

МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»
(РУТ (МИИТ))



Рабочая программа дисциплины (модуля),
как компонент образовательной программы
высшего образования - программы магистратуры
по направлению подготовки
09.04.03 Прикладная информатика,
утвержденной первым проректором РУТ (МИИТ)
Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Современные технологии разработки программного обеспечения

Направление подготовки: 09.04.03 Прикладная информатика

Направленность (профиль): Информационные технологии управления
социально-экономическими системами

Форма обучения: Очная

Рабочая программа дисциплины (модуля) в виде
электронного документа выгружена из единой
корпоративной информационной системы управления
университетом и соответствует оригиналу

Простая электронная подпись, выданная РУТ (МИИТ)
ID подписи: 564169
Подписал: заведующий кафедрой Каргина Лариса Андреевна
Дата: 23.03.2023

1. Общие сведения о дисциплине (модуле).

Целью освоения учебной дисциплины являются:

-познакомить студентов с рядом современных технологий разработки ПО.

Задачами освоения дисциплины являются:

-формирование у учащихся правил работы с основными принципами промышленной разработки ПО,

-выработать навыки использования регулярных выражений в обработке текстов, систем контроля версий и автоматизированного тестирования ПО.

- разработка оригинальных алгоритмов для решения задач научного исследования.

2. Планируемые результаты обучения по дисциплине (модулю).

Перечень формируемых результатов освоения образовательной программы (компетенций) в результате обучения по дисциплине (модулю):

ОПК-2 - Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач;

ОПК-5 - Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем;

ОПК-8 - Способен осуществлять эффективное управление разработкой программных средств и проектов.;

ПК-2 - Способен применять современные методы и инструментальные средства прикладной информатики для автоматизации и информатизации решения прикладных задач различных классов и создания ИС ;

ПК-3 - Способен проектировать архитектуру ИС предприятий и организаций и принимать эффективные проектные решения в условиях неопределенности и риска.

Обучение по дисциплине (модулю) предполагает, что по его результатам обучающийся будет:

Уметь:

- применять регулярные выражения и различные кодировки текста;
- использовать системы контроля версий;
- производить автоматизированное тестирование ПО;
- разрабатывать программы для работы в различных операционных системах;

- создавать кроссплатформенные программы на уровне выполнения;
- создавать программы на кроссплатформенных интерпретируемых языках
- выбирать оптимальный программный продукт и модели информационных технологий из нескольких возможных для решения прикладной задачи;
- выбрать программный продукт и технологии для решения задачи с учетом конкретной предметной области;
- разрабатывать сервисные программы и сервисные оболочки при разработке приложений с учетом конкретной предметной области.

Знать:

- понятия регулярных выражений, кодировок текста, системы контроля версий и автоматизированного тестирования ПО;
- модели, процессы, этапы жизненного цикла ИС;
- процессы реинжиниринга ИС;
- современные методологии проектирования ИС, распространённые технологические стандарты;
- методики типового проектирования ИС.

Владеть:

- навыками применения регулярные выражения и различные кодировки текста;
- навыками использования системы контроля версий;
- навыками автоматизированного тестирования ПО;
- навыками применения информационных технологий и творческого подхода при решении стандартных и нестандартных задач;
- навыками выбора программных продуктов и мультиплатформенных технологий для решения задачи;
- навыками использования сервисных программ и сервисных оболочек для решения задачи;
- навыками работы с объектно-ориентированными языками программирования;
- навыками работы с интерпретируемыми языками программирования;
- навыками создания кроссплатформенных приложений.

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 5 з.е. (180

академических часа(ов).

3.2. Объем дисциплины (модуля) в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении учебных занятий:

Тип учебных занятий	Количество часов	
	Всего	Сем. №1
Контактная работа при проведении учебных занятий (всего):	32	32
В том числе:		
Занятия лекционного типа	16	16
Занятия семинарского типа	16	16

3.3. Объем дисциплины (модуля) в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации составляет 148 академических часа (ов).

3.4. При обучении по индивидуальному учебному плану, в том числе при ускоренном обучении, объем дисциплины (модуля) может быть реализован полностью в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации.

4. Содержание дисциплины (модуля).

4.1. Занятия лекционного типа.

№ п/п	Тематика лекционных занятий / краткое содержание
1	Сравнительный обзор современных языков программирования. Регулярные выражения Рассматриваемы вопросы: - определение критериев сравнения языков программирования: компилятор или интерпретатор, межплатформенность, поддержка различных архитектур, время работы готовой программы, набор библиотек для различных целей, интегрированные среды разработки, спектр решаемых задач,

№ п/п	Тематика лекционных занятий / краткое содержание
	<p>поддержка разработки на основе различных парадигм программирования</p> <ul style="list-style-type: none"> - типы регулярных выражений. PCRE Основные операции с использованием регулярных выражений Квантификаторы Специальные метасимволы «Жадные» и «ленивые» регулярные выражения Группировки Направление просмотра
2	<p>Кодировки текста. Системы контроля версий</p> <p>Рассматриваемы вопросы:</p> <ul style="list-style-type: none"> - история появления кодировок. Кодировки для кириллического текста. Многобайтовые кодировки. Unicode. Передача данных через почтовый протокол: base64 - история систем контроля версий. Основные понятия. CVS, SVN. Распределённые системы контроля версий. Git Основные сценарии работы
3	<p>Языки программирования - наследники языков C и C++. Языки программирования Python и Ruby</p> <p>Рассматриваемы вопросы:</p> <ul style="list-style-type: none"> - трудности и недостатки языка C++, которые привели к разработке языков Java и C#. Объектно-ориентированные возможности языков Java и C#. Межплатформенная ориентация языка Java. Особенности использования языков Java и C# для разработки приложений в web-архитектуре. Язык программирования Kotlin как современная модификация языка Java. Языки Go, JavaScript, Objective-C как языки, поддерживающие философию языка C - язык программирования Python как универсальный язык программирования. Особенности синтаксиса языка Python. Особая линейка объектов-коллекций и библиотек их обработки. Ориентация языка Python на научные вычисления. Язык Ruby и особенности его синтаксиса. Особенности реализации объектов в Python и Ruby
4	<p>Языки программирования для webю Автоматизация тестирования web-приложений с помощью Selenium WebDriver</p> <p>Рассматриваемы вопросы:</p> <ul style="list-style-type: none"> - особенности современной web-разработки. Серверные технологии программирования: ASP.NET, PHP, Django, Spring framework и пр. Клиентские технологии программирования на примере технологий на основе языка программирования JavaScript. Особенности синтаксиса языка JavaScript. Особенности формирования пользовательского интерфейса и управления им на основе html-модели и использования языка JavaScript. Применение JavaScript для консольных приложений на примере технологии NodeJS. - установка Python3 в Windows Переменная среды PATH Создание виртуального окружения для Selenium WebDriver. Установка Selenium в Windows. Установка драйвера для браузера. Пример работы Selenium WebDriver Напоминание о структуре веб-страницы Поиск элементов на веб-странице. Тестовые фреймворки
5	<p>Современные системы управления базами данных</p> <p>Рассматриваемы вопросы:</p> <ul style="list-style-type: none"> - современные реляционные системы управления базами данных (СУБД): сравнение Oracle, MySQL, MS SQL, PostgreSQL. Объектно-реляционные возможности PostgreSQL. NoSQL СУБД: графовые СУБД на примере Neo4J, СУБД «ключ-значение» на примере Redis, документоориентированные СУБД на примере MongoDB, столбцовые СУБД на примере Cassandra
6	<p>Современные информационные и компьютерные технологии</p> <p>Рассматриваемы вопросы:</p> <ul style="list-style-type: none"> - обзор и сравнительная характеристика различных современных информационных технологий (распределенные приложения, офисные приложения, web-приложения, мобильные приложения, интеллектуальные системы), - обзор с сравнительная характеристика современных парадигм программирования (объектно-ориентированная концепция, функциональное программирование, аспектно-ориентированное программирование и пр.)
7	<p>Современные технологии разработки программного обеспечения</p>

№ п/п	Тематика лекционных занятий / краткое содержание
	Рассматриваемы вопросы: - классические подходы в технологии разработки программного обеспечения (итеративный процесс, стандарты документации, интеграция, тестирование, шаблоны проектирования), - Agile-методологии разработки программного обеспечения (гибкая методология разработки): особенности применения (методики экстремального программирования, DSDM, Scrum, FDD)
8	Управление проектами в сфере разработки программного обеспечения Рассматриваемы вопросы: - Принципы организации процесса разработки: основные этапы разработки программного обеспечения – анализ требований, проектирование, программирование, тестирование и документирование. Отличия организации работы на этих этапах для различных методологий разработки программного обеспечения. - Деловая игра, заключающаяся в командной разработке программного проекта в стиле одной из Agile-методологий разработки

4.2. Занятия семинарского типа.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Определение критериев сравнения языков программирования: компилятор или интерпретатор В результате работы на практическом занятии студент учится: - определять критерии языка программирования
2	Основные приемы разработки программ на C++, Java, C#, Kotlin, Go, JavaScript, Objective-C (по выбору студентов) В результате работы на практическом занятии формируется навык: - работы по разработке программ на языке программирования C++ (C#, Objective-C), - работы по разработке программ на языке программирования Java, C#, Kotlin, Go
3	Языки программирования Python и Ruby В результате работы на практическом занятии студент осваивает: - принципы разработки программ на Python и Ruby
4	Языки программирования для web На практическом занятии отрабатывается: - клиентские технологии программирования на примере технологий на основе языка программирования JavaScript
5	Современные системы управления базами данных В результате работы на практическом занятии студент осваивает: - современные реляционные системы управления базами данных (СУБД): MySQL, MS SQL, PostgreSQL
6	Современные информационные и компьютерные технологии В результате работы на практическом занятии студент учится: - обзору и сравнительной характеристике различных современных информационных технологий, современных парадигм программирования
7	Современные технологии разработки программного обеспечения В результате работы на практическом занятии формируется навык: - применения на практике Agile-методологии разработки программного обеспечения и методики экстремального программирования, DSDM, Scrum, FDD
8	Управление проектами в сфере разработки программного обеспечения

№ п/п	Тематика практических занятий/краткое содержание
	На практическом занятии отрабатывается: - деловая игра, заключающаяся в командной разработке программного проекта в стиле одной из Agile-методологий разработки

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Подготовка к практическим занятиям
2	Работа с лекционным материалом
3	Работа с литературой
4	Подготовка к экзамену
5	Подготовка к промежуточной аттестации.
6	Подготовка к текущему контролю.

5. Перечень изданий, которые рекомендуется использовать при освоении дисциплины (модуля).

№ п/п	Библиографическое описание	Место доступа
1	Принципы, паттерны и методологии разработки программного обеспечения : учебное пособие для вузов / С. А. Чернышев. — Москва : Издательство Юрайт, 2022. — 176 с. — ISBN 978-5-534-14383-6.	Юрайт [сайт]. — URL: https://urait.ru/bcode/497029 (дата обращения: 19.04.2023).— Текст : электронный
2	Технология разработки программного обеспечения : учебное пособие для среднего профессионального образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва : Издательство Юрайт, 2022. — 235 с. — ISBN 978-5-534-05047-9.	Юрайт [сайт]. — URL: https://urait.ru/bcode/492496 (дата обращения: 19.04.2023).— Текст : электронный
3	Технологии и методы программирования : учебное пособие для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва : Издательство Юрайт, 2022. — 235 с.— ISBN 978-5-534-02816-4.	Юрайт [сайт]. — URL: https://urait.ru/bcode/489920 (дата обращения: 19.04.2023).— Текст : электронный

6. Перечень современных профессиональных баз данных и информационных справочных систем, которые могут использоваться при освоении дисциплины (модуля).

Обязательный набор:

Научно-техническая библиотека РУТ (МИИТ): <http://library.miiit.ru>

СПС КонсультантПлюс <http://www.consultant.ru>

Электронная база данных «Scopus» <http://www.scopus.com>

Научная электронная библиотека elibrary (<http://elibrary.ru>)

7. Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства, необходимого для освоения дисциплины (модуля).

Обязательный набор:

Доступ к Internet,

Офисный пакет приложений Microsoft Office

Бесплатный мировой стандарт, который используется для просмотра, печати и комментирования документов в формате PDF Acrobat Reader

Файловый архиватор с высокой степенью сжатия данных 7-Zip

Интерпретатор языка программирования Python 3

Oracle VirtualBox - программный продукт виртуализации для Oracle

GNU/Linux (любой, общего назначения, с поддержкой репозитория) - семейство Unix-подобных операционных систем на базе ядра Linux, включающих тот или иной набор утилит и программ проекта GNU

стандартный набор утилит Linux и Windows

8. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю).

Для проведения лекционных занятий необходима аудитория с мультимедиа аппаратурой. Для проведения практических занятий требуется аудитория, оснащенная мультимедиа аппаратурой и ПК с необходимым программным обеспечением, и подключением к сети интернет.

9. Форма промежуточной аттестации:

Экзамен в 1 семестре.

10. Оценочные материалы.

Оценочные материалы, применяемые при проведении промежуточной аттестации, разрабатываются в соответствии с локальным нормативным актом РУТ (МИИТ).

Авторы:

доцент, к.н. кафедры
«Информационные системы
цифровой экономики»

О.В. Медникова

Согласовано:

Заведующий кафедрой ИСЦЭ
Председатель учебно-методической
комиссии

Л.А. Каргина

М.В. Ишханян