МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное автономное**

**образовательное учреждение высшего образования**

**«Национальный исследовательский Нижегородский государственный университет им. Н.И. Лобачевского»**

|  |
| --- |
| Институт информационных технологий, математики и механики |

|  |
| --- |
| УТВЕРЖДАЮ: |

|  |  |  |
| --- | --- | --- |
| Директор |  | В.П. Гергель |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| « |  | » |  |  | 2017 г. |

**Рабочая программа дисциплины**

|  |
| --- |
| **Интернет-программирование и разработка Web-сайтов** |

Уровень высшего образования

|  |
| --- |
| **Бакалавриат** |

Направление подготовки

|  |
| --- |
| **01.03.02 Прикладная математика и информатика** |

Направленность образовательной программы

|  |
| --- |
| **Прикладная математика и информатика (общий профиль)** |

Квалификация

|  |
| --- |
| **Бакалавр** |

Форма обучения

|  |
| --- |
| **очная** |

Нижний Новгород

2017

**1. Место и цели дисциплины в структуре ОПОП**

Данная дисциплина читается в пятом семестре бакалавриата (Б1.В.ДВ.05.03 – дисциплины по выбору). Дисциплина опирается на материалы курса языки программирования.

**Цель освоения дисциплины**

 Целями освоения дисциплины «Интернет-программирование и разработка Web-сайтов» являются обучение студентов технологиям разработки Web-сайтов и подходам к Internet программированию.

1. **Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников)**

|  |  |
| --- | --- |
| **Формируемые компетенции** | **Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций** |
| *Способность к коммуникации в устной и письменной формах на русском и иностранном языках для решения задач межличностного и межкультурного взаимодействия (ОК-5)**(Завершающий этап)* | ***ЗНАТЬ****З1 (ОК-5)* *типовую структуру отчета и требования к содержанию его основных разделов;****УМЕТЬ****У1 (ОК-5) логически верно, аргументированно и ясно строить устную и письменную речь;****ВЛАДЕТЬ****В1 (ОК-5) навыками разработки отчета о проделанной работе.* |
| *Способность к разработке и применению алгоритмических и программных решений в области* *системного и прикладного программного обеспечения (ПК-7)**(Базовый этап)* | ***ЗНАТЬ****З1 (ПК-7) основы разработки web-сайтов и Internet программирования;**З2 (ПК-7) основы проектирования сайтов и технологии проектирования;**З3 (ПК-7) основы программирования сайтов различными программными средствами.****УМЕТЬ****У1 (ПК-7) разрабатывать Web-сайты, используя технологии проектирования сайтов и Internet-программирования, и использовать их на практике.****ВЛАДЕТЬ*** *В1(ПК-7) навыками Internet программирования при разработке Web-сайтов.* |

1. **Структура и содержание дисциплины «Интернет-программирование**

 **и разработка Web-сайтов»**

Объем дисциплины составляет 3 зачетных единицы, всего 108 часов, из которых **50** часов составляет **контактная работа** обучающегося с преподавателем:

 16 часов занятия лекционного типа,

 32 часа занятия семинарского типа

2 часа промежуточной аттестации

На самостоятельную работу студента отводится 58 часов (в т.ч. 36 часов подготовки к экзамену)

Содержание дисциплины

|  |  |  |
| --- | --- | --- |
| **Наименование и краткое содержание разделов и тем дисциплины (модуля),** **форма промежуточной аттестации по дисциплине (модулю)** | **Всего****(часы)** | В том числе |
| **Контактная работа (работа во взаимодействии с преподавателем), часы**из них | **Самостоятельная работа обучающегося, часы** |
| **Занятия лекционного типа** | **Занятия семинарского типа** | **Занятия лабораторного типа** |  | **Всего** |
| Очная | Очно-заочная | Заочная | Очная | Очно-заочная | Заочная | Очная | Очно-заочная | Заочная | Очная | Очно-заочная | Заочная |  |  |  | Очная | Очно-заочная | Заочная | Очная | Очно-заочная | Заочная |
| Межсетевое взаимодействие в TCP/IP сетях | 14 |  |  | 2 |  |  | 4 |  |  |  |  |  |  |  |  | 6 |  |  | 8 |  |  |
| Протоколы прикладного уровня | 14 |  |  | 2 |  |  | 4 |  |  |  |  |  |  |  |  | 6 |  |  | 8 |  |  |
| Принципы работы web-сервера | 11 |  |  | 1 |  |  | 2 |  |  |  |  |  |  |  |  | 3 |  |  | 8 |  |  |
| Расширение функциональности Web-сервера | 11 |  |  | 1 |  |  | 2 |  |  |  |  |  |  |  |  | 3 |  |  | 8 |  |  |
| Client-side технологии как часть контента, интерпретируемая клиентским процессом | 11 |  |  | 2 |  |  | 4 |  |  |  |  |  |  |  |  | 6 |  |  | 8 |  |  |
| Server-side технологии | 27 |  |  | 6 |  |  | 12 |  |  |  |  |  |  |  |  | 18 |  |  | 9 |  |  |
| СУБД как составная часть Web приложения | 15 |  |  | 2 |  |  | 4 |  |  |  |  |  |  |  |  | 6 |  |  | 9 |  |  |
| В т.ч. текущий контроль | 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Промежуточная аттестация - Экзамен**  |

1. **Образовательные технологии**

 **Обучение данной дисциплине** организовано следующим образом: предусмотрено 2 вида занятий: лекционные и практические аудиторные. В рамках лекционных занятий рассматривается теоретический материал по тематике «интернет-программирование». В рамках практических аудиторных занятий формулируются задания для самостоятельного решения, обсуждаются возможные подходы к решению задач. Результаты самостоятельной работы студентов проверяются преподавателем.

1. **Учебно-методическое обеспечение самостоятельной работы обучающихся**
	1. **Виды самостоятельной работы студентов**

Выполнение практических работ на следующие темы:

* Web-проводник
* Chat с регистрацией
* Web-ICQ
* Web-mailer
* Календарь
* Форум с разделами
* Калькулятор
* Проверить правильность расстановки открывающихся и закрывающихся скобок
* Демонстрация работы (организации) стека
* Демонстрация работы (организации) очереди
* Сетевая игра «крестики-нолики»
* Игра «быки-коровы»
* Реализация склада
* Электронная записная книжка
* «Напоминатель» о днях рождения
* Универсальная система голосования
* Счетчик посещений сайта с выводом статистики
* Угадай число (больше-меньше)
* Игра «Реверси» для двух человек
* POP3 клиент
* Online-магазин с корзиной
* Тетрис
* Электронный школьный классный журнал
* Блокнот «кому что отдал»
* Перевод текста из русского в транслитерацию и обратно
* Ханойские башни
* Лотерея
* Игра «5 в ряд» для двух игроков по сети
* Каталогизатор дисков
* Перевод арифметического выражения из привычной формы записи в постфиксную
* Вычисление результата выражения, введенного в постфиксном формате
* Очиститель текста
* Web FTP-клиент
* Постраничный вывод некоторого списка
* Фото галлерея
* Получение курса валют
	1. **Образовательные материалы для самостоятельной работы студентов**
1. Кузнецова Л. Лекции по современным веб-технологиям. – ИНТУИТ. <http://www.intuit.ru/studies/courses/610/466/info>.
2. Брик С., Русак А., Сурин А. ,Храмцов П. Введение в HTML. . – ИНТУИТ. <http://www.intuit.ru/studies/courses/33/33/info>.
3. Гейн А. Web-программирование на PHP 5.2. – ИНТУИТ. <http://www.intuit.ru/studies/courses/985/308/info>.
4. Кан М. Основы программирования на JavaScript. – ИНТУИТ. http://www.intuit.ru/studies/courses/1093/132/info.
5. **Фонд оценочных средств для промежуточной аттестации по дисциплине**, включающий:
	1. Перечень компетенций выпускников образовательной программы с указанием результатов обучения (знаний, умений, владений), характеризующих этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования

*Оценка уровня формирования компетенции (ОК-5)*

|  |  |  |
| --- | --- | --- |
| Индикаторы компетенции | Критерии оценивания (дескрипторы) | Шкала оценивания |
| ***ЗНАТЬ****З1 (ОК-5)* *типовую структуру отчета и требования к содержанию его основных разделов;****УМЕТЬ****У1 (ОК-5) логически верно, аргументированно и ясно строить устную и письменную речь;****ВЛАДЕТЬ****В1 (ОК-5) навыками разработки отчета о проделанной работе.* | Отсутствие знаний материала, знания типовой структуры отчета о проделанной работе и сданных отчетов по практическим работам, полное отсутствие навыков, предусмотренных компетенцией. | Плохой уровень формирования компетенции. «Плохо» |
| Неудовлетворительный уровень понимания структуры и назначения отчета о проделанной работе, отчеты по практическим работам не сданы или сданы с грубыми ошибками, исключающими возможность их принятия (отсутствие ключевых разделов отчета, наличие грубых ошибок в описании постановки задачи и алгоритмов и др.). | Неудовлетворительный уровень формирования компетенции. «неудовлетворительно» |
| **Знать** З1 частично. **Уметь** У1 с погрешностями. **Владеть** некоторыми основными навыками подготовки отчета (В1), отчеты подготовлены со значительными погрешностями. | Удовлетворительный уровень формирования компетенции. «Удовлетворительно» |
| **Знать** и понимать З1 в целом.**Уметь** У1 с незначительными погрешностями. **Владеть** основными навыками, демонстрируя их в стандартных ситуациях. Отчеты по практическим работам сданы, но содержат определенные погрешности. | Хороший уровеньформирования компетенции. «Хорошо» |
| **Знать** и понимать З1. **Уметь** У1 с незначительными погрешностями. **Владеть** всеми основными навыками, демонстрируя при сдаче практическим работ и подготовке отчетов по ним. | Очень хороший уровеньформирования компетенции  «Очень хорошо» |
| **Знать** и понимать З1. **Уметь** У1 в полном объеме. **Владеть** всеми навыками, демонстрируя их при сдаче практических работ и отчетов по ним. | Отличный уровеньформирования компетенции  «Отлично» |
| **Знать** основной и дополнительный материал без ошибок и погрешностей. **Уметь** У1 в полном объеме. Свободно **Владеть** всеми навыками, демонстрируя их при сдаче практических работ и отчетов по ним. | Превосходный уровеньформирования компетенции  «Превосходно» |

*Оценка уровня формирования компетенции (ПК-7)*

|  |  |  |
| --- | --- | --- |
| Индикаторы компетенции | Критерии оценивания (дескрипторы) | Шкала оценивания |
| ***ЗНАТЬ****З1 (ПК-7) основы разработки web-сайтов и Internet программирования;**З2 (ПК-7) основы проектирования сайтов и технологии проектирования;**З3 (ПК-7) основы программирования сайтов различными программными средствами.****УМЕТЬ****У1 (ПК-7 разрабатывать Web-сайты, используя технологии проектирования сайтов и Internet-программирования, и использовать их на практике.****ВЛАДЕТЬ*** *В1(ПК-7) навыками Internet программирования при разработке Web-сайтов.* | Отсутствие знаний материала, отсутствует способность решения стандартных задач, полное отсутствие навыков, предусмотренных компетенцией. | Плохой уровень формирования компетенции. «Плохо» |
| Наличие грубых ошибок в основном материале, наличие грубых ошибок при решении стандартных задач, отсутствие навыков, предусмотренных данной компетенцией | Неудовлетворительный уровень формирования компетенции. «неудовлетворительно» |
| **Знать** некоторые основные понятия, изучаемые в рамках дисциплины (З1–З3). **Уметь** У1 с погрешностями. **Владеть** некоторыми основными навыками, демонстрируя их в стандартных ситуациях. | Удовлетворительный уровень формирования компетенции. «Удовлетворительно» |
| **Знать** большинство основных понятий, изучаемых в рамках дисциплины (З1–З3). **Уметь** У1 с незначительными погрешностями. **Владеть** основными навыками, демонстрируя их в стандартных ситуациях | Хороший уровеньформирования компетенции. «Хорошо» |
| **Знать** основные понятия, изучаемые в рамках дисциплины (З1–З3). **Уметь** У1 с незначительными погрешностями. **Владеть** всеми основными навыками, демонстрируя их в стандартных ситуациях, в том числе при решении дополнительных задач | Очень хороший уровеньформирования компетенции  «Очень хорошо» |
| **Знать** основной материал, предусмотренный компетенцией, без ошибок и погрешностей. **Уметь** У1 в полном объеме. **Владеть** всеми навыками, демонстрируя их не только в стандартных ситуациях, но и при решении нестандартных задач | Отличный уровеньформирования компетенции  «Отлично» |
| **Знать** основной и дополнительный материал, предусмотренный компетенцией, без ошибок и погрешностей. **Уметь** У1 в полном объеме. **Владеть** всеми навыками, демонстрируя их не только в стандартных ситуациях, но и при решении нестандартных задач | Превосходный уровеньформирования компетенции  «Превосходно» |

* 1. Описание шкал оценивания

Для оценивания результатов учебной деятельности студентов при изучении дисциплины «Интернет-программирование и разработка Web-сайтов» используется следующая шкала оценивания:

|  |  |
| --- | --- |
| **Оценка** | **Уровень подготовки** |
| Превосходно | На экзамене обучаемый показал высокий уровень подготовки, безупречное владение теоретическим материалом, знания, выходящие за рамки рассмотренного в курсе материала, творческий поход к разрешению нестандартных ситуаций. Имеет полностью и творчески выполненные самостоятельные задания на протяжении семестра. Демонстрирует способность решать дополнительные предложенные задачи, требующие оригинальности мышления.Освоение материала на 100%. |
| Отлично | Обучаемый показал высокий уровень владения материалом курса. Имеет выполненные самостоятельные задания. Демонстрирует способность решать дополнительные предложенные задачи, требующие оригинальности мышления.Освоение материала на уровне 90-99%. |
| Очень хорошо | В целом – весьма хорошая подготовка. Обучаемый дает ответы на все теоретические вопросы билета, но с рядом ошибок и неточностей; может решать задачи из всех основных разделов, имеет выполненные самостоятельные задания. Освоение материала на уровне 80-90%. |
| Хорошо | Достаточно хорошая подготовка, но с заметными ошибками или недочетами; получен полный ответ на все теоретические вопросы билета, но с рядом ошибок. Практические задания обучаемый выполняет, но с недочетами, самостоятельные задания, в основном, выполнены.Освоение материала на уровне 60-79%. |
| Удовлетворительно | Минимально достаточный уровень подготовки. Обучаемый в значительной части отвечает на все вопросы билета, но с множеством ошибок, не носящих грубого характера; имеет задолженности или низкую оценку по самостоятельным заданиям; предложенные практические задания выполняет с заметными ошибками. Освоение материала на уровне 35-59%. |
| Неудовлетворительно | Подготовка не достаточна и требует дополнительного изучения материала. Студент дает ошибочные ответы, как на теоретические вопросы билета, так и на наводящие и дополнительные вопросы экзаменатора. Как правило, имеет задолженности по самостоятельным заданиям и не может правильно решить предложенные практические задачи. Освоение материала порядка 15-34%. |
| Плохо | Подготовка абсолютно недостаточна. Обучаемый не отвечает на поставленные вопросы, не понимает терминологию; имеет задолженности по самостоятельным заданиям, не знает подходов к решению практических задач.Освоение материала ниже 15%. |

* 1. Критерии и процедуры оценивания результатов обучения по дисциплине, характеризующих этапы формирования компетенций

Для оценивания результатов обучения в виде знаний используются устные ответы на вопросы.

Для оценивания результатов обучения в виде умений и владений используются следующие процедуры и технологии:

–выполнение студентами практических (самостоятельных) работ, включающих постановку одной достаточно сложной учебной задачи в виде краткой формулировки действий, которые следует выполнить, и описания результата, который нужно получить.

– решение студентами экзаменационных задач.

* 1. Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения, характеризующих этапы формирования компетенций и (или) для итогового контроля сформированности компетенции.

***Примеры контрольных вопросов для оценивания результатов формирования ОК-5, ПК-7.***

1. Глобальные компьютерные сети: основные понятия, принципы функционирования. Каталоги ресурсов. Поисковые системы.
2. Язык гипертекстовой разметки страниц HTML: общая структура документа, абзацы, цвета, ссылки.
3. Язык гипертекстовой разметки страниц HTML: списки, графика (графические форматы, графический объект как ссылка), таблицы, фреймы.
4. Общие подходы к дизайну сайта. Разработка макета страницы.
5. Язык гипертекстовой разметки страниц HTML: формы.
6. Хостинг. Бесплатный хостинг. FTP. Размещение Интернет-ресурса на сервере провайдера. Регистрация Интернет-ресурса в каталогах и поисковых системах.
7. Преимущества и ограничения программ, работающих на стороне клиента. Язык JavaScript: основы синтаксиса.
8. Объектная модель HTML страницы.
9. Событийная модель DHTML: связывание событий с кодом, всплытие событий, объект Event.
10. Применение DHTML.
11. Введение в программирование на стороне сервера на примере PHP. Принцип работы.
12. Синтаксис языка программирования PHP.
13. Переменные. Константы. Операторы в PHP. Циклы. Массивы. Работа со строками.
14. Функции в PHP. Встроенные функции.
15. Работа с датой и временем в PHP.
16. Методы передачи параметров между страницами (GET, POST). Обработка действий пользователя при помощи форм.
17. Принципы хранения информации в базах данных MySQL. Архитектура базы данных MySQL (таблицы, связи, триггеры).
18. Подключение к базе данных из PHP файла. Вывод данных на PHP-страницу, попавших в выборку по SQL запросу. Передача параметров в запрос.
19. Принципы проектирования страниц. Разделение информации по таблицам в базе данных. Вывод группы данных, сортировка данных.
20. Создание HTML-страниц средствами PHP.

***Примеры практических (самостоятельных) задач:***

1. Создайте страницу html, содержащую функцию buildTable, которая, принимая двумерный массив элементов добавляет на страницу таблицу html.

Таблица должна содержать заголовок, в качестве которого используется первая строка массива. Задание должно быть выполнено с использованием работы с DOM текущей страницы.

2. Создайте страницу html, содержащую функцию buildTable, которая, принимая массив объектов с одинаковыми свойствами, которая строит структуру DOM, представляющую таблицу. У таблицы должна быть строка с заголовками, где имена свойств обёрнуты в элементы <th>, и должно быть по одной строчке на объект из массива, где его свойства обёрнуты в элементы <td>. Здесь пригодится функция Object.keys, возвращающая массив, содержащий имена свойств объекта.

3. В промежутке с 1928 по 2013 год турецкие законы запрещали использование букв Q, W и X в официальных документах. Запрограммируйте поле для ввода текста так, чтобы эти буквы нельзя было туда вписать.

4. Создайте страницу HTML, на которой при движении по ней мышью, следом за мышью движется произвольное изображение.

5. Создайте страницу HTML, содержащую четыре кнопки «создать таблицу», “удалить таблицу”, “добавить строку”, «добавить столбец». При нажатии на кнопку «создать таблицу» на странице отображается таблица с одной строкой и одним столбцом. При нажатии кнопок добавить строку или добавить столбец в таблицу добавляется строка или столбец соответственно. При нажатии кнопки «удалить таблицу» вся таблица удаляется.

6. Создайте страницу, содержащую тестовое поле с возможностью ввести несколько строк текста и кнопку. В текстовое поле можно ввести код на языке JavaScript, а при нажатии на кнопку это код должен выполняться.

7. Найдите в интернете изображения с игральными картами. Выберите одну любую масть и создайте страницу HTML со следующим функционалом. При старте страницы отображается колода карт, лежащих друг на друге рубашками вверх (все карты одной масти). При нажатии мышью на рубашке верхней карты карта «открывается» и откладывается вправо, уже рубашкой вниз. Нажав мышью 9 раз мы откроем всю колоду. Начальный порядок карт – случайный.

8. Создайте страницу, содержащую текстовое поле. Создайте скрипт, который при вводе текста в текстовое поле под текстовым полем отображает список вариантов автозавершения. У вас есть заданный в тексте программы массив возможных вариантов, и показывать нужно те из них, которые начинаются с вводимого текста. Когда пользователь щёлкает по предложенному варианту, он меняет содержимое поля на него.

9. Создайте страницу, содержащую элемент canvas. С этим элементом должна быть связаны функция «рисования», работающая следующим образом. Если нажать клавишу мыши на элементе, затем продвинуть мышь в сторону и отпустить, должна быть нарисована линия с началом в первой точке (точке нажатия клавиши мыши) и окончания во второй точке (точке, где клавиша мыши была отпущена).

10. Создайте страницу, содержащую 4 поля редактирования: одно поля типа текст, одно поле для ввода чисел, одно поле для выбора даты и одно поле для выбора цвета. При нажатии кнопки «добавить», значения, введенные пользователем в эти поля, должны быть добавлены в первую строку таблицы, расположенной на этой же странице. Таким образом, каждый раз добавляя записи в таблицу, мы как бы «сдвигаем» строки таблицы вниз раз за разом.

11. Создайте страницу следующей структуры: на ней должны быть три «чекбокса» помеченные метками «фрукты», «овощи», «ягоды». В программе должны быть определены три массива, содержащие по меньшей мере по пять элементов для названий фруктов, овощей и ягод. Также на странице должен быть выпадающий список, содержимое которого должно зависеть от выбранных элементов. При выборе фруктов в нем должны быть названия фруктов, при выборе и фруктов и овощей – названия и фруктов и овощей и т.д.

12. Создайте страницу, содержащую функцию, реализованную на JavaScript со следующим функционалом. Функция принимает массив строк в качестве аргумента, выделяет на странице прямоугольные области одинакового размера (размер областей одинаков и задается как константа) и размещает элементы массива в этих областях. Т.е. 0-й элемент массива в первой области, 1-й во второй и т.д. Указанные элементы должны полностью по ширине убираться в ширину окна браузера, не приводя к появлению горизонтального скроллера.

13. Создайте страницу, содержащую форму, включающую 4 текстовых поля. Под формой расположите два переключателя с подписями «горизонтально» и «вертикально». При выборе переключателя «горизонтально» все элементы формы должны выстроиться в одну линию, горизонтально. При этом возможно появление горизонтального скроллера в браузере. При выборе переключателя «вертикально», все элементы формы должны выстроиться друг под другом, с выравниванием по левому краю относительно формы.

14. Создайте страницу, которая при загрузке вызывает функцию, написанную на JavaScript. Функция принимает массив строк, каждая строка массива может содержать значения “text”, “number”,”date”или”color”. Функция должна сформировать на странице форму, включающую элементы управления, заданные элементами массива. Т.е. для элемента text в форму должно быть включено текстовое поле, для элемента number – поле для ввода чисел и т.д.

15. Создайте страницу, содержащую два поля ввода и две кнопки. Первое поле ввода служит для ввода номера элемента списка, после которого нужно осуществлять вставку. Второе поле ввода служит для ввода текста, который нужно вставлять. С помощью первой кнопки текст можно вставить на этом же уровне, что и элемент, после которого мы его вставляем. С помощью второй кнопки можно вставить текст как вложенный список (его уровень будет на один уровень ниже). Таким образом, можно выбрать элемент списка, ввести текст, который будет вставлен после этого элемента и выбрать, будет ли вставленный текст вставлен на этом же уровне, или же уровнем ниже. Использовать нумерованный список (элемент <ol>).

16. Создайте страницу, содержащую пустую форму (не содержащую элементов), поле редактирования (для задания значение метки) и четыре кнопки. Каждая кнопка добавляет в форму один элемент. Первая кнопка – тестовое поле, вторая кнопка – кнопку, третья – метку (label), причем метка должна быть привязана к последнему добавленному до нее элементу управления, четвертая-поле для ввода даты. В случае добавления метки, текстовое значение для нее берется из поля редактирования.

17. Создайте страницу, содержащую таблицу (элемент <table>). При нажатии на значение заголовка столбца таблицы, строки таблицы должны сортироваться по этому столбцу. При нажатии на любую строку таблицы (кроме строки заголовка) должен выводиться вопрос «удалить строку таблицы?» и при нажатии кнопки Ok, строка таблицы – удаляться.

18. Создайте страницу, содержащую элемент canvas. С этим элементом должна быть связаны функция «рисования» квадратов, работающая следующим образом. Если нажать клавишу мыши на элементе, затем продвинуть мышь в сторону и отпустить, должен быть нарисован квадрат с центром в первой точке (точке нажатия клавиши мыши) и углом во второй точке (точке, где клавиша мыши была отпущена). Предполагается, что первая и вторая точки не находятся на одной горизонтали или вертикали.

19. Создайте страницу, содержащую произвольный ненумерованный список (<ul>). Список может включать в себя вложенные списки. Уровень вложенности и количество элементов в списке – произвольные. Реализуйте функцию на JavaScript, которая принимая в качестве аргумента корневой элемент списка (<ul>) сортирует в лексикографическом порядке все элементы списка, включая и вложенные списки. Результатом работы функции должно быть изменение первоначального списка. Функция должна запускаться при щелчке мышью на любом элементе списка.

20. Напишите функцию на JavaScript, работающую следующим образом. Она принимает в качестве аргумента элемент, являющийся <div>, этот элемент определен на текущей странице и содержит произвольное изображение. При приближении курсора мыши к этому элементу, элемент пытается «убежать» от мыши, перескакивая в произвольном направлении.

21. Напишите функцию на JavaScript, принимающую элемент, являющийся таблицей (<table>) и массив, содержащий номера столбцов таблицы. Для каждого из столбцов, содержащихся в массиве:

a. Если столбец таблицы с этим номером содержит числовые данные – подсчитать сумму всех элементов этого столбца

b. Если столбец таблицы содержит хотя бы одно значение, не являющееся числом – подсчитать количество НЕПУСТЫХ значений в этом столбце

В конец таблицы добавить строку, в которой для каждого из столбцов, переданных в функцию вывести либо сумму по столбцу (для числовых данных), либо количество не пустых элементов.

22. Напишите функцию на JavaScript, принимающую в качестве аргумента элемент html и выводящую на страницу «дерево» DOM, для которого переданный в качестве аргумента элемент является корневым. Дерево должно выводиться в виде ненумерованного списка (<ul>) с вложениями. В качестве текста, выводимого в элементе списка использовать имя элемента (т.е. head, body, p, div и др) и id элемента в том случае, если он указан. Вызов функции на странице осуществлять при нажатии кнопки, элемент, начиная с которого выводить дерево, указывать через его id, которое вводить в текстовом поле.

***Образец экзаменационных билетов на оценивание формирования компетенций ОК-5, ПК-7***

Национальный исследовательский Нижегородский государственный университет

им. Н.И. Лобачевского

Институт ИТММ

Кафедра математического обеспечения суперкомпьютерных технологий

Дисциплина Интернет-программирование и разработка Web-сайтов

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

1. Составные элементы HTML-документа. Типы данных HTML. Структура HTML-документа.
2. PHP. Работа с переменными. Область видимости переменной. Неопределенные и неинициализированные переменные

Зав. кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Экзаменатор\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* 1. Методические материалы, определяющие процедуры оценивания

Положение о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся в ННГУ от 13.02.2014. <http://www.unn.ru/site/images/docs/obrazov-org/Formi_stroki_kontrolya_13.02.2014.pdf>

Положение о фонде оценочных средств, утвержденное приказом ректора ННГУ от 10.06.2015 №247-ОД.

1. **Учебно-методическое и информационное обеспечение дисциплины**

а) основная литература:

1. Кузнецова Л. Лекции по современным веб-технологиям. – ИНТУИТ. <http://www.intuit.ru/studies/courses/610/466/info>.
2. Брик С., Русак А., Сурин А. ,Храмцов П. Введение в HTML. . – ИНТУИТ. <http://www.intuit.ru/studies/courses/33/33/info>.
3. Гейн А. Web-программирование на PHP 5.2. – ИНТУИТ. <http://www.intuit.ru/studies/courses/985/308/info>.

б) дополнительная литература:

1. Кан М. Основы программирования на JavaScript. – ИНТУИТ. <http://www.intuit.ru/studies/courses/1093/132/info>.

в) программное обеспечение и Интернет-ресурсы:

Операционные системы семейства MicrosoftWindows, лицензия по подписке MicrosoftImagine.

Среда разработки семейства MicrosoftVisualStudio, лицензия по подписке MicrosoftImagine.

Microsoft Office (лицензия)

1. **Материально-техническое обеспечение дисциплины**

Учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, помещения для самостоятельной работы обучающихся, оснащенные компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду ННГУ. Наличие рекомендованной литературы. Используемое лицензионные программное обеспечение:

* Операционные системы семейства MicrosoftWindows, лицензия по подписке MicrosoftImagine.
* Среда разработки семейства MicrosoftVisualStudio, лицензия по подписке MicrosoftImagine.
* Microsoft Office, – лицензионное ПО приобретено в 2006/2007 гг при выполнении нац. проекта «Образование», ключ у системного администратора.

Программа составлена в соответствии с требованиями ФГОС по направлению подготовки 01.03.02 Прикладная математика и информатика

Авторы: \_\_\_\_\_\_\_\_\_\_\_\_\_Мееров И.Б.

 \_\_\_\_\_\_\_\_\_\_\_\_ Лабутин Д.Ю.

Рецензент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Заведующий кафедрой МОСТ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Р.Г. Стронгин

Программа одобрена на заседании методической комиссии Института информационных технологий, математики и механики ННГУ им. Н.И. Лобачевского

от 29 августа 2017 года, протокол № 20