

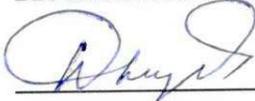
МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ИМЕНИ В.Ф. УТКИНА»

Факультет вычислительной техники
Кафедра «Вычислительная и прикладная математика»

СОГЛАСОВАНО

Декан факультета
вычислительной техники


Д.А. Перепелкин
« 26 » 06 2020 г.

УТВЕРЖДАЮ

Проректор РОПиМД



А.В. Корячко
« 26 » 06 2020 г.

Заведующий кафедрой:

ВПМ  Г.В. Овечкин
« 26 » 06 2020 г.

РАБОЧАЯ ПРОГРАММА

дисциплины

Б1.В.13 «Web–программирование»

Направление подготовки — 09.03.04 «Программная инженерия»

Уровень подготовки — академический бакалавриат

Направленность (профиль) подготовки

Программная инженерия

Квалификация (степень) выпускника — бакалавр

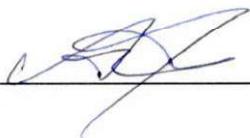
Форма обучения — очная

Рязань 2020 г.

ЛИСТ СОГЛАСОВАНИЙ

Рабочая программа составлена с учетом требований федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.04 – «Программная инженерия»,
утвержденного _____ №920 от 19.09.2017 г. _____
(дата утверждения ФГОС ВО)

Разработчик:
доцент кафедры ВПМ



О.В. Антипов

Программа рассмотрена и одобрена на заседании кафедры ВПМ «11» июня 2020 г.,
протокол № 11.

Заведующий кафедрой ВПМ



Г.В. Овечкин

1. ЦЕЛЬ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью изучения дисциплины является формирование у студентов компетентности в области разработки программного обеспечения с использованием современных методов Web-программирования на основе представлений о современных языках программирования, в частности, ознакомить студентов с базовыми возможностями языка Javascript, с его сильными и слабыми сторонами, а также дать понятие об области его применения и подходах к решению стандартных алгоритмических задач.

Основная задача дисциплины – подготовка специалистов, способных самостоятельно и эффективно использовать принципы и понятия современных технологий программирования при разработке прикладных программ. В частности, познакомить студентов с прототипным подходом в программировании, который имеет отличия в работе с объектами по сравнению с традиционными класс-ориентированными языками. Познакомить студентов с такими гибкими возможностями языка Javascript как функции как объекты первого класса, объекты как списки, анонимные функции, замыкания, автоматическое приведение типов.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Дисциплина относится к вариативной части профессиональных дисциплин блока № 1 (Б1.В.13), изучается по очной форме обучения на 3 курсе в 6 семестре, базируется на знаниях, полученных в ходе изучения дисциплин: «Информатика», «Алгоритмические языки и программирование», «Объектно-ориентированное программирование», «Функциональное программирование», «Основы программной инженерии».

Пререквизиты дисциплины. Для освоения дисциплины обучающиеся должны

- *знать:*
 - основные понятия теории алгоритмов и базовые принципы алгоритмизации;
 - основы процедурного программирования, архитектуру программ, стандартные управляющие конструкции, иметь понятие о структурах и типах данных;
 - стандартные алгоритмы обработки структур данных;
 - способы представления данных в памяти компьютера;
 - принципы нисходящего проектирования и структурного программирования;
 - основы объектно-ориентированного программирования;
 - основы HTML и CSS;
- *уметь:*
 - применять свои знания при решении различных предметных задач;
 - работать в средах программирования;
 - разрабатывать, отлаживать и выполнять императивные программы;
- *владеть:*
 - навыками алгоритмизации, разработки и отладки программ;
 - методами и приемами анализа и структурирования сложных программ
 - базовыми навыками вёрстки HTML и CSS.

Постреквизиты дисциплины. Полученные знания используются при изучении дисциплин «Параллельное программирование», «Конструирование программного обеспечения», «Разработка компиляторов», «Разработка системных утилит», при выполнении НИР и при подготовке выпускной квалификационной работы.

В результате изучения дисциплины студенты должны

- *иметь представление и знать:*
 - представление о динамической и слабой типизации, автоматическом управлении памятью;
 - понимание прототипного подхода в объектно-ориентированном программировании;
 - основные положения современной технологии Web-программирования;
 - назначение и области применения языка Javascript, синтаксис и семантику;

- операторы и операции языка, специальные значения;
- стандартные типы данных языка и встроенные объекты;
- *иметь навыки и уметь:*
 - навыки и умение отладки Javascript-программ с помощью встроенных средств отладки в браузере firefox;
 - навыки и умение работы с функциями как с объектами, уметь применять правильно замыкания и функции высших порядков.

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Процесс изучения дисциплины направлен на формирование следующих компетенций в соответствии с ФГОС ВО, ПООП (при наличии) по данному направлению подготовки, а также компетенций (при наличии), установленных университетом.

Рекомендуемые профессиональные компетенции выпускников и индикаторы их достижения (при наличии)

Задача ПД	Объект или область знания	Код и наименование профессиональной компетенции	Код и наименование индикатора достижения профессиональной компетенции	Обоснование (ПС, анализ опыта)
Направленность (профиль), специализация: Программная инженерия				
Тип задач профессиональной деятельности: проектный				
Формирование требований к информатизации и автоматизации прикладных процессов, формализация предметной области проекта; технико-экономическое обоснование проектных решений и составление технического задания на разработку ПП; проектирование программно-аппаратных средств в соответствии с техническим заданием; применение современных инструментальных средств при разработке ПО; документирование компонентов ИС на всех стадиях ЖЦ	Прикладные и информационные процессы Информационные технологии Программное обеспечение	ПК-4. Владение навыками моделирования, анализа и использования формальных методов конструирования ПО	ИД – 1 _{ПК-4} <i>Знать:</i> основы моделирования и формальные методы конструирования ПО. ИД – 2 _{ПК-4} <i>Уметь:</i> использовать формальные методы моделирования и конструирования ПО. ИД – 3 _{ПК-4} <i>Владеть:</i> методами формализации, моделирования и конструирования ПО.	06.001 Программист 06.004 Специалист по тестированию в области информационных технологий 06.022 Системный аналитик 06.028 Системный программист
Тип задач профессиональной деятельности: производственно-технологический				
Проведение работ по инсталляции ПО АС и загрузки БД; настройка параметров ИС и тестирование результатов настройки; ведение технической документации; техническое сопровождение ИС в процессе эксплуатации;	Информационные технологии Программное обеспечение	ПК-10. Владение навыками использования различных технологий разработки ПО	ИД – 1 _{ПК-10} <i>Знать:</i> современные технологии разработки ПО. ИД – 2 _{ПК-10} <i>Уметь:</i> использовать современные технологии разработки ПО.	06.001 Программист 06.004 Специалист по тестированию в области информационных технологий

применение Web-технологий при реализации удаленного доступа в системах клиент-сервер и распределенных вычислений			ИД – 3 _{ПК-10} <i>Владеть:</i> навыками использования современных технологий разработки ПО.	06.022 Системный аналитик 06.028 Системный программист
--	--	--	---	---

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. Объем дисциплины по семестрам (курсам) и видам занятий в зачетных единицах

с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам занятий) и на самостоятельную работу обучающихся

Общая трудоемкость изучения дисциплины для очной формы обучения составляет 3 зачетных единиц (ЗЕ), 108 академических часа.

Объем дисциплины	Всего часов	Семестр 6
Общая трудоемкость дисциплины, в том числе:	108	108
1. Контактная работа обучающихся с преподавателем (всего), в том числе:	48,25	48,25
Лекции	24	24
лабораторные работы	16	16
практические занятия	8	8
иная контактная работа (ИКР)	0,25	0,25
консультация	-	-
2. Самостоятельная работа	51	51
3. Курсовой проект	-	-
4. Контроль	8,75	8,75
Вид промежуточной аттестации		Зачет

4.2. Содержание дисциплины, структурированное по разделам и темам

В структурном отношении программа дисциплины представлена следующими разделами и темами:

Раздел 1. Гипертекстовая информационная система и введение в Javascript.

Тема 1. Появление технологии WEB. Гипертекст. Гипертекстовая информационная система. Технология WorldWideWeb. URL, HTTP, HTML.

Тема 2. Появление языка Javascript, задачи и область его применения. Описание языка JavaScript. Основные характеристики языка JavaScript. Основные способы функционального применения языка JavaScript. Отладка Javascript.

Раздел 2. Лексическая структура JavaScript.

Тема 1. Алфавит и синтаксис языка Javascript. Регистр букв. Пробелы, переводы строк и символы управления форматом.

Тема 2. Семантика языка Javascript. Комментарии. Литералы. Идентификаторы. За-резервированные слова. Точки с запятой.

Раздел 3. Типы данных. Значения. Переменные.

Тема 1. Типы данных. Типы данных в Javascript. Простые типы. Объекты. Функции. Сборка мусора. Изменяемые и неизменяемые типы данных.

Тема 2. Значения и переменные. Числа. Целые литералы. Литералы вещественных чисел. Дата и время. Логические значения. Строки. Значения null и undefined. Объявление переменных. Область видимости переменной.

Раздел 4. Операции и операторы.

Тема 1. Операции в Javascript. Приоритет выполнения операций, их назначение и примеры использования.

Тема 2. Встроенные математические функции, работа со строками и специальные значения. Объект Math. Основные методы. Работа со строками. Специальные значения. Преобразования типов. Преобразование и равенство. Явные преобразования.

Тема 3. Операторы языка. Оператор if. Оператор switch. Операторы while, do...while. Оператор for. Примеры использования.

Раздел 5. Массивы.

Тема 1. Понятие массива и работа с его элементами. Коллекция. Массивы и их характеристики. Создание массивов. Чтение, запись, добавление и удаление элементов массива. Обход элементов массива. Многомерные массивы.

Тема 2. Методы класса Array. Метод join. Метод reverse. Метод concat. Методы класса Array. Метод slice. Метод splice. Методы push() и pop(), unshift() и shift() для работы с массивом как со стеком.

Раздел 6. Объекты.

Тема 1. Объекты и их свойства. Понятие объекта в языке Javascript. Свойства объектов. Создание объектов. Получение, изменение, удаление свойств объекта. Проверка существования свойств. Перечисление свойств.

Тема 2. Сериализация и клонирование объектов. Сериализация объектов. Формат JSON. Методы JSON.stringify() и JSON.parse(). Передача объектов по ссылке. Клонирование объектов.

Раздел 7. Функции.

Тема 1. Базовое использование функций в языке Javascript. Понятие функции в языке Javascript. Параметры и аргументы. Возвращаемое значение. Контекст вызова функций. Определение функций. Вложенные функции. Вызов функций. Вызов функций как методов. Необязательные аргументы.

Тема 2. Расширенное использование функций в языке Javascript. Составление цепочек вызовов методов. Списки аргументов функций переменной длины. Функции как данные. Определение собственных свойств функций. Функции как пространства имён. Замыкания. Методы call и apply. Метод bind. Конструктор Function. Его особенности. Функции высшего порядка. Мемоизация функций.

Раздел 8. Классы объектов.

Тема 1. Классы в языке Javascript. Понятие классов объектов. Экземпляры. Прототипы. Наследование. Функция inherit. Классы и прототипы. Фабричная функция.

Тема 2. Применение функций-конструкторов в языке Javascript. Классы и конструкторы. Конструкторы и идентификация класса. Свойство constructor. Нарращивание возможностей классов.

**4.3. Разделы дисциплины и трудоемкость по видам учебных занятий
(в академических часах)**

№ п/п	Тема	Общая трудо- емкость, всего часов	Контактная работа обучающихся с преподавателем				Само- стоятель- ная работа обучаю- щихся
			всего	лекции	лабора- торные работы	прак- тичес- кие заня- тия	
1	2	3	4	5	6	7	8
1.	Гипертекстовая информа- ционная система и введе- ние в Javascript	8	4	2	0	2	4
1.1.	<i>Появление технологии WEB</i>	4	2	1		1	2
1.2.	<i>Появление языка Javascript, задачи и область его приме- нения</i>	4	2	1		1	2
2.	Лексическая структура Ja- vaScript	6	4	2		2	2
2.1.	<i>Алфавит и синтаксис языка Javascript</i>	3	2	1		1	1
2.2.	<i>Семантика языка Javascript</i>	3	2	1		1	1
3.	Типы данных. Значения. Переменные	10	4	2		2	6
3.1.	<i>Типы данных</i>	6	2	1		1	4
3.2.	<i>Значения и переменные</i>	4	2	1		1	2
4.	Операции и операторы	12	6	4		2	6
4.1.	<i>Операции в Javascript</i>	3,5	1,5	1		0,5	2
4.2.	<i>Встроенные математиче- ские функции, работа со строками и специальные значения</i>	3,5	1,5	1		0,5	2
4.3.	<i>Операторы языка</i>	5	3	2		1	2
5.	Массивы	14	6	2	4		8
5.1.	<i>Понятие массива и работа с его элементами</i>	7	3	1	2		4
5.2.	<i>Методы класса Array</i>	7	3	1	2		4
6.	Объекты	16	8	4	4		8
6.1.	<i>Объекты и их свойства</i>	8	4	2	2		4
6.2.	<i>Сериализация и клонирова- ние объектов</i>	8	4	2	2		4

7.	Функции	16	8	4	4		8
7.1.	<i>Базовое использование функций в языке Javascript</i>	8	4	2	2		4
7.2.	<i>Расширенное использование функций в языке Javascript</i>	8	4	2	2		4
8.	Классы объектов	17	8	4	4		9
8.1.	<i>Классы в языке Javascript</i>	8	4	2	2		4
8.2.	<i>Применение функций-конструкторов в языке Javascript</i>	9	4	2	2		5
	Зачет	9					9
	Всего:	108	48	24	16	8	60

4.3.1. Лекционные занятия

№ п/п	Темы лекционных занятий	Трудоемкость (час.)	Формируемые компетенции	Форма контроля
1.	Гипертекстовая информационная система и введение в Javascript. Появление технологии WEB. Появление языка Javascript, задачи и область его применения	2	ПК-4, ПК-10	зачет
2.	Лексическая структура JavaScript	2	ПК-4, ПК-10	зачет
3.	Типы данных. Значения. Переменные	2	ПК-4, ПК-10	зачет
4.	Операции и операторы	4	ПК-4, ПК-10	зачет
5.	Массивы. Методы класса Array	2	ПК-4, ПК-10	зачет
6.	Объекты и их свойства	2	ПК-4, ПК-10	зачет
7.	Сериализация и клонирование объектов	2	ПК-4, ПК-10	зачет
8.	Функции. Базовое использование функций в языке Javascript	2	ПК-4, ПК-10	зачет
9.	Расширенное использование функций в языке Javascript	2	ПК-4, ПК-10	зачет
10.	Классы объектов в языке Javascript	2	ПК-4, ПК-10	зачет
11.	Применение функций-конструкторов в языке Javascript	2	ПК-4, ПК-10	зачет

4.3.2. Практические занятия (семинары)

№ п/п	Тематика практических занятий (семинаров)	Трудоемкость (час.)	Формируемые компетенции	Форма контроля
1.	Гипертекстовая информационная система и введение в Javascript	2	ПК-4, ПК-10	зачет
2.	Лексическая структура JavaScript	2	ПК-4, ПК-10	зачет
3.	Типы данных. Значения. Переменные	2	ПК-4, ПК-10	зачет
4.	Операции и операторы	2	ПК-4, ПК-10	зачет

План практических занятий

Тема 1. Гипертекстовая информационная система и введение в Javascript.

Технология WEB. Гипертекст. Гипертекстовая информационная система. Технология WorldWideWeb. Языки URL, HTTP, HTML. *Язык Javascript, задачи.* Основные характеристики. Основные способы функционального применения. Отладка.

Цель занятия. Закрепить теоретические знания о технологии прототипного программирования и поддерживающих языков программирования; об их синтаксисе и семантике, особенностях описания, нюансах выполнения, написанных на них программ.

Задача закрепления теоретических знаний и практических навыков: студент должен знать существующие технологии и языки программирования прототипного программирования, их классификацию; существующие парадигмы организации этих языков.

Форма проведения: подготовка докладов для выступления на практическом занятии и коллективной дискуссии.

Учебно-методическая литература [1 – 17].

Тема 2. Лексическая структура JavaScript.

Алфавит и синтаксис языка Javascript. Регистр букв. Пробелы, переводы строк и символы управления форматом. *Семантика языка Javascript.* Комментарии. Литералы. Идентификаторы. Зарезервированные слова. Точки с запятой.

Цель занятия. Освоить основы лексической структуры JavaScript.

Задачи закрепления теоретических знаний и практических навыков: студент должен знать основы лексической, синтаксической и семантической структуры языка, генерации и оптимизации кода программы.

Форма проведения: подготовка докладов для выступления на практическом занятии и коллективной дискуссии.

Учебно-методическая литература [1 – 17].

Тема 3. Типы данных. Значения. Переменные.

Типы данных. Типы данных в Javascript. Простые типы. Объекты. Функции. Сборка мусора. Изменяемые и неизменяемые типы данных. *Значения и переменные.* Числа. Целые литералы. Литералы вещественных чисел. Дата и время. Логические значения. Строки. Значения null и undefined. Объявление переменных. Область видимости переменной.

Цель занятия. Освоить основные типы данных языка и правила их использования на конкретных примерах.

Задачи закрепления теоретических знаний и практических умений и навыков: студент должен знать основные типы данных языка и правила их использования.

Форма проведения: подготовка докладов для выступления на практическом занятии и коллективной дискуссии.

Учебно-методическая литература [1 – 17].

Тема 4. Операции и операторы.

Операции в Javascript. Приоритет выполнения операций, их назначение, примеры использования. *Встроенные математические функции, работа со строками и специальные значения.* Объект Math. Основные методы. Работа со строками. Специальные значения. Преобразования типов. Преобразование и равенство. Явные преобразования. *Операторы языка.* Оператор if. Оператор switch. Операторы while, do while. Оператор for. Примеры.

Цель занятия. Освоить структуру и средства исполнительской части языка и правила их использования на конкретных примерах.

Задачи закрепления теоретических знаний и практических умений и навыков: студент должен знать основные операции и операторы языка и правила их использования.

Форма проведения: подготовка докладов для выступления на практическом занятии и коллективной дискуссии.

Учебно-методическая литература [1 – 17].

4.3.3. Лабораторные занятия

Лабораторные работы по дисциплине «Web-программирование» для направления подготовки 09.03.04 – «Программная инженерия» имеют своей целью изучение технологии и основных приемов Web-программирования, освоение языка программирования JavaScript, приобретение навыков разработки и отладки Web-программ.

В результате выполнения лабораторного практикума студенты должны уметь:

- разрабатывать, отлаживать и выполнять программы на языке программирования JavaScript;
- применить свои знания к решению определенных предметных задач.

№ п/п	Наименование лабораторных работ	Трудоемкость (час.)	Формируемые компетенции	Форма контроля
1.	Создание статичного 4-х страничного сайта с использованием HTML и CSS	2	ПК-4, ПК-10	зачет
2.	Внедрение использования Javascript в сайт, созданный на предыдущей лабораторной работе	2	ПК-4, ПК-10	зачет
3.	Ввод данных через форму на HTML, проверка введенных данных на странице и табулирование сложных функций по нескольким переменным	2	ПК-4, ПК-10	зачет
4.	Генерация специальной таблицы на HTML при помощи Javascript, с заданной структурой ячеек	2	ПК-4, ПК-10	зачет
5.	Формирование и обработка элементов одномерных и двумерных массивов на Javascript	2	ПК-4, ПК-10	зачет
6.	Формирование и обработка массивов объектов, представляющих заданную предметную область	2	ПК-4, ПК-10	зачет
7.	Работа с функциями как с объектами. Мемоизация функций. Функции высших порядков	2	ПК-4, ПК-10	зачет
8.	Разработка классов, представляющих собой законченный функционал для решения определённой задачи	2	ПК-4, ПК-10	зачет

4.3.4. Самостоятельная работа

Самостоятельное изучение тем учебной дисциплины способствует закреплению знаний, умений и навыков, полученных в ходе аудиторных занятий, углублению и расширению знаний по отдельным вопросам и темам дисциплины; освоению умений практического использования полученных знаний по разработке программ различного прикладного характера.

Самостоятельная работа обучающихся по данному курсу заключается:

- при подготовке к лекциям и практическим занятиям в изучении и доработке конспекта лекции с применением учебно-методической литературы, подборе дополнительных примеров к теоретическим положениям курса по данной теме;
- при подготовке к лабораторным работам в разработке, отладке и выполнении программы своего варианта задания по данной теме;
- при самостоятельном изучении отдельных вопросов и тем курса с применением рекомендуемой учебно-методической литературы;
- при подготовке к зачету в изучении, осмыслении и повторении пройденного теоретического материала и выполненных практических заданий с применением конспекта лекций и учебно-методической литературы.

№ п/п	Тематика самостоятельной работы	Трудоемкость (час.)	Формируемые компетенции	Форма контроля
1.	Гипертекстовая информационная система и введение в Javascript. Появление технологии WEB. Появление языка Javascript	4	ПК-4, ПК-10	зачет
2.	Лексическая структура JavaScript	2	ПК-4, ПК-10	зачет
3.	Типы данных. Значения. Переменные	6	ПК-4, ПК-10	зачет
4.	Операции и операторы	6	ПК-4, ПК-10	зачет
5.	Массивы. Методы класса Array	8	ПК-4, ПК-10	зачет
6.	Объекты и их свойства	4	ПК-4, ПК-10	зачет
7.	Сериализация и клонирование объектов	4	ПК-4, ПК-10	зачет
8.	Функции. Базовое использование функций в языке Javascript	4	ПК-4, ПК-10	зачет
9.	Расширенное использование функций в языке Javascript	4	ПК-4, ПК-10	зачет
10.	Классы объектов в языке Javascript	4	ПК-4, ПК-10	зачет
11.	Применение функций-конструкторов в языке Javascript	5	ПК-4, ПК-10	зачет

5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине представлен в виде оценочных материалов и приведен в **Приложении** к рабочей программе дисциплины (см. документ «Оценочные материалы по дисциплине «Web-программирование»).

6. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Основная литература

1. Дейтел, Х.М. Как программировать для INTERNET&WWW: Пер. с англ. / Х.М. Дейтел, Дейтел П. Дж., Т.Р. Нието. — М.: БИНОМ, 2005. — 1184 с. (20 экз.)
2. Сапрыкин А.Н. Веб-программирование: Метод. ук. к лаб. раб. Ч.1 / А.Н. Сапрыкин, А.М. Гостин, РГРТУ. — Рязань, 2016. — 16 с. №5070 (40 экз.)
3. Сапрыкин А.Н. Веб-программирование: Метод. ук. к лаб. раб. Ч.2 / А.Н. Сапрыкин, А.М. Гостин, РГРТУ. — Рязань, 2017. — 16 с. №5137 (40 экз.)
4. Гостин А.М. Интернет-технологии: уч. пособие / А.М. Гостин, А.Н. Сапрыкин, РГРТУ. — Рязань, 2017. — 64 с. (39 экз.)

6.2. Дополнительная литература

5. Флэнаган Д. JavaScript. Подробное руководство. — СПб: Символ Плюс, 2012. — 1080 с.
6. Маккоу А. Веб-приложения на JavaScript. — СПб.: Питер, 2012. — 288 с.
7. Фримен Э., Робсон Э. Изучаем программирование на JavaScript. — СПб.: Питер, 2015. — 640 с.
8. Флэнаган Д. JavaScript: карманный справочник, 3-е изд. — ООО «И.Д. Вильямс», 2013. — 320 с.
9. Никсон Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5. 3-е изд. — СПб.: Питер, 2015. — 688 с.
10. Херман Д. Сила JavaScript. — СПб.: Питер, 2013. — 288 с.
11. Эспозито Д., Эспозито Ф. Разработка приложений для Windows8 на HTML5 и JavaScript. — СПб.: Питер, 2014. — 384 с.
12. Закас Н. JavaScript для профессиональных веб-разработчиков. — СПб.: Питер, 2015. — 960 с.
13. Моррисон М. Изучаем JavaScript. — СПб.: Питер, 2012. — 608 с.
14. Вахтуров В. В. JavaScript. Освой на примерах. — СПб.: БХВ-Петербург, 2007. — 400 с.
15. Прохоренок Н. А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера. — 3-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2010. — 912 с.
16. Закас Н. JavaScript. Оптимизация производительности. — СПб.: СимволПлюс, 2012. — 256 с.
17. Вейнер, П. Языки программирования Java и JavaScript: Пер. с англ. / Вейнер Питер. — М.: ЛОРИ, 1998. — 242 с.

6.3. Методические указания для обучающихся по освоению дисциплины

6.3.1. Методические указания к освоению лекционного материала

Для освоения лекционного материала следует: изучить конспект лекции в тот же день, после лекции: 10 – 15 минут, повторно прочитать конспект лекции за день перед следующей лекцией: 10 – 15 минут. Также следует изучить теоретический лекционный материал по рекомендуемому учебнику/учебному пособию: 1 час в неделю.

Следует максимально использовать лекционное время для изучения дисциплины, понимания лекционного материала и написания конспекта лекций. В процессе лекционного занятия студент должен уметь выделять важные моменты и основные положения. При написании *конспекта лекций* следует придерживаться следующих правил и рекомендаций.

1. При ведении конспекта рекомендуется структурировать материал по разделам, главам, темам. Вести нумерацию формул, схем, рисунков. Выделять по каждой теме постановку задачи, основные положения, выводы. Кратко записывать те пояснения лектора, которые показались особенно важными. Это позволит при подготовке к сдаче зачёта и экзамена не запутаться в структуре лекционного материала.

2. Лекционный материал следует записывать в конспект лишь после того, как излагаемый лектором тезис будет дослушан до конца и понят.

3. При конспектировании следует отмечать непонятные, на данном этапе, положения, доказательства и пр.

6.3.2. Методические указания к практическим/лабораторным занятиям

Практические занятия по дисциплине существенно дополняют лекции. В процессе анализа теоретических положений и решения практических задач студенты расширяют и углубляют свои знания, полученные из лекционного курса и учебников, приобретают умение применять общие закономерности к конкретным случаям. В процессе решения задач развивается логическое мышление, и вырабатываются навыки вычислений, работы со справочной литературой. Практические занятия способствуют закреплению знаний и практических навыков, формированию конструктивного стиля мышления, расширению кругозора.

При подготовке к практическому занятию необходимо внимательно ознакомиться с соответствующим теоретическим материалом по конспекту лекций и рекомендуемому учебнику, затем изучить конспект или материалы предыдущего практического занятия и выполнить заданное расчетное задание: 1 – 2 часа в неделю.

Следует максимально использовать аудиторное время практических занятий. В процессе занятия студент должен активно участвовать в дискуссиях, обсуждениях и решениях практических задач и вести *конспект практических занятий* отдельно от конспекта лекций.

Дополнительно в часы самостоятельной работы студенты могут повторно решить задачи, с которыми они плохо освоились во время аудиторных занятий, и обязательно те задачи, которые не получились дома при предыдущей подготовке к практическим занятиям.

Подготовка к лабораторным работам. Перед началом лабораторной работы необходимо ознакомиться с методическими указаниями к данной лабораторной работе, внимательно ознакомиться с заданием и желательно заранее выполнить подготовку программного проекта в используемой инструментальной среде, чтобы время лабораторного занятия использовать для исправления ошибок, модификации проекта и защиты данной работы.

Выполнение каждой из запланированных работ заканчивается предоставлением отчета. Требования к форме и содержанию отчета приведены в методических указаниях к лабораторным работам или определяются преподавателем на первом занятии.

Отчет по лабораторной работе студент должен начать оформлять еще на этапе подготовки к ее выполнению. Для допуска к лабораторной работе, студент должен представить преподавателю «заготовку» отчета, содержащую: оформленный титульный лист или название и номер работы при ведении общего конспекта, цель работы, задание, проект решения, и при наличии полученные результаты, выводы.

Изучение методических указаний к лабораторной работе – 2 часа перед выполнением лабораторной работы и в ходе разработки проекта и 2 часа для оформления отчета, отладки проекта и подготовки к сдаче работы.

После выполнения лабораторной работы необходимо согласовать полученные результаты с преподавателем. Важным этапом является *защита лабораторной работы*. В процессе защиты студент отвечает на вопросы преподавателя, касающиеся теоретического материала, относящегося к данной работе, и проекта, реализующего его задание, комментирует полученные в ходе работы результаты. При подготовке к защите лабораторной работы рекомендуется ознакомиться со списком вопросов по изучаемой теме и попытаться самостоятельно на них ответить, используя конспект лекций и рекомендуемую литературу. Кроме чтения учебной литературы рекомендуется активно использовать информационные ресурсы сети Интернет по изучаемой теме.

6.3.3. Методические указания к подготовке и сдаче теоретического зачета

Теоретический зачет – форма промежуточной проверки знаний, умений, навыков, степени освоения дисциплины. Главная задача зачета состоит в том, чтобы у студента по окончании изучения данной дисциплины сформировались определенное представление об

общем содержании дисциплины, определенные теоретические знания и практические навыки, определенный кругозор. Готовясь к зачету, студент приводит в систему знания, полученные на лекциях, на практических и лабораторных занятиях, разбирается в том, что осталось непонятным, и тогда изучаемая им дисциплина может быть воспринята в полном объеме с присущей ей строгостью и логичностью, ее практической направленностью.

Зачеты дают возможность преподавателю определить теоретические знания студента и его практические навыки при решении определенных прикладных задач. Оцениваются: понимание и степень усвоения теоретического материала; степень знакомства с основной и дополнительно литературой, а также с современными публикациями; умение применить теорию к практике, решать определенные практические задачи данной предметной области, правильно проводить расчеты и т. д.; знакомство с историей данной науки; логика, структура и стиль ответа, умение защищать выдвигаемые положения.

Значение зачетов не ограничивается проверкой знаний, являясь естественным завершением обучения студента по данной дисциплине, они способствуют обобщению и закреплению знаний и умений, приведению их в стройную систему, а также устранению возникших в процессе обучения пробелов.

Подготовка к зачету – это тщательное изучение и систематизация учебного материала, осмысление и запоминание теоретических положений, формулировок, формул, установление и осмысление внутри предметных связей между различными темами и разделами дисциплины, закрепление теоретических знаний путем решения определенных задач.

Перед зачетом преподаватель может назначить *консультацию*, ее цель – дать ответы на вопросы, возникшие в ходе самостоятельной подготовки студента, студент имеет возможность получить ответ на все неясные ему вопросы, кроме того, преподаватель будет отвечать на вопросы других студентов, что будет способствовать повторению и закреплению знаний всех присутствующих. Преподаватель на консультации, как правило, обращает внимание на те разделы, по которым на предыдущих зачетах ответы были неудовлетворительными, а также фиксирует внимание на наиболее трудных разделах или темах курса.

Планируя подготовку к сдаче зачета, обучаемый должен учитывать сразу несколько факторов: сложность учебного материала и степень его проработки в ходе обучения, свои индивидуальные способности. Подготовку к зачету следует начинать с общего планирования деятельности, с определения объема материала, подлежащего проработке, необходимо внимательно сверить конспекты с программой дисциплины, чтобы убедиться, все ли разделы отражены в лекциях, отсутствующие темы изучить по учебнику. Второй этап предусматривает системное изучение материала по предмету с обязательной записью всех выкладок, выводов, формул. На третьем этапе – этапе закрепления – полезно чередовать повторение особенно сложных вопросов с беглым повторением всего материала.

6.3.4. Рекомендации по работе с литературой

Теоретический материал курса становится более понятным, когда дополнительно к прослушиванию лекции и изучению конспекта изучаются и книги по данному предмету. Литературу по дисциплине рекомендуется читать как в бумажном, так и в электронном виде (если отсутствует бумажный аналог). Полезно использовать несколько учебников и пособий по дисциплине. Рекомендуется после изучения очередного параграфа ответить на несколько вопросов по данной теме. Полезно мысленно задать себе следующие вопросы (и попробовать ответить на них): «о чем этот параграф?», «какие новые понятия введены, каков их смысл?», «зачем мне это нужно по специальности?».

Рекомендуется самостоятельно изучать материал, который еще не прочитан на лекции и не применялся на лабораторном или практическом занятии, тогда занятия будут гораздо понятнее. В течение недели рекомендуется выбрать время (1 час) для работы с литературой.

6.3.5. Образовательные технологии, используемые в учебном процессе

Изучение дисциплины построено на методике *когнитивного или проблемного обучения*, как наиболее эффективной технологии при подготовке специалиста технического профиля. В рамках такого обучения студенты с самого начала учатся мыслить как программные инженеры, концентрировать внимание на решаемой проблеме, изучать требования и особенности проектирования перед созданием кода программы, работать по итеративной схеме, т.е. вырабатывают в себе привычку думать обо всех деталях, необходимых для разработки больших программ. При подготовке программных инженеров программирование остается основополагающим навыком, чем раньше и дольше студенты будут программировать, тем лучшими специалистами они смогут стать в дальнейшем, большинство возможностей их будущей работы требуют наличия профессиональных навыков программирования, поэтому необходимо делать основной упор на получение таких навыков. Применение данного подхода способствует тому, что студенты с самого начала чувствуют, что обучаются выбранной специальности.

Для обеспечения результативности профессиональной практики рекомендуется поддерживать рейтинговую система оценки знаний текущей успеваемости, выполнения лабораторных заданий. Активно поощряется профессионально выполненная работа с обязательным профессиональным оформлением отчетов.

Принятая технология обучения предусматривает применение активных форм обучения, таких как интерактивные методы чтения лекций и проведения практических и лабораторных занятий, организация устных презентаций в группе для конструктивного обсуждения и оценивания выполненных студенческих работ с целью демонстрации прогресса в обучении и для выработки у студентов лучшего понимания изучаемой проблематики, использование стандартизированных тестов для отслеживания и оценивания общего прогресса в группе и у каждого студента в отдельности. Чтение лекций и проведение практических и лабораторных занятий осуществляется с использованием мультимедийных средств и раздаточных материалов, с привлечением обучающихся и контролирующих учебных курсов.

7. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ ИНТЕРНЕТ, НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1. Электронно-библиотечная система «Лань». – Режим доступа: с любого компьютера РГРТУ без пароля. – URL: <https://e.lanbook.com/>
2. Электронно-библиотечная система «IPRbooks». – Режим доступа: с любого компьютера РГРТУ без пароля, из сети Интернет по паролю. – URL: <https://iprbookshop.ru/>.
3. Электронная библиотека РГРТУ. – URL: <http://weblib.rrtu.ebs>.
4. Научная электронная библиотека eLibrary. – URL: <http://e.lib/vlsu.ru/www.uisrussia.msu.ru/elibrary.ru>
5. Библиотека и форум по программированию. – URL: <http://www.cyberforum.ru>
6. Национальный открытый университет ИНТУИТ. – URL: <http://www.intuit.ru/>
7. Информационно-справочная система. – URL: <http://window.edu.ru>

8. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ, ВКЛЮЧАЯ ПЕРЕЧЕНЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ

1. Операционная система Windows XP (MicrosoftImagine, номерподписки 700102019, бессрочно).
2. Операционная система Windows XP (MicrosoftImagine, номерподписки ID 700565239, бессрочно).
3. OpenOffice (OpenOffice.org, OO.org, OO.o, OOo) – свободно распространяемое ПО.
4. MicrosoftOfficeVisio (MicrosoftImagine:Номер подписки 700102019, бессрочно).

5. Свободно распространяемое ПО: Microsoft Visual Studio Express (<https://www.visualstudio.com/ru/vs/express>), Qt Framework 5.10 Open Source LGPL license (<https://www.qt.io/>).

9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для освоения дисциплины необходимы следующие материально-технические ресурсы:

1) аудитория для проведения лекционных и практических занятий, групповых и индивидуальных консультаций, текущего контроля, промежуточной аттестации, оборудованная проектором и маркерной (меловой) доской;

2) классы, оснащенные персональными компьютерами для проведения лабораторных занятий;

3) аудитория для самостоятельной работы, оснащенная компьютерной техникой с подключением к локальной вычислительной сети и сети Интернет:

Наименование специальных помещений и помещений для самостоятельной работы	Оснащенность специальных помещений и помещений для самостоятельной работы	Перечень лицензированного ПО. Реквизиты подтверждающего документа
Учебная аудитория для проведения практических занятий, лабораторных работ и самостоятельной работы №206-3	Проектор: InFocus LP640 18 ПК с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду: ЦП: Intel Pentium 4 class 2800 ОЗУ: 1 Гб ПЗУ: 50 Гб (11 шт.) ЦП: Intel Pentium 4 class 3200 ОЗУ: 1 Гб ПЗУ: 50 Гб (5 шт.) ЦП: Intel Pentium 4 class 2800 ОЗУ: 500 Мб ПЗУ: 50 Гб (1 шт.) ЦП: Intel Pentium 4 class 2800 ОЗУ: 2 Гб ПЗУ: 50 Гб (1 шт.)	1. Операционная система Windows XP/Vista/7/8/10 (Microsoft Imagine: Номер подписки 700102019, бессрочно). 2. Microsoft Visual Studio (Microsoft Imagine: Номер подписки 700102019, бессрочно). 3. Microsoft Office Access (Microsoft Imagine: Номер подписки 700102019, бессрочно). 4. Microsoft Office Visio (Microsoft Imagine: Номер подписки 700102019, бессрочно). 5. Microsoft SQL Server (Microsoft Imagine: Номер подписки 700102019, бессрочно). 6. Microsoft Project (Microsoft Imagine: Номер подписки 700102019, бессрочно). 7. Свободно распространяемое программное обеспечение под лицензиями GNU, Apache, Oracle, Mozilla, CeCILL.
Учебная аудитория для проведения практических занятий, лабораторных работ и самостоятельной работы №206-4	18 ПК с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду: ЦП: Pentium 4 class 2800 ОЗУ: 1 Гб ПЗУ: 50 Гб (8 шт.) ЦП: Intel Pentium II/III class 2327 ОЗУ: 2 Гб ПЗУ: 50 Гб (10 шт.)	1. Операционная система Windows XP/Vista/7/8/10 (Microsoft Imagine: Номер подписки 700102019, бессрочно). 2. Microsoft Visual Studio (Microsoft Imagine: Номер подписки 700102019, бессрочно). 3. Microsoft Office Access (Microsoft Imagine: Номер подписки 700102019, бессрочно). 4. Microsoft Office Visio (Microsoft Imagine: Номер подписки 700102019, бессрочно).

		<p>5. Microsoft SQL Server (Microsoft Imagine: Номер подписки 700102019, бессрочно).</p> <p>6. Microsoft Project (Microsoft Imagine: Номер подписки 700102019, бессрочно).</p> <p>7. Свободно распространяемое программное обеспечение под лицензиями GNU, Apache, Oracle, Mozilla, CeCILL.</p>
<p>Учебная аудитория для проведения практической занятий, лабораторных работ и самостоятельной работы №206-5</p>	<p>24 ПК с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду: ЦП: Intel Pentium II/III class 2394 ОЗУ: 2 Гб ПЗУ: 70 Гб (17 шт.) ЦП: Intel Pentium II/III class 2327 ОЗУ: 2 Гб ПЗУ: 100 Гб (1 шт.) ЦП: Intel Pentium III Xeon 3093 ОЗУ: 4 Гб ПЗУ: 300 Гб (6 шт.)</p>	<p>1. Операционная система Windows XP/Vista/7/8/10 (Microsoft Imagine: Номер подписки 700102019, бессрочно).</p> <p>2. Microsoft Visual Studio (Microsoft Imagine: Номер подписки 700102019, бессрочно).</p> <p>3. Microsoft Office Access (Microsoft Imagine: Номер подписки 700102019, бессрочно).</p> <p>4. Microsoft Office Visio (Microsoft Imagine: Номер подписки 700102019, бессрочно).</p> <p>5. Microsoft SQL Server (Microsoft Imagine: Номер подписки 700102019, бессрочно).</p> <p>6. Microsoft Project (Microsoft Imagine: Номер подписки 700102019, бессрочно).</p> <p>7. Свободно распространяемое программное обеспечение под лицензиями GNU, Apache, Oracle, Mozilla, CeCILL.</p>