех или иных программных продуктов и документов на различных этапах жизненного цикла
- 2) автоматически приводить документы к другой форме представления или переводить документ одного вида к документу другого вида

3) статическую обработку документов, осуществляя различные виды их контроля, выявление определенных их свойств и накопление статистических данных, либо динамический анализ программ

4) выполнять на компьютере описание процессов или отдельных их частей, представленных в виде, отличном от машинного кода

4. Сколько классов инструментальных средств выделяют в инструментальной среде разработки и сопровождения программ?

- 1) 2
- 2) 4
- 3) 3
- 4) 5

5. Среда программирования предназначена для...

- 1) конструирования тех или иных программных продуктов и документов на различных этапах жизненного цикла
- 2) автоматического перевода документов к другой форме представления или перевода документа одного вида к документу другого вида
- 3) поддержки ранних этапов разработки программ и автоматической генерации программ по спецификациям
- 4) поддержки процессов программирования (кодирования), тестирования и отладки программ

6. Инструментальные среды программирования бывают

- 1) языково-ориентированные среды и среды общего назначения
- 2) объектно-ориентированные и языково-ориентированные среды
- 3) среды общего назначения и прикладные среды
- 4) среды общего назначения, прикладные среды, логические и математические среды

7. Для поддержки разработки программного продукта на каком-либо одном языке программирования используют...

- 1) среду программирования общего назначения
- 2) языково-ориентированную среду программирования
- 3) интерпретирующую среду программирования
- 4) прикладную среду программирования

8. Синтаксически-управляемая инструментальная среда программирования базируется на знании

- 1) семантики языка программирования
- 2) синтаксиса языка программирования
- 3) синтаксиса и семантики языка программирования
- 4) основных управляющих структур языка программирования

9. На рисунке представлена классификация

- 1) инструментальной системы технологии программирования
- 2) инструментальной среды разработки и сопровождения программ
- 3) рабочего места компьютерной технологии
- 4) языков программирования

10. Инструментальная система технологии программирования – это...

- 1) программное средство, предназначенное для поддержки разработки других программ
- 2) устройство компьютера, специально предназначенное для поддержки разработки программного средства

- 3) интегрированная совокупность программных и аппаратных инструментов, поддерживающая все процессы разработки и сопровождения больших программных продуктов
- 4) логически связанная совокупность программных и аппаратных инструментов, поддерживающих разработку ПП

Критерии оценивания (оценочное средство - Тест)

Оценка	Критерии оценивания
отлично	85-100% правильных ответов
хорошо	66-84 % правильных ответов
удовлетворительно	50-65 % правильных ответов
неудовлетворительно	меньше 50 % правильных ответов

5.2. Описание шкал оценивания результатов обучения по дисциплине при промежуточной аттестации

Шкала оценивания сформированности компетенций

Уровень сформированности компетенций (индикатора достижения компетенций)	неудовлетворительно	удовлетворительно	хорошо	отлично
	не зачтено	зачтено		
<u>Знания</u>	Уровень знаний ниже минимальных требований. Имели место грубые ошибки	Минимально допустимый уровень знаний. Допущено много негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки, без ошибок
<u>Умения</u>	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки	Продemonстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме	Продemonстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания в полном объеме, но некоторые с недочетами	Продemonстрированы все основные умения. Решены все основные задачи с отдельными незначительными недочетами, выполнены все задания в полном объеме
<u>Навыки</u>	При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки	Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами	Продemonстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продemonстрированы навыки при решении нестандартных задач без ошибок и недочетов

Шкала оценивания при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично», при этом хотя бы одна

		компетенция сформирована на уровне «отлично»
	хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо», при этом хотя бы одна компетенция сформирована на уровне «хорошо»
	удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно».

5.3 Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения на промежуточной аттестации с указанием критериев их оценивания:

5.3.1 Типовые задания (оценочное средство - Контрольные вопросы) для оценки сформированности компетенции ПК-10

1. Классификация по категориям Case-средств
2. Вспомогательные типы Case-средств
3. Объектно-ориентированный подход к разработке ИС
4. Качества организации для успешного внедрения Case-средств
5. Факторы появления Case-технологий
6. Диаграммы «сущность-связь»
7. Язык графического описания UML
8. Диаграмма композитной структуры
9. Диаграмма объектов
10. Диаграмма деятельности
11. IDEF
12. Методология функционального моделирования ИС
13. Иерархия диаграмм
14. Характеристика современных Case-средств
15. Программный продукт ARIS Express
16. Архитектура ARIS
17. Применение имитационного моделирования
18. Дискретно-событийное моделирование
19. Области применения имитационного моделирования
20. Построение концептуальной модели системы
21. Анализ задачи моделирования
22. Выдвижение гипотез и принятие предположений
23. Установление основного содержания модели
24. Определение процедур аппроксимации
25. Проверка достоверности модели

5.3.2 Типовые задания (оценочное средство - Контрольные вопросы) для оценки сформированности компетенции ПК-8

1. Компоненты интегрированного Case-средства
2. Классификация по типам Case-средств
3. Структурный подход к разработке ИС

4. Факторы, усложняющие определение возможного эффекта от использования Case-средств
5. Проблемы использования Case-средств
6. Структурный системный анализ
7. Диаграммы классов
8. Диаграмма компонентов
9. Диаграмма развёртывания
10. Диаграмма пакетов
11. Преимущества UML
12. Диаграммы переходов состояний
13. Состав функциональной модели
14. Типы связей между функциями
15. Методология ARIS
16. Основные элементы, используемые в нотации ARIS
17. Имитационное моделирование
18. Виды имитационного моделирования
19. Системная динамика
20. Основные этапы компьютерного моделирования
21. Постановка задачи машинного моделирования
22. Определение требований к исходной информации
23. Определение параметров и переменных
24. Обоснование критериев оценки эффективности системы
25. Описание концептуальной модели
26. Составление технической документации

Критерии оценивания (оценочное средство - Контрольные вопросы)

Оценка	Критерии оценивания
зачтено	ответ полный и правильный на основании изученной теории; теоретический материал и решение поставленных задач изложены в необходимой логической последовательности, грамотный научный язык; ответ самостоятельный. Могут быть допущены две-три незначительные ошибки, исправленные по требованию преподавателя
не зачтено	ответ обнаруживает непонимание студентом основного содержания учебного материала или допущены существенные ошибки, которые не могут быть исправлены при наводящих вопросах преподавателя

6. Учебно-методическое и информационное обеспечение дисциплины (модуля)

Основная литература:

1. Гагарина Лариса Геннадьевна. Технология разработки программного обеспечения : Учебное пособие / Московский институт электронной техники. - 1. - Москва : Издательский Дом "ФОРУМ", 2022. - 400 с. - ВО - Бакалавриат. - ISBN 978-5-8199-0707-8. - ISBN 978-5-16-104071-3. - ISBN 978-5-16-013286-0., <https://e-lib.unn.ru/MegaPro/UserEntry?>

Action=FindDocs&ids=770689&idb=0.

2. Тузовский А. Ф. Проектирование и разработка web-приложений / Тузовский А. Ф. - Москва : Юрайт, 2022. - 218 с. - (Высшее образование). - URL: <https://urait.ru/bcode/490128> (дата обращения: 05.01.2022). - ISBN 978-5-534-00515-8 : 729.00. - Текст : электронный // ЭБС "Юрайт"., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=788234&idb=0>.

3. Яшин Владимир Николаевич. Информатика: программные средства персонального компьютера : Учебное пособие / Самарский государственный технический университет. - Москва : ООО "Научно-издательский центр ИНФРА-М", 2018. - 236 с. - ВО - Бакалавриат. - ISBN 978-5-16-006788-9. - ISBN 978-5-16-100158-5., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=621527&idb=0>.

Дополнительная литература:

1. Гагарина Лариса Геннадьевна. Введение в теорию алгоритмических языков и компиляторов : Учебное пособие / Московский институт электронной техники; Московский институт электронной техники. - 1. - Москва : Издательский Дом "ФОРУМ", 2022. - 176 с. - ВО - Бакалавриат. - ISBN 978-5-8199-0404-6. - ISBN 978-5-16-110113-1., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=834350&idb=0>.

2. Липунцов Юрий Павлович. Прикладные программные продукты для экономистов. Основы информационного моделирования : Учебное пособие / Московский государственный университет им. М.В. Ломоносова, экономический факультет. - Москва : Издательство "Проспект", 2014. - 252 с. - Аспирантура. - ISBN 978-5-392-17845-2., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=601697&idb=0>.

3. Токарев К. Е. Инструментальные методы и программные средства в экономике : учебное пособие / Токарев К. Е., Рогачев А. Ф. - Волгоград : Волгоградский ГАУ, 2015. - 92 с. - Библиогр.: доступна в карточке книги, на сайте ЭБС Лань. - Книга из коллекции Волгоградский ГАУ - Экономика и менеджмент., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=716634&idb=0>.

Программное обеспечение и Интернет-ресурсы (в соответствии с содержанием дисциплины):

Лицензионное программное обеспечение: Операционная система Windows.

Лицензионное программное обеспечение: Microsoft Office.

Профессиональные базы данных и информационные справочные системы

Российский индекс научного цитирования (РИНЦ), платформа Elibrary: национальная информационно-аналитическая система. Адрес доступа: http://elibrary.ru/project_risc.asp

ГАРАНТ. Информационно-правовой портал [Электронный ресурс].— Адрес доступа: <http://www.garant.ru>

Свободно распространяемое программное обеспечение:

программное обеспечение LibreOffice;

программное обеспечение Yandex Browser;

программное обеспечение Paint.NET;

программное обеспечение 1С:

- * "Бухгалтерия предприятия", редакция 3.0, см. <http://v8.1c.ru/buhv8/> ,
- * "Управление торговлей", редакция 11.1, см. <http://v8.1c.ru/trade/> ,
- * "Зарплата и управление персоналом", редакция 3.0, см. <http://v8.1c.ru/hrm/> ,
- * "Управление небольшой фирмой", редакция 1.5, см. <http://v8.1c.ru/small.biz/> ,
- * "ERP Управление предприятием 2.0", см. <http://v8.1c.ru/erp/> .
- * "Бухгалтерия государственного учреждения", редакция 1.0, см. <http://v8.1c.ru/stateacc/> ,
- * "Зарплата и кадры государственного учреждения", редакция 1.0, <http://v8.1c.ru/statehrm/> .

программное обеспечение PascalABC.NET

Электронные библиотечные системы и библиотеки:

Электронная библиотечная система "Лань" <https://e.lanbook.com/>

Электронная библиотечная система "Консультант студента" <http://www.studentlibrary.ru/>

Электронная библиотечная система "Юрайт" <http://www.urait.ru/ebs>

Электронная библиотечная система "Znanium" <http://znanium.com/>

Электронно-библиотечная система Университетская библиотека ONLINE <http://biblioclub.ru/>

Фундаментальная библиотека ННГУ www.lib.unn.ru/

Сайт библиотеки Арзамасского филиала ННГУ. – Адрес доступа: lib.arz.unn.ru

Ресурс «Массовые открытые онлайн-курсы Нижегородского университета им. Н.И. Лобачевского»
<https://mooc.unn.ru/>

Портал «Современная цифровая образовательная среда Российской Федерации»
<https://online.edu.ru/public/promo>

7. Материально-техническое обеспечение дисциплины (модуля)

Учебные аудитории для проведения учебных занятий, предусмотренных образовательной программой, оснащены мультимедийным оборудованием (проектор, экран), техническими средствами обучения, компьютерами.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ по направлению подготовки/специальности 09.03.03 - Прикладная информатика.

Автор(ы): Сазанов Александр Анатольевич.

Рецензент(ы): Фокеев Максим Игоревич, кандидат педагогических наук.

Заведующий кафедрой: Нестерова Лариса Юрьевна, кандидат педагогических наук.

Программа одобрена на заседании методической комиссии от 10.01.2024 г., протокол № 1.