

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ

ОШСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

КАФЕДРА АСЦТ

«Согласована»
Председатель Методического совета
факультета МИТ
Доцент, канд. п.и.к.: _____ Д. Зулпукарова

«__» _____ 2020 г.

«Утверждена»
на заседании кафедры Программирования
от 27 августа 2020 года, протокол №1
Зав. каф. АСЦТ, доцент

Молдоярв У.Д. _____

УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС

**по дисциплине «Программирование Web-
ориентированных приложений»**

для магистрантов очного отделения, обучающихся по специальности:
710100 «Информатика и вычислительная техника»

Составитель:

доцент Молдоярв У.Д.

2020-2021 – учебный год

Данные о преподавателе:

Молдояров Уларбек Дуйшобекович – доцент кафедры АСЦТ факультета Математика и информационных технологий ОшГУ

Стаж работы – 15 лет.

Образование:

- Высшее, ОшГУ, физико-математический факультет, 2005 г.;
- Магистр, ОшГУ, факультет математики и информационных технологий, 2016 г.

Рабочее место: 723500. главный корпус ОшГУ, ул. Ленина 331, кабинет – 205.

Контактная информация:

Моб. телефон: 0772843108, 0554490100

E-mail: ular_osh@list.ru

Дежурство преподавателя проводится в кабинете 205 по пятницам с 14.00 до 16.00

Оглавление

| | |
|---|----|
| I. АННОТАЦИЯ К ДИСЦИПЛИНЕ “УПРАВЛЕНИЕ ДАННЫМИ” | 4 |
| Внутренняя и внешняя рецензии на рабочую программу..... | 6 |
| II. РАБОЧАЯ ПРОГРАММА | 8 |
| 1. Цели освоения дисциплины | 10 |
| 2. Результаты обучения и компетенции, формируемые в процессе изучения дисциплины "Программирование Web-ориентированных приложений" | 10 |
| 4. Карта компетенций дисциплины в разрезе тем (разделов) | 11 |
| 5. Технологическая карта дисциплины | 12 |
| 6. Карта накапливаемости баллов по дисциплине | 12 |
| 7. Тематический план распределения часов по видам занятий | 14 |
| 8. Программа дисциплины | 16 |
| 9. Календарно-тематический план по видам занятий..... | 16 |
| 10. Разработка проекта..... | 19 |
| 11. Образовательные технологии..... | 19 |
| 12. Учебно-методическое и информационное обеспечение дисциплины | 20 |
| 13. Политика выставления баллов | 20 |
| III. СИЛЛАБУС | 24 |
| 1. Цели освоения дисциплины | 25 |
| 2. Результаты обучения и компетенции, формируемые в процессе изучения дисциплины "Программирование Web-ориентированных приложений" | 26 |
| 5. Технологическая карта дисциплины | 27 |
| 6. Карта накапливаемости баллов по дисциплине | 27 |
| 7. Тематический план распределения часов по видам занятий | 29 |
| 8. Программа дисциплины | 29 |
| 9. Календарно-тематический план по видам занятий..... | 30 |
| 10. Разработка проекта..... | 32 |
| 11. Образовательные технологии..... | 33 |
| 12. Учебно-методическое и информационное обеспечение дисциплины | 33 |
| 13. Политика выставления баллов | 34 |
| 14. Политика курса..... | 36 |
| IV. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ | 38 |
| 1. Общие положения | 39 |
| 2. Перечень компетенций и этапы их формирования..... | 39 |
| 3. Критерии оценки на экзамене | 49 |
| V. ГЛОССАРИЙ (СЛОВАРЬ) | 50 |
| VI. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ МАГИСТРАНТОВ ПО ИЗУЧЕНИЮ ДИСЦИПЛИНЫ | 55 |
| VII. ЛЕКЦИОННЫЕ МАТЕРИАЛЫ | 57 |
| VIII. ЛАБОРАТОРНЫЕ МАТЕРИАЛЫ | 67 |

I. АННОТАЦИЯ К ДИСЦИПЛИНЕ «УПРАВЛЕНИЕ ДАННЫМИ»

Дисциплина «Программирование Web-ориентированных приложений» призвана содействовать знакомству магистрантов с компьютерными телекоммуникациями и возможными подходами к разработке гипертекстовых документов, предназначенных для публикации в глобальной компьютерной сети Internet. Она важна с той точки зрения, что позволяет развивать способности магистрантов, связанные с общей культурой работы в глобальной сети. Курс закрепляет навыки работы с текстом и графикой, а также навыков программирования и проектирования и разработки информационных систем, являясь, таким образом, прямым продолжением курсов «Web программирование», «Разработка клиент-серверных прилож», «Проектирование АСОИ», «Базы данных», «Программирование» и многих других.

Целью курса является освоение практических приемов Web-конструирования и Web-программирования.

Основные задачи курса:

- закрепление знакомства с принципами функционирования глобальной компьютерной сети Internet, общими подходами к поиску и отбору информации в сети;
- обучение разработке Web-страниц на основе комплексного подхода;
- обучение программированию в Internet на стороне клиента и сервера;
- обучение использованию баз данных при разработке Web-проектов;
- обучение способам маркетинга в Internet, рекламы и продвижения разработанных Internet-ресурсов.

В лекционной части курса рассматриваются общие принципы Web-конструирования. Изучение всех тем сопровождается иллюстрирующими примерами.

Лабораторные работы в компьютерных классах служат для индивидуальной работы магистрантов над учебными задачами и итоговым проектом с целью выработки и закрепления практических навыков Web-конструирования и Web-программирования.

Результаты обучения дисциплины основывается на РО-8 в ООП:

При работе в сфере интернет магистрант использовать современные информационные технологии, предусматривающие организацию баз и банков данных, построение схем документооборота и т.д.

Формирующие компетенции:

- способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности; (ОК-2)
- способен к постановке и модернизации отдельных лабораторных работ и практикумов по дисциплинам направления; (ПК-6)
- способен проектировать и применять инструментальные средства реализации программно-аппаратных проектов; (ПК-13)

В процессе обучения дисциплины магистрант получает следующие **результаты обучения:**

- **знать** основные определения и понятия Web-конструирования и Web-программирования, основные приемы создания и продвижения сайтов;
- **уметь** разрабатывать и продвигать проблемно-ориентированные Web-ресурсы;
- **освоить** методы проектирования, разработки и маркетинга проблемно-ориентированных Web-ресурсов;

- **приобрести навыки** проектирования, разработки и маркетинга проблемно-ориентированных Web-ресурсов;
- **иметь представление** о проблемах, тенденциях и перспективах развития Web-конструирования и Web-программирования.

Внутренняя и внешняя рецензии на рабочую программу

РЕЦЕНЗИЯ

на рабочую программу по дисциплине "Программирование Web-ориентированных приложений" для магистрантов очного отделения, обучающихся по направлению 710100 «Информатика и вычислительная техника»

Рабочая программа дисциплины "Программирование Web-ориентированных приложений" разработана в соответствии с требованиями ГОС ВПО КР, на основе бюллетеня №19 ОшГУ и ООП специальности по направлению 710100 «Информатика и вычислительная техника».

Рецензируемая программа предназначена для методического обеспечения учебной работы магистрантов очной формы обучения. Содержание представленной на рецензию рабочей учебной программы включает в себя следующие разделы: цели и задачи освоения дисциплины; место дисциплины в структуре ООП; требования к результатам освоения дисциплины; объем дисциплины и виды учебной работы; содержание дисциплины; библиотечно-информационные ресурсы; оценочные средства; материально-техническое обеспечение.

Содержание курса представлено шести разделами, которые в полной мере отражают необходимый объем изучаемого материала. По каждому разделу составлен перечень вопросов, рассмотрение которых позволит сформировать знания, умения и навыки, отвечающие требованиям ВПО КР.

Информация о видах и объеме учебной работы содержит перечень лабораторных работ и тематику лекционных занятий, призванных помочь магистранту получить теоретические знания и практические навыки создания, внедрения, функционирования, применения системы управления базами данных и информационных систем управления, обеспечивающих поддержку работы специалиста.

Программа соответствует всем требованиям бюллетеня №19 ОшГУ и ООП специальности по направлению 710100 «Информатика и вычислительная техника».

Рецензент, старший специалист отдела
сетевого и технического сопровождения
ОАО Коммерческий банк Кыргызстан

Асранов Б.Б.

РЕЦЕНЗИЯ

на рабочую программу по дисциплине "Программирование Web-ориентированных приложений" для магистрантов очного отделения, обучающихся по направлению 710100 «Информатика и вычислительная техника»

Рабочая программа дисциплины "Программирование Web-ориентированных приложений" разработана в соответствии с требованиями ВПО КР, на основе бюллетеня №19 ОшГУ и ООП специальности по направлению 710100 «Информатика и вычислительная техника».

Рабочая программа содержит: цели дисциплины, результаты обучения и компетенции, формируемые в процессе изучения дисциплины, место дисциплины в структуре ООП, карта компетенций дисциплины в разрезе тем, технологическая карта дисциплины, карта накопления баллов по дисциплине, тематический план распределения часов по видам занятий, программа дисциплины, цели и результаты обучения по темам (разделам) дисциплины, календарно-тематический план по видам занятий с указанием формируемых компетенций, часов, баллов, методов, оценочных средств, литературы, образовательные, технологии учебно-методическое обеспечение дисциплины и политика выставления баллов.

На основании вышеизложенного считаю, что рецензируемая программа полностью соответствует требованиям, предъявляемым к рабочей программе по дисциплине "Программирование Web-ориентированных приложений" и может быть рекомендована для обучения магистрантов по направлению 710100 «Информатика и вычислительная техника».

Рецензент, доцент
кафедры ПИИБ факультета МИТ:

Эркебаев У.

II. РАБОЧАЯ ПРОГРАММА

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ

ОШСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

КАФЕДРА АСЦТ

«Согласована»
Председатель Методического совета
факультета МИТ доцент, канд.
п.и.к.: _____ Д. Зулпукарова

«Утверждена»
на заседании кафедры АСЦТ
от 27 августа 2020 года,
протокол №1 Зав. каф. АСЦТ, доцент

«__» _____ 2020 г.

Молдоярлов У.Д. _____

РАБОЧАЯ ПРОГРАММА

по дисциплине: "Программирование Web-ориентированных приложений" для магистрантов очного отделения, обучающихся по направлению: 710100 «Информатика и вычислительная техника»

Сетка часов по учебному плану

| Программирование Web-ориентированных приложений | Всего | Ауд. зан. | Аудит.зан. | | СРМ | Отчетность | |
|---|----------------|--------------|------------|--------|-------|------------|-------|
| | | | Лекции | Лабор. | | 2 сем | 2 сем |
| 1 курс, 2 сем. | 90ч (3 кр.) | 36 ч. | 20 ч. | 16 ч. | 56 ч. | РК-2 | Экз |

Рабочая программа составлена на основании ООП, утвержденной Ученым Советом факультета _____ протокол №__ от _____ 20__ г.

Составитель: доцент кафедры _____ Молдоярлов У. Д.

2020-2021 – учебный год

СОДЕРЖАНИЕ

| | |
|---|----|
| 1. Цели освоения дисциплины | 10 |
| 2. Результаты обучения и компетенции, формируемые в процессе изучения дисциплины "Программирование Web-ориентированных приложений" | 10 |
| 3. Место дисциплины в структуре ООП | 11 |
| 4. Карта компетенций дисциплины в разрезе тем (разделов)..... | 11 |
| 5. Технологическая карта дисциплины | 12 |
| 6. Карта накапливаемости баллов по дисциплине | 12 |
| 7. Тематический план распределения часов по видам занятий | 14 |
| 8. Программа дисциплины | 16 |
| 9. Цели и результаты обучения по темам дисциплины Ошибка! Закладка не определена. | |
| 10. Календарно-тематический план по видам занятий..... | 16 |
| 11. Образовательные технологии | 19 |
| 12. Учебно-методическое и информационное обеспечение дисциплины..... | 20 |
| 13. Политика выставления баллов..... | 20 |

1. Цели освоения дисциплины

Дисциплина «Программирование Web-ориентированных приложений» призвана содействовать знакомству магистрантов с компьютерными телекоммуникациями и возможными подходами к разработке гипертекстовых документов, предназначенных для публикации в глобальной компьютерной сети Internet. Она важна с той точки зрения, что позволяет развивать способности магистрантов, связанные с общей культурой работы в глобальной сети. Курс закрепляет навыки работы с текстом и графикой, а также навыков программирования и проектирования и разработки информационных систем, являясь, таким образом, прямым продолжением курсов «Web программирование», «Разработка клиент-серверных прилож», «Проектирование АСОИ», «Базы данных», «Программирование» и многих других.

Целью курса является освоение практических приемов Web-конструирования и Web-программирования.

Основные задачи курса:

- закрепление знакомства с принципами функционирования глобальной компьютерной сети Internet, общими подходами к поиску и отбору информации в сети;
- обучение разработке Web-страниц на основе комплексного подхода;
- обучение программированию в Internet на стороне клиента и сервера;
- обучение использованию баз данных при разработке Web-проектов;
- обучение способам маркетинга в Internet, рекламы и продвижения разработанных Internet-ресурсов.

В лекционной части курса рассматриваются общие принципы Web-конструирования. Изучение всех тем сопровождается иллюстрирующими примерами.

Лабораторные работы в компьютерных классах служат для индивидуальной работы магистрантов над учебными задачами и итоговым проектом с целью выработки и закрепления практических навыков Web-конструирования и Web-программирования.

2. Результаты обучения и компетенции, формируемые в процессе изучения дисциплины "Программирование Web-ориентированных приложений"

1. результате изучения дисциплины магистрант достигнет следующих **результатов обучения (РОд)**, соответствующих ожидаемым **результатам освоения образовательной программы (РОоп)** и заданным для дисциплины **компетенциям**:

| Код РОоп и его формулировка | Код компетенции ООП и его формулировка | Код РО дисциплины (РОд) и его формулировка |
|---|---|---|
| РО-8 Веб-активность: умеет создавать веб-приложения, создавать базы данных и работать с системами управления базами данных с использованием новейших технологий и клиент-серверных технологий в Интернете, создавать и реализовывать бизнес- | ОК-2 способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности; | знать: основные определения и понятия Web-конструирования и Web-программирования, основные приемы создания и продвижения сайтов; уметь: разрабатывать и продвигать проблемно-ориентированные Web-ресурсы (контентсервисов); владеть: проектирования, |

| | | |
|---|--|---|
| проекты в Интернете, участвовать в существующих бизнес-проектах | <p>ПК-6 способен к постановке и модернизации отдельных лабораторных работ и практикумов по дисциплинам направления;</p> <p>ПК-13 способен проектировать и применять инструментальные средства реализации программно-аппаратных проектов;</p> | разработки и маркетинга проблемно-ориентированных Web-ресурсов (контент-сервисов) |
|---|--|---|

3. Место дисциплины в структуре ООП

Дисциплина "Программирование Web-ориентированных приложений" относится к дисциплинам базовой части цикла профессиональных дисциплин (БЗ), обеспечивающих теоретическую и практическую подготовку магистрантов в области 710100 «Информатика и вычислительная техника».

Изучение данной дисциплины базируется на дисциплинах базовой части цикла профессиональных дисциплин таких как "Технология разработки программного обеспечения".

Основные положения дисциплины должны быть использованы в дальнейшем при изучении дисциплин "Технологии программирования", "Технологии разработки мобильных приложений".

4. Карта компетенций дисциплины в разрезе тем (разделов)

| Разделы, № и название темы | Компетенции | | | |
|---|-------------|------|-------|-------------------------|
| | ОК-2 | ПК-6 | ПК-13 | Σ общее кол-во комп-ций |
| I модуль | | | | |
| Тема 1. Введение в web-технологии: структура и принципы Web | + | | + | 2 |
| Тема 2. Клиент-серверные технологии Web. Протокол HTTP | + | + | + | 3 |
| Тема 3. Сценарии и приложения, выполняющиеся на стороне клиента | | | + | 1 |
| Тема 4. Серверные web-приложения | + | + | | 2 |
| Тема 5. Языки разработки сценариев Perl и PHP | | + | | 2 |
| II модуль | | | | |
| Тема 6. Введение в C# и платформу Visual Studio.Net | | | + | 1 |
| Тема 7. Архитектура web-приложений ASP.NET. Разработка web-приложений на платформе .NET | | + | | 1 |
| Тема 8. Интерфейсы взаимодействия web-приложений с СУБД | + | + | + | 3 |
| Тема 9. Хостинг. Бесплатный хостинг. FTP. Размещение Интернет-ресурса на сервере провайдера. | + | + | + | 3 |
| Тема 10. Организация процесса разработки web-контента. CMS/CMF | + | + | + | 3 |

5. Технологическая карта дисциплины

| Всего часов | Ауд. часов | СРМ | 1 модуль | | | | 2 модуль | | | | Итоговый контроль (ИК) | | | | Всего | | |
|------------------------------------|------------|-----|--|-------------|-----|-----------|--|-------------|-----|-----------|---|-------------|-----|------------------|------------|------------------|--|
| | | | Ауд.часы | | СРМ | РК-1 | Ауд.часы | | СРМ | РК-2 | Лекция | Лаборатория | СРМ | Итог.контр. (ИК) | | Дополн. балл (Д) | |
| | | | Лекция | Лаборатория | | | Лекция | Лаборатория | | | | | | | | | |
| 90 | 36 | 54 | 10 | 8 | 26 | | | 10 | 8 | 28 | | | | | | | |
| Баллы | | | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 10 | | |
| Итоги модулей и итогового контроля | | | $M1 = (\text{Лек} + \text{Лаб} + \text{СРМ} + \text{РК1}) / 4$ | | | | $M2 = (\text{Лек} + \text{Лаб} + \text{СРМ} + \text{РК2}) / 4$ | | | | $\text{ИК} = (\text{Лек} + \text{Лаб} + \text{СРМ}) / 3$, $\text{Экз} = M1 + M2 + \text{ИК} + \text{Д}$ | | | | 100 | | |

Ауд. – аудиторные, ТК – текущий контроль, РК – рубежный контроль, СРМ – самостоятельная работа магистрантов, ИК – итоговый контроль, Д – дополнительный балл, М – модуль, Лек – лекции, Лаб – лабораторные.

6. Карта накапливаемости баллов по дисциплине

| I модуль | | | | | | |
|--------------|-----------|--|---------------------|--|-----------|--|
| № | Лекции | | Лабораторные работы | | СРМ | |
| | Часы | Балл (№1 тема+№2 тема+...+№11 тема)/11 | Часы | Балл (№1 тема+№2 тема+...+№11 тема)/11 | Часы | Балл (№1 тема+№2 тема+...+№11 тема)/11 |
| ТК-1 | | | | | | |
| №1 тема | 1 | 30 | | 30 | 2 | 30 |
| №2 тема | 1 | 30 | | 30 | 2 | 30 |
| №3 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №4 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №5 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| Всего | 10 | 30 | 3 | 30 | 12 | 30 |
| ТК-2 | | | | | | |
| №1 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №2 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| №3 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №4 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| №5 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| Всего | 10 | 30 | 5 | 30 | 14 | 30 |
| Итого | 10 | $(\text{ТК1} + \text{ТК2}) / 2$ | 8 | $(\text{ТК1} + \text{ТК2}) / 2$ | 26 | $(\text{ТК1} + \text{ТК2}) / 2$ |
| | | 30 | | 30 | | 30 |

| Текущий контроль | | | РК1 | $M1 = (\text{Лек} + \text{Лаб} + \text{СРМ} + \text{РК1}) / 4$ |
|------------------|------|-----|-----|--|
| Лек. | Лаб. | СРМ | | |
| 30 | 30 | 30 | 30 | 30 |

| II модуль | | | | | | |
|--------------|-----------|---|---------------------|---|-----------|---|
| № | Лекции | | Лабораторные работы | | СРМ | |
| | Часы | Балл (№1 тема+№2 тема+...+№11 тема)/11 | Часы | Балл (№1 тема+№2 тема+...+№11 тема)/11 | Часы | Балл (№1 тема+№2 тема+...+№11 тема)/11 |
| ТК-3 | | | | | | |
| №1 тема | 1 | 30 | | 30 | 2 | 30 |
| №2 тема | 1 | 30 | | 30 | 2 | 30 |
| №3 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| №4 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №5 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| Всего | 10 | 30 | 3 | 30 | 14 | 30 |
| ТК-4 | | | | | | |
| №1 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №2 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| №3 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №4 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| №5 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| Всего | 10 | 30 | 5 | 30 | 14 | 30 |
| Итого | 10 | (ТК3+ ТК4)/2 | 8 | (ТК3+ ТК4)/2 | 28 | (ТК3+ТК4)/2 |
| | | 30 | | 30 | | 30 |

| II модуль | | | | |
|------------------|------|-----|-----|------------------------|
| Текущий контроль | | | РК2 | M1=(Лек+Лаб+СРМ+РК2)/4 |
| Лек. | Лаб. | СРМ | | |
| 30 | 30 | 30 | 30 | 30 |

7. Тематический план распределения часов по видам занятий

| Разделы, № и название темы | Всего | Лекции | Лаб. работа. | СРМ | Обр. технол огии | Оценоч. средства |
|---|-----------|-----------|-----------------|-----------|----------------------------|--|
| I модуль | | | | | | |
| Тема 1. Введение в web-технологии: структура и принципы Web | 6 | 2 | | 4 | КОТ, ИТО, ИТ, ГрТ | Устный опрос, презентация, демонстрация |
| Тема 2. Клиент-серверные технологии Web. Протокол HTTP | 8 | 2 | 2 | 6 | КОТ, ИТО, ИТ, ГрТ | Устный опрос, презентация, демонстрация |
| Тема 3. Сценарии и приложения, выполняющиеся на стороне клиента | 8 | 2 | 2 | 4 | КОТ, ИТО, ИТ, ГрТ | Устный опрос, игры, презентация, демонстрация |
| Тема 4. Серверные web-приложения | 10 | 2 | 2 | 6 | КОТ, ИТО, ИТ, ГрТ | Устный опрос, игры, презентация, демонстрация |
| Тема 5. Языки разработки сценариев Perl и PHP | 10 | 2 | 2 | 6 | КОТ, ИТО, ИТ, ГрТ | Устный опрос, презентация, демонстрация |
| Итого Модуль 1: | 42 | 10 | 8 | 26 | | |
| II модуль | | | | | | |
| Тема 6. Введение в C# и платформу Visual Studio.Net | 6 | 2 | | 4 | КОТ, ИТО, ИТ, ГрТ | Устный опрос, презентация, демонстрация |
| Тема 7. Архитектура web-приложений ASP.NET. Разработка web-приложений на платформе .NET | 8 | 2 | 2 | 6 | КОТ, ИТО, ИТ, ГрТ | Устный опрос, игры, презентация, демонстрация |
| Тема 8. Интерфейсы взаимодействия web-приложений с СУБД | 8 | 2 | 2 | 4 | КОТ, ИТО, ИТ, ГрТ | Устный опрос, презентация, демонстрация |
| Тема 9. Хостинг. Бесплатный хостинг. FTP. Размещение Интернет-ресурса на сервере провайдера. | 10 | 2 | 2 | 6 | КОТ, ИТО, ИТ, ГрТ | Устный опрос, презентация, демонстрация |

| | | | | | | |
|---|-----------|-----------|-----------|-----------|-------------------|---|
| Тема 10. Организация процесса разработки web-контента. CMS/CMF | 10 | 2 | 2 | 6 | КОТ, ИТО, ИТ, ГрТ | Устный опрос, игры, презентация, демонстрация |
| Итого Модуль 2: | 42 | 10 | 8 | 28 | | |
| Всего: | 90 | 20 | 16 | 54 | | |

8. Программа дисциплины

Введение в web-технологии: структура и принципы Web. Клиент-серверные технологии Web. Протокол HTTP. Сценарии и приложения, выполняющиеся на стороне клиента. Серверные web-приложения. Языки разработки сценариев Perl и PHP. Введение в C# и платформу Visual Studio.Net. Архитектура web-приложений ASP.NET. Разработка web-приложений на платформе .NET. Интерфейсы взаимодействия web-приложений с СУБД. Хостинг. Бесплатный хостинг. FTP. Размещение Интернет-ресурса на сервере провайдера.

9. Календарно-тематический план по видам занятий

| № и название темы | № лекции, комп. | Наименование изучаемых вопросов | Лит-ра | Лекции | Лаб. работа. | СРМ | Обр. Технологии |
|--|-------------------|--|------------|--------|--------------|-----|-------------------|
| I модуль | | | | | | | |
| Тема 1. Введение в web-технологии: структура и принципы Web | ОК-2, ПК-13 | <p><i>План лекции:</i></p> <p>1.1. Развитие Интернет</p> <p>1.2. Стандартизация в интернет</p> <p>1.3. Стек протоколов TCP/IP</p> <p>1.4. DNS – система доменных имен</p> <p>1.5. World Wide Web (WWW)</p> <p><i>Контрольные вопросы:</i></p> <p>Общее устройство сети интернет.</p> <p>Понятие домена и управление доменами.</p> <p>Протоколы интернет.</p> <p><i>План СРМ:</i></p> <p>3.1. Понятие прокси-сервер</p> <p>3.2. Протоколы Интернет прикладного уровня</p> | 1,2 3,4 | 2 | | 4 | КОТ, ИТО, ИТ, ГрТ |
| Тема 2. Клиент-серверные технологии Web. Протокол HTTP | ОК-2, ПК-6, ПК-13 | <p><i>План лекции:</i></p> <p>Клиент-серверные технологии Web.</p> <p>Протокол HTTP</p> <p>Обеспечение безопасности передачи данных HTTP</p> <p>Cookie</p> <p><i>Контрольные вопросы:</i></p> <p>Протоколы интернет.</p> <p>Понятие домена и управление доменами.</p> <p><i>План лаб. работы:</i></p> <p>Протокол HTTP</p> <p><i>План СРМ:</i></p> <p>Веб-серверы, Веб-страницы, Протоколы</p> | 2 4 | 2 | 2 | 6 | КОТ, ИТО, ИТ, ГрТ |
| Тема 3. Сценарии и приложения, выполняющиеся на стороне клиента | ПК-13 | <p><i>План лекции:</i></p> <p>Программы, выполняющиеся на клиент-машине</p> <p>Программы, выполняющиеся на клиент-машине</p> | 1 3,4 | 2 | 2 | 4 | КОТ, ИТО, ИТ, ГрТ |

| | | | | | | | |
|--|---------------|---|----------|---|---|---|----------------------------|
| | | <p>Введение в JScript <i>Контрольные вопросы:</i> Синтаксис внедрения javascript Динамика пользовательского интерфейса web-приложения <i>План лаб. работы:</i> JavaScript. Динамическое изменение html-документа в браузере. <i>План СРМ:</i> Программирование на JavaScript</p> | | | | | |
| Тема 4. Серверные web-приложения | ОК-2, ПК-6 | <p><i>План лекции:</i> Стандарт CGI Сценарии (Python, Ruby, ASP ISAPI) <i>Контрольные вопросы:</i> Классификация веб-сайтов. Общие понятия языка разметки гипертекста – HTML. <i>План лаб. работы:</i> Методы передачи параметров между страницами (GET, POST). Обработка действий пользователя при помощи форм. Использование вспомогательных переменных <i>План СРМ:</i> Разработка проекта.</p> | 1,2 4 | 2 | 2 | 6 | КОТ, ИТО, ИТ, ГрТ |
| Тема 5. Языки разработки сценариев Perl и PHP | ПК-6 | <p><i>План лекции:</i> Язык Perl Язык PHP Введение в программирование на стороне сервера на примере PHP. Принцип работы. Синтаксис языка программирования PHP. Переменные. Константы. Операторы в PHP. Циклы. <i>Контрольные вопросы:</i> Добавление мультимедийных файлов Язык запросов MySQL <i>План лаб. работы:</i> Массивы. Работа со строками. Функции в PHP. Встроенные функции. Работа с датой и временем в PHP. Связь PHP и HTML <i>План СРМ:</i> Разработка проекта.</p> | 1,2 4 | 2 | 2 | 6 | КОТ, ИТО, ИТ, ГрТ |
| II модуль | | | | | | | |

| | | | | | | | |
|--|-------------------------|--|----------|---|---|---|----------------------------|
| <p>Тема 6. Введение в С# и платформу Visual Studio.Net</p> | ПК-13 | <p><i>План лекции:</i> Введение в С# и платформу Visual Studio.Net Основы С# <i>Контрольные вопросы:</i> Назовите модификаторы доступности, существующие в Visual С#, и поясните их назначение. <i>План лаб. работы:</i> Клиентские сценарии. Использование регулярных выражений <i>План СРМ:</i> Введение в JavaScript. Программное взаимодействие с HTML документами на основе DOM API</p> | 1,2 4 | 2 | | 4 | КОТ, ИТО, ИТ, ГрТ |
| <p>Тема 7. Архитектура web-приложений ASP.NET. Разработка web-приложений на платформе .NET</p> | ПК-6 | <p><i>План лекции:</i> Серверные элементы управления ASP.NET Работа с источниками данных в ASP.NET <i>Контрольные вопросы:</i> Серверные элементы управления ASP.NET Работа с источниками данных в ASP.NET <i>План лаб. работы:</i> Разработка проекта. <i>План СРМ:</i> Разработка проекта.</p> | 1,2 3 | 2 | 2 | 6 | КОТ, ИТО, ИТ, ГрТ |
| <p>Тема 8. Интерфейсы взаимодействия web-приложений с СУБД</p> | ОК-2, ПК-6, ПК-13 | <p><i>План лекции:</i> Интерфейс ODBC. Интерфейс программирования приложений для доступа к данным ADO. Модель доступа приложений к источникам данных ADO.NET <i>Контрольные вопросы:</i> Интерфейсы взаимодействия с БД <i>План лаб. работы:</i> Разработка проекта. <i>План СРМ:</i> Разработка проекта.</p> | 1,2 3 | 2 | 2 | 4 | КОТ, ИТО, ИТ, ГрТ |
| <p>Тема 9. Хостинг. Бесплатный хостинг. FTP. Размещение Интернет-ресурса на сервере провайдера. Регистрация Интернет-ресурса в каталогах и поисковых системах</p> | ОК-2, ПК-6, ПК-13 | <p><i>План лекции:</i> Хостинг. Бесплатный хостинг. FTP. Размещение Интернет-ресурса на сервере провайдера. <i>Контрольные вопросы:</i> Регистрация Интернет-ресурса в каталогах и поисковых системах <i>План лаб. работы:</i> Режимы работы с базой данных Представление прочитанных данных <i>План СРМ:</i></p> | 1 3,4 | 2 | 2 | 6 | КОТ, ИТО, ИТ, ГрТ |

| | | | | | | | |
|---|-------------------------|--|----------|---|---|---|----------------------------|
| | | Хостинг. Подобрать бесплатный хостинг для сайта. Ознакомиться с условиями размещения. Подготовить материалы для размещения и популяризации ресурса. | | | | | |
| Тема 10. Организация процесса разработки web-контента. CMS/CMF | ОК-2, ПК-6, ПК-13 | <i>План лекции:</i> База данных и сервисы Firebase Работа с Realtime Database <i>Контрольные вопросы:</i> Проблемы совместимости мобильных приложений со старыми версиями операционных систем Проблемы масштабирования СУБД в мобильных приложениях <i>План лаб. работы:</i> Создание приложений с поддержкой Realtime Database Соединение с Firebase <i>План СРМ:</i> Разработка проекта. | 1 3,4 | 2 | 2 | 6 | КОТ, ИТО, ИТ, ГрТ |

10. Разработка проекта.

Выбрать проект (либо из списка; либо предложить инициативный проект). Разработать структуру БД. Создать и заполнить БД. Разработать программное обеспечение проекта, обеспечивающее работу с информацией из БД как на стороне клиента, так и владельца ресурса.

1. Каталог фильмов
2. Форум
3. Гостевая книга
4. Сборник задач
5. Магазин
6. Кафе
7. Каталог книг
8. Журнал класса
9. Каталог услуг
10. Каталог продукции

11. Образовательные технологии

Групповые технологии (ГТ) - обеспечение активности учебного процесса, достижение высокого уровня усвоения содержания. Особенности организации:

- магистранты на уроке делится на группы для решения конкретных учебных задач,
- каждая группа получает определённое задание и выполняет его сообща под руководством лидера группы или учителя; -задания в группе выполняются таким способом, который позволяет учитывать и оценивать индивидуальный вклад каждого члена группы,
- состав группы непостоянный, он подбирается с учётом того, чтобы могли реализовываться учебные возможности каждого члена группы, в зависимости от содержания и характера предстоящей работы.

Игровые технологии (ИТ) - цель ставится перед учащимися в форме игровой задачи, учебная деятельность подчиняется правилам игры, учебный материал используется в качестве средства игры, в учебную деятельность включается элемент соревнования, успешное выполнение задания связывается с игровым результатом.

Педагогические игры по характеру педагогического процесса подразделяются на группы:

- а) обучающие, тренировочные, контролирующие и обобщающие;
- б) познавательные, воспитательные, развивающие;
- в) репродуктивные, продуктивные, творческие.

Информационные технологии обучения (ИТО)- формирование умений работать с информацией, развитие коммуникативных способностей, подготовка личности «информационного общества», дать магистранту так много учебного материала, как только он может усвоить, формирование исследовательских умений, умений принимать оптимальные решения. Компьютер используется на всех этапах процесса обучения- при объяснении нового материала, при закреплении знаний.

12. Учебно-методическое и информационное обеспечение дисциплины

12.1. Основные литературы

[1. Android. Программирование для профессионалов / Б. Харди \[и др.\]. – 2-е изд. – СПб.: Питер, 2016. – 640 с.](#)

[2. Основы создания приложений // Developers \[Электронный ре- сурс\]. – Режим доступа: <https://developer.android.com/guide>. – Дата доступа: 11.08.2019.](#)

12.2. Дополнительные литературы

[3. Программирование под Android. / Медникс З. \[и др.\]. – 2-е изд. – СПб.: Питер, 2013. – 560 с.](#)

[4. Start Android: учебник по Android для начинающих и продвинутых // STARTANDROID \[Электронный ресурс\]. – Режим доступа: <https://startandroid.ru/ru/>. – Дата доступа: 11.08.2019.](#)

12.3. Материально-техническое и информационное обеспечение дисциплины

При освоении дисциплины для выполнения лабораторных работ необходимы персональные компьютеры с набором программного обеспечения: Adobe Photoshop, пакет Subline Text 3, OpenServer, web-браузер.

УМК по дисциплине в электронном виде находится в KelBil, AVN и кабинете кафедры АСЦТ.

12.4. Методические указания магистрантам

Изучение программы курса. На лекциях преподаватель рассматривает вопросы программы курса, составленной в соответствии с государственным образовательным стандартом. Из-за недостаточного количества аудиторных часов некоторые темы не удается осветить в полном объеме, поэтому преподаватель, по своему усмотрению, некоторые вопросы выносит на самостоятельную работу магистрантов, рекомендуя ту или иную литературу.

Кроме этого, для лучшего освоения материала и систематизации знаний по дисциплине, необходимо постоянно разбирать материалы лекций по конспектам и учебным пособиям. В случае необходимости обращаться к преподавателю за консультацией. Полный список литературы по дисциплине приведен в пункте **12.1.** «Учебно-методическое обеспечение дисциплины».

В целом, на один час аудиторных занятий отводится один час самостоятельной работы.

Лабораторные работы. При изучении курса «Программирование Web-ориентированных приложений» необходимо выполнять и вовремя сдавать преподавателю индивидуальные лабораторные работы.

13. Политика выставления баллов

Выставление оценок на экзаменах осуществляется на основе принципов объективности, справедливости, всестороннего анализа качества знаний магистрантов, и других положений, способствующих повышению надежности оценки знаний, обучающихся и устранению субъективных факторов.

Оценка знаний (академической успеваемости) магистранту осуществляется по 30 и 100 балльной системам (шкале) следующим образом:

| 30 балльная система | Баллы (Рейтинг) | Оценка по буквенной системе | Цифровой эквивалент оценки по GPA | Оценка по традиционной системе |
|---------------------|-----------------|-----------------------------|-----------------------------------|--------------------------------|
| 26- 30 | 87 – 100 | A | 4,0 | Отлично |
| 24- 25 | 80–86 | B | 3,33 | Хорошо |
| 22- 23 | 74–79 | C | 3,0 | |
| 20- 21 | 68–73 | D | 2,33 | Удовлетворительно |
| 18- 19 | 61–67 | E | 2,0 | |
| 9-17 | 31 -60 | FX | 0 | Неудовлетворительно |
| 0- 8 | 0-30 | F | 0 | |

1. Критерий оценки магистранта на лекционных занятиях Шкала оценивания 26 – 30 баллов:

- магистрант демонстрирует полное понимание темы согласно цели пройденной лекции;
- магистрант способен успешно обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

Шкала оценивания 24 – 25 баллов:

- магистрант демонстрирует полное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

Шкала оценивания 22 – 23 баллов:

- магистрант демонстрирует полное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

Шкала оценивания 20 – 21 баллов:

- магистрант демонстрирует неполное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

Шкала оценивания 18 – 19 баллов:

- магистрант демонстрирует неполное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет частично систематизировать, структурировать и аргументировать материал.

Шкала оценивания 9 – 17 баллов:

- магистрант не полностью понимает темы согласно цели пройденной лекции;
- магистрант не способен обосновывать свою точку зрения;
- не умеет систематизировать, структурировать и аргументировать материал.

Шкала оценивания 0– 18 баллов:

- магистрант непонимает темы согласно цели пройденной лекции;
- магистрант не способен обосновывать свою точку зрения;
- не умеет систематизировать, структурировать и аргументировать материал.

2. Критерий оценки магистранта на лабораторных занятиях

Шкала оценивания 26 – 30 баллов:

- магистрант демонстрирует полное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно выполняет поставленную задачу;
- уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 24 – 25 баллов:

- магистрант демонстрирует полное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно выполняет поставленную задачу;
- полностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 22 – 23 баллов:

- магистрант демонстрирует неполное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно не может выполнить поставленную задачу;
- неуверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 20 – 21 баллов:

- магистрант демонстрирует неполное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно не может выполнить поставленную задачу;
- полностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 18 – 19 баллов:

- магистрант демонстрирует неполное понимание цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- полностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- частично соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 9 – 17 баллов:

- магистрант не понимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- неуверенно излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- не соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 0 – 18 баллов:

- магистрант непонимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- не может изложить свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- не соблюдает правила техники безопасности при выполнении лабораторной работы.

3. Критерий оценки магистранта на самостоятельных работах магистрантов

Шкала оценивания 26 – 30 баллов:

- магистрант демонстрирует полное понимание проблемы;
- магистрант выполняет все требования, предъявляемые к заданию;
- уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 24 – 25 баллов:

- магистрант демонстрирует значительное понимание проблемы;
- магистрант выполняет все требования, предъявляемые к заданию;
- не полностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 22 – 23 баллов:

- магистрант демонстрирует частичное понимание проблемы;
- магистрант выполняет большинство требования, предъявляемые к заданию;
- неуверенно излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 20 – 21 баллов:

- магистрант демонстрирует небольшое понимание проблемы;
- магистрант выполняет немногие требования, предъявляемые к заданию;
- не полностью излагает свое решение при выполнении лабораторной работы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 18 – 19 баллов:

- магистрант демонстрирует непонимание проблемы;
- не полностью излагает свое решение при выполнении лабораторной работы;
- частично соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 9 – 17 баллов:

- магистрант не понимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- неуверенно излагает свое решение при выполнении лабораторной работы;
- несоблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 0 – 18 баллов:

- магистрант непонимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- нет ответа, не было попытки решить задачу;
- несоблюдает правила техники безопасности при выполнении лабораторной работы.

III. СИЛЛАБУС

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ

ОШСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

КАФЕДРА АСЦТ

«Согласована»

Председатель Методического совета
факультета МИТ доцент, канд.
п.и.к.: _____ Д. Зулпукарова

«Утверждена»

на заседании кафедры АСЦТ
от 27 августа 2020 года, протокол №1
Зав. каф. АСЦТ, доцент

« ___ » _____ 2020 г.

Молдоярлов У.Д. _____

СИЛЛАБУС

по дисциплине: "Программирование Web-ориентированных приложений" для магистрантов очного отделения, обучающихся по направлению: 710100 «Информатика и вычислительная техника»

Сетка часов по учебному плану

| <u>Программирование Web-ориентированных приложений</u> | Всего | Ауд. зан. | Аудит.зан. | | СРМ | Отчетность | |
|--|----------------|--------------|------------|--------|-------|------------|-------|
| | | | Лекции | Лабор. | | 2 сем | 2 сем |
| 1 курс, 2 сем. | 90ч (3 кр.) | 36 ч. | 20 ч. | 16 ч. | 56 ч. | РК-2 | Экз |

Данные о преподавателе:

Молдоярлов Уларбек Дуйшобекович – доцент кафедры АСЦТ
факультета Математика и информационных технологий ОшГУ

Стаж работы: 15 лет.

Рабочее место:

723500. Главный корпус ОшГУ, ул. Ленина 331, кабинет – 205.

Контактная информация:

Моб. телефон: 0772843108, 0554490100

E-mail: ular@oshsu.kg

Дежурство преподавателя проводится в кабинете 205 по пятницам с 14.00 до 16.00

Дата: 2020 -2021 учебный год

СОДЕРЖАНИЕ

| | |
|---|--|
| <u>1. Цели освоения дисциплины</u> | 10 |
| <u>2. Результаты обучения и компетенции, формируемые в процессе изучения дисциплины "Программирование Web-ориентированных приложений"</u> | 10 |
| <u>3. Место дисциплины в структуре ООП</u> | 11 |
| <u>4. Карта компетенций дисциплины в разрезе тем (разделов)</u> | 11 |
| <u>5. Технологическая карта дисциплины</u> | 12 |
| <u>6. Карта накапливаемости баллов по дисциплине</u> | 12 |
| <u>7. Тематический план распределения часов по видам занятий</u> | 14 |
| <u>8. Программа дисциплины</u> | 16 |
| <u>9. Цели и результаты обучения по темам дисциплины</u> . Ошибка! Закладка не определена. | |
| <u>10. Календарно-тематический план по видам занятий</u> | 16 |
| <u>11. Образовательные технологии</u> | 19 |
| <u>12. Учебно-методическое и информационное обеспечение дисциплины</u> | 20 |
| <u>13. Политика выставления баллов</u> | 20 |
| <u>14. Политика курса</u> | Ошибка! Закладка не определена. |

1. Цели освоения дисциплины

Дисциплина «Программирование Web-ориентированных приложений» призвана содействовать знакомству магистрантов с компьютерными телекоммуникациями и возможными подходами к разработке гипертекстовых документов, предназначенных для публикации в глобальной компьютерной сети Internet. Она важна с той точки зрения, что позволяет развивать способности магистрантов, связанные с общей культурой работы в глобальной сети. Курс закрепляет навыки работы с текстом и графикой, а также навыков программирования и проектирования и разработки информационных систем, являясь, таким образом, прямым продолжением курсов «Web программирование», «Разработка клиент-серверных прилож», «Проектирование АСОИ», «Базы данных», «Программирование» и многих других.

Целью курса является освоение практических приемов Web-конструирования и Web-программирования.

Основные задачи курса:

- закрепление знакомства с принципами функционирования глобальной компьютерной сети Internet, общими подходами к поиску и отбору информации в сети;
- обучение разработке Web-страниц на основе комплексного подхода;
- обучение программированию в Internet на стороне клиента и сервера;
- обучение использованию баз данных при разработке Web-проектов;
- обучение способам маркетинга в Internet, рекламы и продвижения разработанных Internet-ресурсов.

В лекционной части курса рассматриваются общие принципы Web-конструирования. Изучение всех тем сопровождается иллюстрирующими примерами.

Лабораторные работы в компьютерных классах служат для индивидуальной работы магистрантов над учебными задачами и итоговым проектом с целью выработки и закрепления практических навыков Web-конструирования и Web-программирования.

2. Результаты обучения и компетенции, формируемые в процессе изучения дисциплины "Программирование Web-ориентированных приложений"

2. результате изучения дисциплины магистрант достигнет следующих **результатов обучения (РОд)**, соответствующих ожидаемым **результатам освоения образовательной программы (РОоп)** и заданным для дисциплины **компетенциям**:

| Код РОоп и его формулировка | Код компетенции ООП и его формулировка | Код РО дисциплины (РОд) и его формулировка |
|---|---|--|
| <p>РО-8 Веб-активность: умеет создавать веб-приложения, создавать базы данных и работать с системами управления базами данных с использованием новейших технологий и клиент-серверных технологий в Интернете, создавать и реализовывать бизнес-проекты в Интернете, участвовать в существующих бизнес-проектах</p> | <p>ОК-2 способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности;</p> <p>ПК-6 способен к постановке и модернизации отдельных лабораторных работ и практикумов по дисциплинам направления;</p> <p>ПК-13 способен проектировать и применять инструментальные средства реализации программно-аппаратных проектов;</p> | <p>знать: основные определения и понятия Web-конструирования и Web-программирования, основные приемы создания и продвижения сайтов;</p> <p>уметь: разрабатывать и продвигать проблемно-ориентированные Web-ресурсы (контентсервисов);</p> <p>владеть: проектирования, разработки и маркетинга проблемно-ориентированных Web-ресурсов (контент-сервисов)</p> |

3. Место дисциплины в структуре ООП

Дисциплина "Программирование Web-ориентированных приложений" относится к дисциплинам базовой части цикла профессиональных дисциплин (БЗ), обеспечивающих теоретическую и практическую подготовку магистрантов в области 710100 «Информатика и вычислительная техника».

Изучение данной дисциплины базируется на дисциплинах базовой части цикла профессиональных дисциплин таких как "Технология разработки программного обеспечения".

Основные положения дисциплины должны быть использованы в дальнейшем при изучении дисциплин "Технологии программирования", "Технологии разработки мобильных приложений".

5. Технологическая карта дисциплины

| Всего часов | Ауд. часов | СРМ | 1 модуль | | | | 2 модуль | | | | Итоговый контроль (ИК) | | | | | Всего | | |
|------------------------------------|------------|-----|--|-------------|-----|-----------|--|-------------|-----|------|--|-------------|-----|-------------------|------------------|------------|-----------|--|
| | | | Ауд. часы | | СРМ | РК-1 | Ауд. часы | | СРМ | РК-2 | Лекция | Лаборатория | СРМ | Итог. контр. (ИК) | Дополн. балл (Д) | | | |
| | | | Лекция | Лаборатория | | | Лекция | Лаборатория | | | | | | | | | | |
| 90 | 36 | 54 | 10 | 8 | 26 | | | 10 | 8 | 28 | | | | | | | | |
| Баллы | | | 30 | 30 | 30 | 30 | | | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 10 | |
| Итоги модулей и итогового контроля | | | $M1 = (\text{Лек} + \text{Лаб} + \text{СРМ} + \text{РК1})/4$ | | | | $M2 = (\text{Лек} + \text{Лаб} + \text{СРМ} + \text{РК2})/4$ | | | | $\text{ИК} = (\text{Лек} + \text{Лаб} + \text{СРМ})/3,$ $\text{Экз} = M1 + M2 + \text{ИК} + \text{Д}$ | | | | | 100 | | |

Ауд. – аудиторные, ТК – текущий контроль, РК – рубежный контроль, СРМ – самостоятельная работа магистрантов, ИК – итоговый контроль, Д – дополнительный балл, М – модуль, Лек – лекции, Лаб – лабораторные.

6. Карта накапливаемости баллов по дисциплине

| I модуль | | | | | | |
|--------------|-----------|--|---------------------|--|-----------|--|
| № | Лекции | | Лабораторные работы | | СРМ | |
| | Часы | Балл (№1 тема+№2 тема+...+№11 тема)/11 | Часы | Балл (№1 тема+№2 тема+...+№11 тема)/11 | Часы | Балл (№1 тема+№2 тема+...+№11 тема)/11 |
| ТК1 | | | | | | |
| №1 тема | 1 | 30 | | 30 | 2 | 30 |
| №2 тема | 1 | 30 | | 30 | 2 | 30 |
| №3 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №4 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №5 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| Всего | 10 | 30 | 3 | 30 | 12 | 30 |
| ТК2 | | | | | | |
| №1 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №2 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| №3 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №4 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| №5 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| Всего | 10 | 30 | 5 | 30 | 14 | 30 |
| Итого | 10 | (ТК1+ ТК2)/2 | 8 | (ТК1+ ТК2)/2 | 26 | (ТК1+ТК-2)/2 |
| | | 30 | | 30 | | 30 |

I модуль

| Текущий контроль | | | РК1 | $M1 = (\text{Лек} + \text{Лаб} + \text{СРМ} + \text{РК1})/4$ |
|------------------|------|-----|-----|--|
| Лек. | Лаб. | СРМ | | |
| 30 | 30 | 30 | 30 | 30 |

II модуль

| № | Лекции | | Лабораторные работы | | СРМ | |
|--------------|-----------|---|---------------------|---|-----------|---|
| | Часы | Балл (№1 тема+№2 тема+...+№11 тема)/11 | Часы | Балл (№1 тема+№2 тема+...+№11 тема)/11 | Часы | Балл (№1 тема+№2 тема+...+№11 тема)/11 |
| ТК3 | | | | | | |
| №1 тема | 1 | 30 | | 30 | 2 | 30 |
| №2 тема | 1 | 30 | | 30 | 2 | 30 |
| №3 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| №4 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №5 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| Всего | 10 | 30 | 3 | 30 | 14 | 30 |
| ТК4 | | | | | | |
| №1 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №2 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| №3 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| №4 тема | 1 | 30 | 1 | 30 | 4 | 30 |
| №5 тема | 1 | 30 | 1 | 30 | 2 | 30 |
| Всего | 10 | 30 | 5 | 30 | 14 | 30 |
| Итого | 10 | (ТК3+ ТК4)/2 | 8 | (ТК3+ ТК4)/2 | 28 | (ТК3+ТК4)/2 |
| | | 30 | | 30 | | 30 |

II модуль

| Текущий контроль | | | РК2 | M1=(Лек+Лаб+СРМ+РК2)/4 |
|------------------|------|-----|-----|------------------------|
| Лек. | Лаб. | СРМ | | |
| 30 | 30 | 30 | 30 | 30 |

7. Тематический план распределения часов по видам занятий

| Разделы, № и название темы | Всего | Лекции | Лаб. работа. | СРМ |
|---|-----------|-----------|--------------|-----------|
| I модуль | | | | |
| Тема 1. Введение в web-технологии: структура и принципы Web | 6 | 2 | | 4 |
| Тема 2. Клиент-серверные технологии Web. Протокол HTTP | 8 | 2 | 2 | 6 |
| Тема 3. Сценарии и приложения, выполняющиеся на стороне клиента | 8 | 2 | 2 | 4 |
| Тема 4. Серверные web-приложения | 10 | 2 | 2 | 6 |
| Тема 5. Языки разработки сценариев Perl и PHP | 10 | 2 | 2 | 6 |
| Итого Модуль 1: | 42 | 10 | 8 | 26 |
| II модуль | | | | |
| Тема 6. Введение в C# и платформу Visual Studio.Net | 6 | 2 | | 4 |
| Тема 7. Архитектура web-приложений ASP.NET. Разработка web-приложений на платформе .NET | 8 | 2 | 2 | 6 |
| Тема 8. Интерфейсы взаимодействия web-приложений с СУБД | 8 | 2 | 2 | 4 |
| Тема 9. Хостинг. Бесплатный хостинг. FTP. Размещение Интернет-ресурса на сервере провайдера. | 10 | 2 | 2 | 6 |
| Тема 10. Организация процесса разработки web-контента. CMS/CMF | 10 | 2 | 2 | 6 |
| Итого Модуль 2: | 42 | 10 | 8 | 28 |
| Всего: | 90 | 20 | 16 | 54 |

8. Программа дисциплины

Введение в web-технологии: структура и принципы Web. Клиент-серверные технологии Web. Протокол HTTP. Сценарии и приложения, выполняющиеся на стороне клиента. Серверные web-приложения. Языки разработки сценариев Perl и PHP. Введение в C# и платформу Visual Studio.Net. Архитектура web-приложений ASP.NET. Разработка web-приложений на платформе .NET. Интерфейсы взаимодействия web-приложений с СУБД. Хостинг. Бесплатный хостинг. FTP. Размещение Интернет-ресурса на сервере провайдера.

9. Календарно-тематический план по видам занятий

| № и название темы | Наименование изучаемых вопросов | Лит-ра | Лекции | Лаб. работа. | СРМ |
|--|--|------------|--------|--------------|-----|
| I модуль | | | | | |
| Тема 1. Введение в web-технологии: структура и принципы Web | <p><i>План лекции:</i></p> <p>1.1. Развитие Интернет</p> <p>1.2. Стандартизация в интернет</p> <p>1.3. Стек протоколов TCP/IP</p> <p>1.4. DNS – система доменных имен</p> <p>1.5. World Wide Web (WWW)</p> <p><i>Контрольные вопросы:</i></p> <p>Общее устройство сети интернет.</p> <p>Понятие домена и управление доменами.</p> <p>Протоколы интернет.</p> <p><i>План СРМ:</i></p> <p>Понятие прокси-сервер</p> <p>Протоколы Интернет прикладного уровня</p> | 1,2 3,4 | 2 | | 4 |
| Тема 2. Клиент-серверные технологии Web. Протокол HTTP | <p><i>План лекции:</i></p> <p>Клиент-серверные технологии Web.</p> <p>Протокол HTTP</p> <p>Обеспечение безопасности передачи данных HTTP</p> <p>Cookie</p> <p><i>Контрольные вопросы:</i></p> <p>Протоколы интернет.</p> <p>Понятие домена и управление доменами.</p> <p><i>План лаб. работы:</i></p> <p>Протокол HTTP</p> <p><i>План СРМ:</i></p> <p>Веб-серверы, Веб-страницы, Протоколы</p> | 2 4 | 2 | 2 | 6 |
| Тема 3. Сценарии и приложения, выполняющиеся на стороне клиента | <p><i>План лекции:</i></p> <p>Программы, выполняющиеся на клиент-машине</p> <p>Программы, выполняющиеся на клиент-машине</p> <p>Введение в JScript</p> <p><i>Контрольные вопросы:</i></p> <p>Синтаксис внедрения javascript</p> <p>Динамика пользовательского интерфейса web-приложения</p> <p><i>План лаб. работы:</i></p> <p>JavaScript. Динамическое изменение html-документа в браузере.</p> <p><i>План СРМ:</i></p> <p>Программирование на JavaScript</p> | 1 3,4 | 2 | 2 | 4 |

| | | | | | |
|---|--|----------|---|---|---|
| <p>Тема 4. Серверные web-приложения</p> | <p><i>План лекции:</i> Стандарт CGI Сценарии (Python, Ruby, ASP ISAPI) <i>Контрольные вопросы:</i> Классификация веб-сайтов. Общие понятия языка разметки гипертекста – HTML. <i>План лаб. работы:</i> Методы передачи параметров между страницами (GET, POST). Обработка действий пользователя при помощи форм. Использование вспомогательных переменных <i>План СРМ:</i> Разработка проекта.</p> | 1,2 4 | 2 | 2 | 6 |
| <p>Тема 5. Языки разработки сценариев Perl и PHP</p> | <p><i>План лекции:</i> Язык Perl Язык PHP Введение в программирование на стороне сервера на примере PHP. Принцип работы. Синтаксис языка программирования PHP. Переменные. Константы. Операторы в PHP. Циклы. <i>Контрольные вопросы:</i> Добавление мультимедийных файлов Язык запросов MySQL <i>План лаб. работы:</i> Массивы. Работа со строками. Функции в PHP. Встроенные функции. Работа с датой и временем в PHP. Связь PHP и HTML <i>План СРМ:</i> Разработка проекта.</p> | 1,2 4 | 2 | 2 | 6 |
| II модуль | | | | | |
| <p>Тема 6. Введение в C# и платформу Visual Studio.Net</p> | <p><i>План лекции:</i> Введение в C# и платформу Visual Studio.Net Основы C# <i>Контрольные вопросы:</i> Назовите модификаторы доступности, существующие в Visual C#, и поясните их назначение. <i>План лаб. работы:</i> Клиентские сценарии. Использование регулярных выражений <i>План СРМ:</i> Введение в JavaScript. Программное взаимодействие с HTML документами на основе DOM API</p> | 1,2 4 | 2 | | 4 |
| <p>Тема 7. Архитектура web-приложений ASP.NET. Разработка web-приложений на платформе .NET</p> | <p><i>План лекции:</i> Серверные элементы управления ASP.NET Работа с источниками данных в ASP.NET <i>Контрольные вопросы:</i> Серверные элементы управления ASP.NET Работа с источниками данных в ASP.NET <i>План лаб. работы:</i> Разработка проекта. <i>План СРМ:</i> Разработка проекта.</p> | 1,2 3 | 2 | 2 | 6 |
| <p>Тема 8. Интерфейсы</p> | <p><i>План лекции:</i></p> | 1,2 | 2 | 2 | 4 |

| | | | | | |
|---|--|----------|---|---|---|
| взаимодействия web-приложений с СУБД | Интерфейс ODBC. Интерфейс программирования приложений для доступа к данным ADO. Модель доступа приложений к источникам данных ADO.NET <i>Контрольные вопросы:</i> Интерфейсы взаимодействия с БД <i>План лаб. работы:</i> Разработка проекта. <i>План СРМ:</i> Разработка проекта. | 3 | | | |
| Тема 9. Хостинг. Бесплатный хостинг. FTP. Размещение Интернет-ресурса на сервере провайдера. Регистрация Интернет-ресурса в каталогах и поисковых системах | <i>План лекции:</i> Хостинг. Бесплатный хостинг. FTP. Размещение Интернет-ресурса на сервере провайдера. <i>Контрольные вопросы:</i> Регистрация Интернет-ресурса в каталогах и поисковых системах <i>План лаб. работы:</i> Режимы работы с базой данных Представление прочитанных данных <i>План СРМ:</i> Хостинг. Подобрать бесплатный хостинг для сайта. Ознакомиться с условиями размещения. Подготовить материалы для размещения и популяризации ресурса. | 1 3,4 | 2 | 2 | 6 |
| Тема 10. Организация процесса разработки web-контента. CMS/CMF | <i>План лекции:</i> База данных и сервисы Firebase Работа с Realtime Database <i>Контрольные вопросы:</i> Проблемы совместимости мобильных приложений со старыми версиями операционных систем Проблемы масштабирования СУБД в мобильных приложениях <i>План лаб. работы:</i> Создание приложений с поддержкой Realtime Database Соединение с Firebase <i>План СРМ:</i> Разработка проекта. | 1 3,4 | 2 | 2 | 6 |

10. Разработка проекта.

Выбрать проект (либо из списка; либо предложить инициативный проект). Разработать структуру БД. Создать и заполнить БД. Разработать программное обеспечение проекта, обеспечивающее работу с информацией из БД как на стороне клиента, так и владельца ресурса.

1. Каталог фильмов
2. Форум
3. Гостевая книга
4. Сборник задач
5. Магазин
6. Кафе
7. Каталог книг
8. Журнал класса

- 9. Каталог услуг
- 10. Каталог продукции

11. Образовательные технологии

Групповые технологии (ГТ) - обеспечение активности учебного процесса, достижение высокого уровня усвоения содержания. Особенности организации:

- магистранты на уроке делится на группы для решения конкретных учебных задач,
- каждая группа получает определённое задание и выполняет его сообща под руководством лидера группы или учителя; задания в группе выполняются таким способом, который позволяет учитывать и оценивать индивидуальный вклад каждого члена группы,
- состав группы непостоянный, он подбирается с учётом того, чтобы могли реализовываться учебные возможности каждого члена группы, в зависимости от содержания и характера предстоящей работы.

Игровые технологии (ИТ) - цель ставится перед учащимися в форме игровой задачи, учебная деятельность подчиняется правилам игры, учебный материал используется в качестве средства игры, в учебную деятельность включается элемент соревнования, успешное выполнение задания связывается с игровым результатом.

Педагогические игры по характеру педагогического процесса подразделяются на группы:

- а) обучающие, тренировочные, контролируемые и обобщающие;
- б) познавательные, воспитательные, развивающие;
- в) репродуктивные, продуктивные, творческие.

Информационные технологии обучения (ИТО)- формирование умений работать с информацией, развитие коммуникативных способностей, подготовка личности «информационного общества», дать магистранту так много учебного материала, как только он может усвоить, формирование исследовательских умений, умений принимать оптимальные решения. Компьютер используется на всех этапах процесса обучения- при объяснении нового материала, при закреплении знаний.

12. Учебно-методическое и информационное обеспечение дисциплины

12.1. Основные литературы

[1. Android. Программирование для профессионалов / Б. Харди \[и др.\]. – 2-е изд. – СПб.: Питер, 2016. – 640 с.](#)

[2. Основы создания приложений // Developers \[Электронный ре- курс\]. – Режим доступа: <https://developer.android.com/guide>. – Дата доступа: 11.08.2019.](#)

12.2. Дополнительные литературы

[3. Программирование под Android. / Медникс З. \[и др.\]. – 2-е изд. – СПб.: Питер, 2013. – 560 с.](#)

[4. Start Android: учебник по Android для начинающих и продвинутых // STARTANDROID \[Электронный ресурс\]. – Режим доступа: <https://startandroid.ru/ru/>. – Дата доступа: 11.08.2019.](#)

12.3. Материально-техническое и информационное обеспечение дисциплины

При освоении дисциплины для выполнения лабораторных работ необходимы персональные компьютеры с набором программного обеспечения: Adobe Photoshop, пакет Subline Text 3, web-браузер.

УМК по дисциплине в электронном виде находится в KelBil, AVN и кабинете кафедры АСЦТ.

12.4. Методические указания магистрантам

Изучение программы курса. На лекциях преподаватель рассматривает вопросы программы курса, составленной в соответствии с государственным образовательным стандартом. Из-за недостаточного количества аудиторных часов некоторые темы не удается

осветить в полном объеме, поэтому преподаватель, по своему усмотрению, некоторые вопросы выносит на самостоятельную работу магистрантов, рекомендуя ту или иную литературу.

Кроме этого, для лучшего освоения материала и систематизации знаний по дисциплине, необходимо постоянно разбирать материалы лекций по конспектам и учебным пособиям. В случае необходимости обращаться к преподавателю за консультацией. Полный список литературы по дисциплине приведен в пункте 12.1. «Учебно-методическое обеспечение дисциплины».

В целом, на один час аудиторных занятий отводится один час самостоятельной работы.

Лабораторные работы. При изучении курса «Программирование Web-ориентированных приложений» необходимо выполнять и вовремя сдавать преподавателю индивидуальные лабораторные работы.

13. Политика выставления баллов

Выставление оценок на экзаменах осуществляется на основе принципов объективности, справедливости, всестороннего анализа качества знаний магистрантов, и других положений, способствующих повышению надежности оценки знаний, обучающихся и устранению субъективных факторов.

Оценка знаний (академической успеваемости) магистранту осуществляется по 30 и 100 балльной системам (шкале) следующим образом:

| 30 балльная система | Баллы (Рейтинг) | Оценка по буквенной системе | Цифровой эквивалент оценки по GPA | Оценка по традиционной системе |
|---------------------|-----------------|-----------------------------|-----------------------------------|--------------------------------|
| 26- 30 | 87 – 100 | A | 4,0 | Отлично |
| 24- 25 | 80–86 | B | 3,33 | Хорошо |
| 22- 23 | 74–79 | C | 3,0 | |
| 20- 21 | 68–73 | D | 2,33 | Удовлетворительно |
| 18- 19 | 61–67 | E | 2,0 | |
| 9-17 | 31 -60 | FX | 0 | Неудовлетворительно |
| 0- 8 | 0-30 | F | 0 | |

1. Критерий оценки магистранта на лекционных занятиях Шкала оценивания 26 – 30 баллов:

- магистрант демонстрирует полное понимание темы согласно цели пройденной лекции;
- магистрант способен успешно обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

Шкала оценивания 24 – 25 баллов:

- магистрант демонстрирует полное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

Шкала оценивания 22 – 23 баллов:

- магистрант демонстрирует полное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

Шкала оценивания 20 – 21 баллов:

- магистрант демонстрирует неполное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

Шкала оценивания 18 – 19 баллов:

- магистрант демонстрирует неполное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет частично систематизировать, структурировать и аргументировать материал.

Шкала оценивания 9 – 17 баллов:

- магистрант не полностью понимает темы согласно цели пройденной лекции;
- магистрант не способен обосновывать свою точку зрения;
- не умеет систематизировать, структурировать и аргументировать материал.

Шкала оценивания 0– 18 баллов:

- магистрант непонимает темы согласно цели пройденной лекции;
- магистрант не способен обосновывать свою точку зрения;
- не умеет систематизировать, структурировать и аргументировать материал.

2. Критерий оценки магистранта на лабораторных занятиях

Шкала оценивания 26 – 30 баллов:

- магистрант демонстрирует полное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно выполняет поставленную задачу;
- уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 24 – 25 баллов:

- магистрант демонстрирует полное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно выполняет поставленную задачу;
- не полностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 22 – 23 баллов:

- магистрант демонстрирует неполное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно не может выполнить поставленную задачу;
- неуверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 20 – 21 баллов:

- магистрант демонстрирует неполное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно не может выполнить поставленную задачу;
- не полностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 18 – 19 баллов:

- магистрант демонстрирует неполное понимание цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- не полностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- частично соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 9 – 17 баллов:

- магистрант не понимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- неуверенно излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- несоблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 0 – 18 баллов:

- магистрант непонимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- не может изложить свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- несоблюдает правила техники безопасности при выполнении лабораторной работы.

3. Критерий оценки магистранта на самостоятельных работах магистрантов

Шкала оценивания 26 – 30 баллов:

- магистрант демонстрирует полное понимание проблемы;
- магистрант выполняет все требования, предъявляемые к заданию;
- уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 24 – 25 баллов:

- магистрант демонстрирует значительное понимание проблемы;
- магистрант выполняет все требования, предъявляемые к заданию;
- неполностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 22 – 23 баллов:

- магистрант демонстрирует частичное понимание проблемы;
- магистрант выполняет большинство требования, предъявляемые к заданию;
- неуверенно излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 20 – 21 баллов:

- магистрант демонстрирует небольшое понимание проблемы;
- магистрант выполняет немногие требования, предъявляемые к заданию;
- неполностью излагает свое решение при выполнении лабораторной работы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 18 – 19 баллов:

- магистрант демонстрирует непонимание проблемы;
- неполностью излагает свое решение при выполнении лабораторной работы;
- частично соблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 9 – 17 баллов:

- магистрант не понимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- неуверенно излагает свое решение при выполнении лабораторной работы;
- несоблюдает правила техники безопасности при выполнении лабораторной работы.

Шкала оценивания 0 – 18 баллов:

- магистрант непонимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- нет ответа, не было попытки решить задачу;
- несоблюдает правила техники безопасности при выполнении лабораторной работы.

14. Политика курса

Основные требования к компонентам курса и его изучению:

- Магистрант должен посещать занятия, принимать активное участие в работе группы при выполнении СРМ и на лабораторных занятиях.
- На лекционных занятиях делать записи содержания лекций, внимательно слушать, не нарушая дисциплину.

- На практическом занятии важно не только выступить, но и внимательно слушать своих сокурсников, оценивать их ответы, вести запись новой информации.
- Не опаздывать, в аудиторию входить до звонка.
- Отключать мобильные телефоны.
- Не перебивать преподавателя и своих сокурсников в ходе беседы или при чтении лекции.

IV. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ
РЕСПУБЛИКИ**

ОШСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

КАФЕДРА АСЦТ

«Согласована»

Председатель Методического совета
факультета МИТ доцент, канд.
п.и.к.: _____ Д. Зулпукарова

« ____ » _____ 2020 г.

«Утверждена»

на заседании кафедры АСЦТ
от 27 августа 2020 года, протокол №1
Зав. каф. АСЦТ, доцент

доцент
Молдоярлов У.Д. _____

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине: "Программирование Web-ориентированных
приложений" для магистрантов очного отделения, обучающихся
по направлению: 710100 «Информатика и вычислительная техника»

Составитель: доцент кафедры

Молдоярлов У.Д.

1. Общие положения

1.1. Область применения фонда оценочных средств (ФОС)

ФОС по дисциплине является частью программы подготовки специалистов среднего звена по специальности 710100 «Информатика и вычислительная техника»

2. Перечень компетенций и этапы их формирования

Компетенции обучающегося, формируемые в результате освоения дисциплины

1. ОК-2: способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности;
2. ПК-6: способен к постановке и модернизации отдельных лабораторных работ и практикумов по дисциплинам направления;
3. ПК-13: способен проектировать и применять инструментальные средства реализации программно-аппаратных проектов;

Этапы формирования компетенций

| | |
|---|-------------------|
| Тема 1. Введение в web-технологии: структура и принципы Web | ОК-2, ПК-13 |
| Тема 2. Клиент-серверные технологии Web. Протокол HTTP | ОК-2, ПК-6, ПК-13 |
| Тема 3. Сценарии и приложения, выполняющиеся на стороне клиента | ПК-13 |
| Тема 4. Серверные web-приложения | ОК-2, ПК-6 |
| Тема 5. Языки разработки сценариев Perl и PHP | ПК-6 |
| Тема 6. Введение в C# и платформу Visual Studio.Net | ПК-13 |
| Тема 7. Архитектура web-приложений ASP.NET. Разработка web-приложений на платформе .NET | ПК-6 |
| Тема 8. Интерфейсы взаимодействия web-приложений с СУБД | ОК-2, ПК-6, ПК-13 |
| Тема 9. Хостинг. Бесплатный хостинг. FTP. Размещение Интернет-ресурса на сервере провайдера. | ОК-2, ПК-6, ПК-13 |
| Тема 10. Организация процесса разработки web-контента. CMS/CMF | ОК-2, ПК-6, ПК-13 |

В результате освоения дисциплины «Программирование Web-ориентированных приложений» формируется компетенция ОК-2: способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности; ИК-4: способен к дальнейшему самостоятельному обучению; ПК-6: способен к постановке и модернизации отдельных лабораторных работ и практикумов по дисциплинам направления; ПК-9: способен разрабатывать проекты автоматизированных систем различного назначения, обосновывать выбор аппаратно-программных средств автоматизации и информатизации предприятий и организаций; ПК-13: способен проектировать и применять инструментальные средства реализации программно-аппаратных проектов;

Показатели, критерии и шкала оценивания компетенций текущего контроля знаний по учебной дисциплине «Программирование Web-ориентированных приложений»

Текущий контроль знаний, согласно положению о проведении текущего контроля

успеваемости и промежуточной аттестации обучающихся (далее Положение) в рамках изучения дисциплины «Программирование Web-ориентированных приложений» предполагает тестирование, выполнение заданий по лабораторным работам и выполнение заданий по практическим работам.

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ ЗНАНИЙ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ “ПРОГРАММИРОВАНИЕ WEB- ОРИЕНТИРОВАННЫХ ПРИЛОЖЕНИЙ”

2.1 Текущий контроль (ТК) № 1

Тема занятия: 1.2.1. Приемы масштабирования и распределения нагрузки

Метод и форма контроля: Письменный опрос (Опрос)

Вид контроля: Проверочная работа

Занятие(-я):

1.1.1. Основные понятия Web-разработки, Протокол HTTP

1.1.2. Основные понятия: Web-серверы и прокси-серверы

Задание №1

Приведите в соответствие термин и определение

| | |
|---|---|
| 1. Сервер, веб-сервер (<u>Web Server</u>) | a) любое информационно значимое наполнение сайта - тексты, графика, мультимедиа (видео и аудиофайлы, флэш-ролики и т.д.) |
| 2. Контент | b) определенная последовательность знаков, обозначающая имя сайта |
| 3. Доменное имя | c) стандартный язык создания веб-страниц в сети Интернет |
| 4. Браузер | d) компьютер, собранный из <u>специальных</u> комплектующих, на котором запущены и настроены специальные программы |
| 5. Хостинг | e) программное обеспечение для просмотра веб-сайтов, то есть для запроса веб |
| 6. HTML | f) услуга по предоставлению дискового пространства для физического размещения информации на сервере, постоянно находящемся в сети (обычно Интернет) |

| <i>Балл</i> | <i>Показатели оценки</i> |
|-------------|--------------------------------------|
| 5 | Приведены в соответствие 3 термина |
| 10 | Приведены в соответствие 5 терминов |
| 15 | Приведены в соответствие все термины |

Занятие(-я):

1.1.3. Статический и динамический сайт

Задание №1

Составьте сравнительную таблицу

| Название/ критерии сравнения | <u>Web-серверы</u> | прокси-серверы |
|------------------------------|--------------------|----------------|
| Определение | | |
| Особенности (назначение) | | |
| Примеры | | |

| <i>Балл</i> | <i>Показатели оценки</i> |
|-------------|---|
| 5 | Указаны определения. |
| | Функционально правильно представлены особенности |
| 10 | Указаны определения. |
| | Функционально правильно представлены особенности серверов (достоинства и недостатки). |
| 15 | Указаны определения. |
| | Функционально правильно представлены особенности серверов (достоинства и недостатки). |
| | Приведены примеры. |

Занятие(-я):

1.1.4. Технологии веб разработки

Задание №1

Продолжите фразы

1. Две основные (и связанные между собой) задачи масштабирования -это
2. Типичная архитектура сайта подразумевает
3. При небольших объемах данных и больших нагрузках применяю.....
4. При больших объемах данных необходимо распределить базу данных - разделить ее и
5. Балансировка (распределение запросов) может быть равномерная или с; с узлом, либо на стороне.....
6. Правильное сочетание методов позволит держать

НАПРИМЕР:

1. Две основные (и связанные между собой) задачи масштабирования - это распределение вычислений и распределение данных
2. Типичная архитектура сайта подразумевает разделение ролей и включает frontend, backend, базу данных и иногда хранилище файлов

Занятие(-я):

1.1.4. Технологии веб разработки

Задание №1

1. Сформулируйте определение Что такое web-приложение?
2. Схематично изобразите и опишите цикл обработки запроса к веб-приложению от клиента.
3. Для чего необходимы технологии разработки веб-приложений (такие как ASP.NET, PHP, Ruby On Rails и др.).

| <i>Балл</i> | <i>Показатели оценки</i> |
|-------------|---|
| 5 | Представлены 2 развернутых вопроса, приведены примеры web приложений и схематично представлены цикл обработки запроса к веб-приложению от клиента. |
| 10 | Представлены 3 развернутых вопроса, приведены примеры и схематично представлены цикл обработки запроса к веб- приложению от клиента, перечислены технологии разработки веб- приложений. |
| 15 | Представлены 3 развернутых вопроса, приведены примеры и схематично представлены цикл обработки запроса к веб- приложению от клиента, описаны технологии разработки веб- приложений. |

2.2 Текущий контроль (ТК) № 2

Тема занятия: 2.1.7.Разработка статических страниц

Метод и форма контроля: Практическая работа (Информационно-аналитический)

Вид контроля: Практическая работа по вариантам

Занятие(-я):

2.1.1.Введение в HTML. Инструментарий HTML

2.1.2.Структура документа HTML. Текст. Ссылки. 2.1.3.Якоря. Изображения. Списки.

2.1.4.Таблицы.

2.1.5. Фреймы. Валидация документов. 2.1.6.Разработка статических страниц Задание №1

Создать HTML-документ с нижеописанным содержимым (один из вариантов):

1. Создайте заголовок первого уровня и «отцентрируйте» его.
2. Создайте маркированный и нумерованный списки.
3. Создайте гипертекстовую ссылку (с последнего пункта нумерованного списка).
4. Поместите изображение на страницу.
5. Сделайте размещенное изображение ссылкой.
6. Еще раз разместите это же изображение на странице, и также сделайте его гипертекстовой ссылкой. Добейтесь, чтобы рамка вокруг изображения не была видна. Отцентрируйте изображение и текст под ним.
7. Сделайте задний фон страницы светло-желтым, цвет не посещенных ссылок – красным, посещенных – синим.
8. В нумерованном списке текст пункта 4 сделайте **жирным** а пункта 5 - курсивным

(наклонным)

9. Оформите в абзац текст, представленный ниже. Словосочетание «гипертекстовая ссылка» оформите зеленым цветом
10. Создайте абсолютную ссылку на сайт www.mail.ru

Пример:

Виды операционных систем:

1. Unix
2. Linux
3. FreeBSD
4. **OS/2**
5. MacOS
6. VMSP5
7. SunOS
8. VMESA
9. Windows

Активный контент:

1. Java
2. ActiveX
3. JavaScript
4. VBScript
5. Action
6. Script
7. Cookies

Пример:

Что такое гипертекстовая ссылка (или, как чаще говорят, просто ссылка)?

С помощью ссылки устанавливается связь с другим (или с тем же) документом. Ссылкой может быть фрагмент текста или изображение. По умолчанию текстовая ссылка подчеркнута. При подведении курсора мыши к гипертекстовой ссылке он превращается в раскрытую ладонь. При нажатии на ссылку браузер обращается к серверу, запрашивая документ, который указан в html-теге ссылки.

Пример:

Для поиска информации воспользуйтесь поисковой машиной www.mail.ru

| <i>Балл</i> | <i>Показатели оценки</i> |
|-------------|--------------------------|
| 5 | Выполнено 6 заданий |
| 10 | Выполнено 8 заданий |
| 15 | Выполнено 10 заданий |

2.3 Текущий контроль (ТК) № 3

Тема занятия: 2.2.10.Разработка статических страниц с использованием CSS

Метод и форма контроля: Практическая работа (Информационно-аналитический)

Вид контроля: Практическая работа

HTML, таблицы стилей CSS

Занятие(-я):

1.2.1.Приемы масштабирования и распределения нагрузки

2.1.2.Структура документа HTML. Текст. Ссылки.

2.1.3.Якоря. Изображения. Списки.

2.2.1. Введение в CSS. Преимущества стилей. Способы добавления стилей на страницу

2.2.2. Введение в CSS. Преимущества стилей. Способы добавления стилей на страницу

2.2.3. Базовый синтаксис CSS. Значения стилевых свойств. Селекторы тегов 2.2.4.Классы. Идентификаторы. Виды селекторов.
2.2.5.Псевдоклассы.Псевдоэлементы.

Задание №1

Ответьте на вопросы:

1. Перечислите варианты подключения к CSS стилям и приведите примеры.
2. Дайте определение: Что такое селектор? Что выступает в качестве селектора? Способ (пример) записи.
3. Перечислите виды селекторов.
4. Перечислите 5 псевдоклассов определяющие состояние элементов.
5. Напишите синтаксис использования псевдоэлементов.

| Балл | Показатели оценки |
|------|--------------------------------------|
| 5 | Раскрыты и представлены 3 вопроса |
| 10 | Раскрыты и представлены 4 вопроса |
| 15 | Раскрыты и представлены все вопроса. |

Занятие(-я):

- 2.1.7.Разработка статических страниц
- 2.2.1. Введение в CSS. Преимущества стилей. Способы добавления стилей на страницу
- 2.2.2. Введение в CSS. Преимущества стилей. Способы добавления стилей на страницу
- 2.2.3. Базовый синтаксис CSS. Значения стилевых свойств. Селекторы тегов
- 2.2.4. Классы. Идентификаторы. Виды селекторов.
- 2.2.5. Псевдоклассы.Псевдоэлементы.
- 2.2.6. Группирование.Наследование.Каскадирование. Валидация CSS
- 2.2.7. Идентификаторы и классы. Написание эффективного кода
- 2.2.8. Создание меню
- 2.2.9. Разработка статических страниц с использованием CSS

Задание №1

1. Создайте web документ, которые представляет текст, содержащий фрагменты, оформленные при помощи следующих свойств CSS:
 1. letter-spacing
 2. line-height
 3. text-align
 4. text-decoration
 5. text-indent
 6. text-transform
 7. vertical-align
 8. white-space
 9. word-spacing
2. Создайте web документ, которые демонстрирует использование следующих свойств CSS:
 1. background
 2. background-attachment
 3. background-color
 4. background-image
 5. background-position
 6. background-repeat

| Балл | Показатели оценки |
|------|---|
| 5 | Создано 2 документа соответственно заданию, использовано 4 свойства в каждом задании. |
| 10 | Создано 2 документа соответственно заданию, использовано 5 свойств в каждом задании. |

2.4 Текущий контроль (ТК) № 4**Тема занятия:** 2.3.6.Разработка статических страниц с использованием CSS3**Метод и форма контроля:** Практическая работа (Информационно-аналитический)**Вид контроля:** Практическая работа**Занятие(-я):**

2.3. Разработка статических страниц с использованием CSS

2.3.1.Введение в CSS3.

2.3.2. Введение в CSS3

2.3.3. Стиль CSS3: Фон. Цвет. Градиент.

2.3.4.Стиль CSS3: Шрифт. Текст. Прозрачность.

2.3.5. Стиль CSS3: Трансформация. Переходы. Анимация.

Задание №1

Создать документ содержащий элементы, оформленные при помощи стилей CSS:

1. Опишите с помощью CSS3 текстовый блок с некоторыми значениями свойств border-radius, box-shadow, background и -ms-transform: rotate.
2. Добавьте линейные градиенты с помощью CSS3.
3. Добавьте эффекты перехода используя функции:
 - linear
 - ease
 - ease-in
 - ease-out
 - ease-in-out
 - cubic-bezier(x,x,x,x)

| Балл | <i>Показатели оценки</i> |
|-------------|---|
| 5 | Представленный документ содержит все задания, освещающие пункты с CSS3, но не все свойства и функции. |
| 10 | Представленный документ содержит все задания, освещающие пункты с CSS3, но не функции. |
| 15 | Представленный документ содержит все задания, освещающие пункты с CSS3. |

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ РУБЕЖНОГО КОНТРОЛЯ

Рубежный контроль (РК) № 1

Метод и форма контроля: Практическая работа (Опрос)

Вид контроля: выполнить одно теоретическое задание (по вариантам) и одно практическое задание (по вариантам предметной области)

Задание №1

Ответить на вопросы:

1. Что такое Web приложение?
2. В чем особенность работы Web-сервера? Приведите примеры.
3. Перечислите и раскройте функции прокси-сервера.

| <i>Балл</i> | <i>Показатели оценки</i> |
|-------------|----------------------------------|
| 2 | Представлен ответ на 1 вопрос |
| 3 | Представлены ответы на 2 вопроса |
| 5 | Представлены ответы на 3 вопроса |

Задание №2

Ответить на вопросы:

1. Дайте определение масштабируемости
2. Что такое оптимизация и чем она отличается от масштабируемости
3. Перечислите способы масштабируемости их охарактерисуйте их.

| <i>Балл</i> | <i>Показатели оценки</i> |
|-------------|---|
| 2 | Представлено общее понятие масштабируемости. Перечислены виды. |
| 3 | Представлено общее понятие масштабируемости. Перечислены виды. |
| | Дана характеристика оптимизации и отличия ее от масштабируемости. Приведены примеры масштабируемости. |
| 5 | Представлено общее понятие масштабируемости. Перечислены виды. |
| | Дана характеристика оптимизации и отличия ее от масштабируемости. Приведены примеры масштабируемости. |
| | Как происходит масштабируемость БД. Дано определение механизма партиционирование, шардинг, масштабируемость ФС и масштабируемость кода (краткая характеристика) |

Задание №3

Выполнить задание:

1. Как сделать ссылку на телефонный номер, скайп или адрес электронной почты?
2. Как сделать гиперссылки на сайте? Приведите 5 способов с примерами.
3. Как создать таблицу, строки (ряды) таблицы и сделать ячейку заголовка столбца таблицы. Приведите примеры.

| <i>Балл</i> | <i>Показатели оценки</i> |
|-------------|--|
| 2 | Даны ответы на 2 вопроса, но без примеров. |
| 3 | Даны ответы на 2 вопроса с примерами. |
| 5 | Даны развернуты ответы на 3 вопроса с примерами. |

Рубежный контроль (РК) № 2

Метод и форма контроля: Практическая работа

Вид контроля: выполнить практическое задание

Задание №4

Создать HTML-документ, который будет содержать в себе базовую структуру, doctype, head, html, body.

Требования:

1. Вывести в созданном документе заглавия - h1, h2, h3

2. Вывести жирный, курсив и подчеркнутый текст
3. Вывести текст в абзаце
4. Вывести ссылки на внешний ресурс. Первая ссылка должна открываться в текущей вкладке браузера, а вторая ссылка, должна открывать внешний ресурс в новой вкладке.
5. Создать блочный и строчный тег (div, span)
6. Вывести таблицу, где в ней будет 5 столбцов: ID, имя, фамилия, отчество, дата рождения. Таблицу заполнить 2 записями
7. Создайте переменную name и присвойте ей значение, содержащее ваше имя, например "Василий"
8. Создайте переменную age и присвойте ей значение, содержащее ваш возраст, например 21.
9. Выведите с помощью echo фразу "Меня зовут: %ваше_имя%", например: "Меня зовут: Василий".
10. Выведите фразу "Мне %ваш_возраст% лет", например: "Мне 21 лет"
11. Напишите конструкцию if, которая выводит фразу: "Вам еще работать и работать" при условии, что значение переменной age попадает в диапазон чисел от 18 до 59 (включительно).
12. Расширьте конструкцию if из предыдущего пункта, выводя фразу: "Вам пора на пенсию" при условии, что значение переменной age больше 60.
13. Расширьте конструкцию if из предыдущего пункта, выводя фразу: "Вам еще рано работать" при условии, что значение переменной age попадает в диапазон чисел от 0 до 17 (включительно)
14. Расширьте конструкцию if из из предыдущего пункта, выводя фразу: "Неизвестный возраст" при условии, что значение переменной age является отрицательным числом, или вовсе числом не является.
15. Создать HTML-форму. Внутри формы создать поля для ввода: имени, фамилии, отчества, даты рождения, пароля и подтверждения пароля. Также, ко всем полям ввода нужно привязать метки label, в которых будет выведено имя поля. Метки должны находиться с левой стороны от полей ввода. Форма должна содержать поле-флаг, который будет нужен для подтверждения лицензионного соглашения и радио-кнопки для выбора гендерной принадлежности. Внизу формы должна быть кнопка для отправки введенных данных.
16. Вывести список элементов, построчно: яблоко, груша, персик. Список должен быть нумерованным.
17. Вывести выпадающий список элементов: яблоко, груша, персик. По умолчанию должна быть выбрана груша.
18. Нарисовать горизонтальную линию с шириной линии 150px. Ширину задать с помощью CSS правил (в атрибуте style).
19. Вывести изображение, которое должно быть сжато до 200px по ширине. Также у картинки должен быть заполнен атрибут альтернативного имени.

| <i>Балл</i> | <i>Показатели оценки</i> |
|-------------|---|
| 1 | Выполнено 9 требований в задании на выбор. |
| 3 | Выполнено 13 требований в задании на выбор. |
| 6 | Выполнено все требования в задании. |

Задание №5 (из текущего контроля)

1. Напишите правильно способы подключения JavaScript, заполнив пропуски. Приведите примеры.
 1. `cript>`
 2. `cript>`
2. Для чего используются следующие функции и метод, заполните пропуски:
 - `document.write()` для.....
 - `document.writeln()` для
 - `alert()` для
3. Ответьте на вопросы
 1. Перечислите 3 требования к Идентификаторам (identifiers) — имена переменных, методов и объектов:

2. Перечислите функции преобразования данных.
3. Какие методы для вывода модальных окон в javascript Вы знаете?
4. Какой метод позволяет вывести модальное окно для ввода данных?

| Балл | Показатели оценки |
|-------------|---|
| 2 | Даны ответы на 4 вопроса |
| 3 | Даны ответы на 4 вопроса и заполнены пропуски в функциях и методах |
| 5 | Даны ответы на 4 вопроса и заполнены пропуски в функциях и методах и способах подключения. Приведены примеры в пером задании. |

Задание №6 (из текущего контроля)

Сформулируйте ответы по "Безопасность Web- приложений"

1. Что такое SQL инъекции?
2. На какие два вида делятся HTML инъекции?
3. Перечислите 22 вида уязвимостей веб сайтов.
4. Дайте характеристику 10 видам уязвимостей веб сайтов.
5. Назовите виды сетевых атак.

| Балл | Показатели оценки |
|-------------|---|
| 2 | Сформулированы ответы на три вопроса на выбор. |
| 3 | Сформулированы ответы на четыре вопроса на выбор, включая обязательно четвертый вопрос. |
| 5 | Сформулированы все ответы, приведены примеры. |

Карта накапливаемости баллов по дисциплине

I модуль

| Текущий контроль | РК1 | M1=(TK1+TK2+PK1)/4 |
|-------------------------|------------|---------------------------|
| (TK1+TK2)/2 | | |
| 30 балл | 30 балл | 30 балл |

II модуль

| Текущий контроль | РК2 | M2=(TK3+TK4+PK2)/4 |
|-------------------------|------------|---------------------------|
| (TK3+TK4)/2 | | |
| 30 балл | 30 балл | 30 балл |

3. Критерии оценки на экзамене

Выставление оценок на экзаменах осуществляется на основе принципов объективности, справедливости, всестороннего анализа качества знаний магистрантов, и других положений, способствующих повышению надежности оценки знаний обучающихся и устранению субъективных факторов.

В соответствии с действующими нормативными актами и рекомендациями Министерства образования и науки КР устанавливаются следующие критерии выставления оценок на экзаменах по гуманитарным, естественным, техническим и другим дисциплинам:

- оценка **"отлично"** выставляется магистранту, который обнаружил на экзамене всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, который усвоил основную литературу и ознакомился с дополнительной литературой, рекомендованной программой. Как правило, оценка "отлично" выставляется магистрантам, усвоившим взаимосвязь основных понятий дисциплины и их значений для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала;

- оценка **"хорошо"** выставляется магистранту, который на экзамене обнаружил полное знание учебно-программного материала, успешно выполнил предусмотренные в программе задания, усвоил основную литературу, рекомендованную в программе. Как правило, оценка "хорошо" выставляется магистрантам, показавшим систематический характер знаний по дисциплине и способным к их самостоятельному выполнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности;

- оценка **"удовлетворительно"** выставляется магистранту, обнаружившему знание основного учебного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справляющемуся с выполнением заданий, предусмотренных программой, который ознакомился с основной литературой, рекомендованной программой. Как правило, оценка "удовлетворительно" выставляется магистрантам, допустившим погрешности в ответе на экзамене и при выполнении экзаменационных заданий, но обладающим необходимыми знаниями для их устранения под руководством преподавателя;

- оценка **"неудовлетворительно"** выставляется магистранту, обнаружившему пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий, не ознакомившемуся с основной литературой, предусмотренной программой, и не овладевшему базовыми знаниями, предусмотренными по данной дисциплине и определенными соответствующей программой курса (перечень основных знаний и умений, которыми должны овладеть магистранты, является обязательным элементом рабочей программы курса).

V. ГЛОССАРИЙ (СЛОВАРЬ)

Апплет

Апплет — это программный компонент (программный код), который загружается браузером и реализует определенные функции на веб-странице. Апплеты Java обеспечивают интерактивные функции в веб-браузерах с помощью Java Virtual Machine (JVM).

Установить/Выбрать

Один из методов, с помощью которого пользователи вводят данные в диалоговом окне или на веб-странице, либо установив флажок с помощью курсора, либо переместив курсор на графическую селективную кнопку и щелкнув на ней.

Диалоговое окно

Специальное окно в графическом интерфейсе пользователя, которое создает уведомления для пользователя определенного ресурса или запрашивает входные данные.

Включить/Отключить

Доступные для пользователя настройки программного обеспечения, которые включают или отключают определенные функции или возможности. Используется для устранения ошибок конфигурации Java, когда пользователям может потребоваться использовать диалоговое окно для включения или отключения определенной настройки или функции.

Брандмауэр

Интернет-брандмауэры могут быть как программными, так и аппаратными. Они защищают компьютер от внешних интернет-атак, которые могут представлять угрозу интернет-безопасности и повредить файлы на компьютере. В брандмауэре можно задавать правила, которые разрешают или запрещают передачу интернет-трафика.

Архив Java (JAR)

Архив Java (.jar) — это формат файла, используемый для объединения нескольких файлов в один архивный файл. Как правило, JAR-файл содержит файлы класса и вспомогательные ресурсы, ассоциированные с апплетами и приложениями.

Подключаемый модуль Java

Технология подключаемого модуля Java является частью среды выполнения Java, которая подключает популярные браузеры к платформе Java. С помощью такого подключения возможен запуск апплетов на веб-сайтах и их отображение в окне установленного на компьютере браузера.

Кэш среды выполнения Java (JRE)

Область хранения на консоли Java, которую иногда необходимо очищать вручную, чтобы обеспечить возможность загрузки и установки последней версии Java.

Виртуальная машина Java (JVM)

В Java набор программ, обеспечивающих поддержку выполнения инструкций, обычно написанных на Java. JVM доступны для большинства аппаратных и программных платформ.

javaw.exe

Процесс javaw.exe - программа, принадлежащая корпорации Oracle, которая используется в браузере Internet Explorer в качестве подключаемого модуля Java. Она является аналогом программы java.exe. Единственным отличием является то, что при работе процесса javaw.exe не отображается окно консоли. Если вы не хотите, чтобы отображалось окно командной строки, вполне вероятно, что может использоваться процесс javaw.exe. javaw.exe является исполняемым файлом, отображающим диалоговое окно при сбоях запуска программы.

jucheck.exe

Процесс jucheck.exe устанавливается вместе с Java на компьютер под управлением Windows и используется для проверки новых версий Java. Процесс не устанавливает Java самостоятельно, а только предупреждает о доступных для загрузки новых версиях. » Дополнительная информация

IFTW

Установка через Интернет, или интерактивная установка.

Процедуры установки

Пользователи Java могут выбрать наиболее подходящую из трех процедур установки – интерактивную, автономную или ручную.

Интерактивная установка выполняется автоматически при наличии подключения к Интернету после нажатия кнопки "Загрузить Java бесплатно" на странице Загрузить Java бесплатно.

Для установки в автономном режиме необходимо загрузить исполняемый файл, доступный на странице ручной загрузки Java где можно найти все файлы, необходимые для полной установки среды по усмотрению пользователя. Во время данной процедуры установки подключение к Интернету не требуется. Данный файл также можно скопировать и установить на другой компьютер, не подключенный к Интернету.

Во время установки вручную загружается исполняемый программный файл IFTW (Install From The Web). Такой режим установки требует минимального участия пользователя. При запуске этой программы все необходимые файлы загружаются из Интернета, поэтому во время данной процедуры установки требуется постоянное подключение к Интернету.

Манифест

Манифест — это специальный файл, который может содержать информацию о файлах, упакованных в JAR-файле.

Операционная система

Общий термин, обозначающий программное обеспечение, которое используется для управления основными задачами ресурсов компьютера и которое используется программистами для получения доступа к этим ресурсам. Наиболее

распространенные настольные операционные системы включают в себя Linux, Mac OS X, Microsoft Windows и Solaris.

Параметр

В программировании параметром называется значение, которое передается функции. Функция либо использует параметр в соответствующей задаче, либо выполняет операцию с параметром.

Физическая память

Наиболее часто на сайте java.com этот термин указывает на форму полупроводниковой памяти на компьютерах, которая называется оперативной памятью (ОЗУ).

Процесс

Общий термин, который используется для описания программы, запущенной для выполнения определенной задачи – часто одновременно с другими программами.

Прокси-сервер

Компьютер, выступающий посредником при подключении пользователя к сети Интернет. Он используется для ведения журнала сеансов подключения к Интернету и для блокирования доступа к отдельным веб-сайтам. Брандмауэр, устанавливаемый на прокси-сервере, может по разным причинам блокировать отдельные веб-сайты и веб-страницы. По этой причине пользователи могут испытывать затруднения при попытке загрузить Java или запустить те или иные Java-апплеты без настройки параметров определенного прокси-сервера в веб-браузере компьютера.

Настройки прокси-сервера

Правильно настроенные параметры прокси-сервера обеспечивают для пользователей возможность подключения к сети Интернет при использовании прокси-сервера. Как правило, пользователям необходимо связаться со своими администраторами сети для получения необходимых сведений для настройки параметров прокси-сервера.

Кэш веб-браузера

Кэшем называется область временного хранения, где можно сохранить часто используемые данные для быстрой загрузки. Кэш веб-браузера содержит копии страниц, просматриваемых пользователем. Для устранения определенных проблем установки или настройки Java иногда требуется выполнять очистку кэша вручную путем обращения к диалоговому окну.

Временное решение

Временным решением обычно называется решение, которое позволяет обойти определенную проблему в системе до появления решения, которое позволит окончательно устранить эту ошибку.

Безопасность

Список сайтов-исключений

Функция "Exception Site List" позволяет конечным пользователям выполнять

апплеты Java и приложения Java Web Start, которые не соответствуют последним требованиям к безопасности. RIA-приложения, которые размещаются на сайте из списка исключений, разрешается выполнять с применимыми запросами безопасности.

Базовый уровень безопасности

Базовый уровень безопасности — это минимальное рекомендуемое обновление для Java. Пользователям не рекомендуется запускать версии Java, уровень которых ниже последнего базового уровня безопасности. Каждая основная версия Java (например, Java 6, Java 7) может иметь свое собственное обновление базового уровня безопасности.

Исправление

Означает дополнительные изменения для установленного программного обеспечения. Может включать в себя исправления проблем, связанных с общей производительностью и системой безопасности ПО.

Критическое обновление (CPU)

Критические обновления представляют собой регулярно выпускаемые и общедоступные наборы исправлений системы безопасности, предназначенные для снижения затрат клиентов, связанных с управлением безопасностью.

Предупреждение безопасности

В случае крайней необходимости применения исправлений до выхода следующего пакета критических обновлений, Oracle выпускает предупреждение безопасности (т.е. исправление системы безопасности, распространяемое ранее очередного выпуска критического обновления исправлений).

Общие уязвимости и внешние воздействия (CVE)

Номера CVE являются уникальными стандартными идентификаторами для определения общеизвестной информации об уязвимостях безопасности. Программа CVE спонсируется отделом обеспечения кибербезопасности и связи департамента США по национальной безопасности и управляется корпорацией MITRE.

Центр сертификации (CA)

Центром сертификации называется доверенная третья сторона - обычно, коммерческая компания, - которая выпускает цифровые сертификаты. Сертификаты выдаются организациям и частным лицам после процесса удостоверения подлинности. Цифровой сертификат добавляется к компьютерным приложениям для подтверждения того, что приложение выпущено владельцем сертификата.

Доверенный сертификат

Если у приложения имеется сертификат доверенного центра сертификации, отображается синий значок щита вместо желтых предупреждающих сигналов (щита или значка треугольника). Удостоверьтесь, что приложение получено от надежного издателя, при помощи сертификата, выпущенного доверенным центром сертификации.

Самоподписанный сертификат

Самоподписанные сертификаты не удостоверяются доверенным центром сертификации. Использование приложений этого типа обеспечивает самый высокий уровень угрозы безопасности из-за отсутствия удостоверения личности издателя.

Приложение "из песочницы"

Обозначает уровень доступа к системным файлам, таким как файлы на жестком диске, и к сети. Песочница обеспечивает защиту от доступа недоверенных и, возможно, вредоносных приложений к ресурсам системы.

Подписанные и неподписанные приложения

Подписанный код - это технология безопасности, при использовании которой приложения снабжаются цифровой подписью, удостоверяющей автора или издателя программного обеспечения. Подписанными приложениями называются такие приложения, которые были подписаны поставщиком приложения (самоподписанные) или центром сертификации. Это не является гарантией того, что код является доверенным, а означает только то, что приложение получено от указанного источника. Неподписанными приложениями называются такие приложения, которые не были подписаны, поэтому их источник не может быть проверен.

VI. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ МАГИСТРАНТОВ ПО ИЗУЧЕНИЮ ДИСЦИПЛИНЫ

6.1. Изучение программы курса

На лекциях преподаватель рассматривает контрольные вопросы курса, составленной в соответствии с государственным образовательным стандартом. Из-за недостаточного количества аудиторных часов некоторые темы не удастся осветить в полном объеме, в следствии этого эти контрольные вопросы выносятся на самостоятельное изучение магистрантами, с рекомендациями той или иной литературы и с предоставлением ссылок на компьютерные онлайн источники:

1. <https://startandroid.ru/ru/uroki/vse-uroki-spiskom.html>
2. <https://itproger.com/course/java-android>

Кроме этого, для лучшего освоения материала и систематизации знаний по дисциплине, необходимо постоянно разбирать материалы лекций по конспектам и учебным пособиям. В случае необходимости обращаться к преподавателю за консультацией. Полный список литературы по дисциплине приведен в конце комплекса.

6.2. Контрольные вопросы

После изучения некоторых разделов практической части курса проводятся контрольные аудиторные работы в различных формах (презентации в группах, в парах, в письменных работах и др.). Готовиться к контрольным работам нужно по материалам лекций и рекомендованным источникам.

Примерный перечень контрольных вопросов приведен в рабочей программе.

Также в конце каждой лабораторной работы необходимо проводить письменный анализ выполненных задач по алгоритмизации и программированию, а также заполнять глоссарий в тетради.

Для итогового оценивания в учебно-методическом комплексе приведен перечень примерных контрольных вопросов и СРМ по которым составляются билеты экзамена.

Структура экзаменационных билетов ориентирована на такие навыки, как умение, способность и владение, то есть отражают обретенные компетенции.

6.3. Лабораторные работы

Знания, умения и навыки, приобретенные в результате освоения практикума, существенно помогут магистрантам при дальнейшем обучении и обобщении навыков программирования.

Структура лабораторного практикума включает в себя комплекс работ, скомпонованных по принципу от простого к сложному, даются примеры работ и рекомендации по выполнению. В лабораторных работах дается подробное описание команд, их опций и указана последовательность действий при решении определенных задач, что позволяет магистранту практически самостоятельно освоить команды, функции и процедуры, которые используются при алгоритмизации и программировании поставленных задач.

Для закрепления каждой пройденной темы приводятся описания лабораторных

работ. При изучении курса необходимо своевременно выполнять и сдавать преподавателю выполненные задания. В учебном процессе магистрантами широко используется учебно – методический комплекс с теоретическим содержанием, лабораторными описаниями и контрольными вопросами по данному курсу.

При изучении дисциплины магистранты должны выполнить лабораторные работы по темам, оговоренным рабочей программой.

На лабораторных работах магистранты осваивают работу с программным обеспечением путем своевременного выполнения индивидуальных заданий и ответов на контрольные. Устный опрос по каждой теме. Лабораторные задания необходимо выполнять в строгой последовательности, так как каждое последующее задание является обобщением предыдущей.

Также для более углубленного изучения и практического освоения принципов алгоритмизации, программирования и управления базами данных, магистрантам по каждой пройденной теме даются задания для самостоятельного выполнения. Самостоятельные задания включают в себя обобщающие задачи и проекты.

Своевременной называется исполнение лабораторной работы в течении недели с момента предоставления задания по плану занятий. По результатам выполнения каждой лабораторной работы магистранту выставляется балл.

6.4. Самостоятельная работа магистрантов

Самостоятельная работа магистрантов включает в себя следующие части:

Для более углубленного изучения и практического освоения программы, магистрантам по каждой пройденной теме даются задания для самостоятельного выполнения. Самостоятельные задания включают в себя обобщающие задания и проекты. Задания даются для индивидуального выполнения, а также в группах. При работе в группах, первая группа, завершившая практическое задание получает поощрительный накопительный балл. Данные баллы учитываются при итоговом оценивании. Группа не выполнившая задание получает минусовой балл. Данный вид самостоятельной работы выполняется своевременно в течении семестра и принимается в ходе практических занятий и во время дежурств преподавателя.

По итогам пройденных материалов и дополнительно изученных сведений, магистрантами выполняется объемная проектная работа по алгоритмизации и программированию с включением базы данных. Проект выполняется по всем установленным правилам и техникам, с проведением технического анализа, выбором наиболее оптимальных методов и приемов. Результаты работ принимаются и оцениваются в I и II модульном контроле (30+30 баллов).

Все виды самостоятельных работ магистрантов охватывают весь лекционный курс, а также материалы и сведения не включенные в аудиторный курс.

В ходе работы над проектом, магистрант имеет право получать консультацию по выполнению у преподавателя во время дежурств.

VII. ЛЕКЦИОННЫЕ МАТЕРИАЛЫ

№1. Лекция. Введение в web-технологии: структура и принципы Web

Аннотация: Интернет: понятие, история развития. Стандартизация в Интернет. RFC-документы. Стек протоколов TCP/IP. Система доменных имен DNS. Структура и принципы WWW. Прокси-серверы. Протоколы Интернет прикладного уровня. (ОК-2, ПК-13)

План лекции:

- 1.1. Развитие Интернет
- 1.2. Стандартизация в интернет
- 1.3. Стек протоколов TCP/IP
- 1.4. DNS – система доменных имен
- 1.5. World Wide Web (WWW)

Развитие Интернет

Интернет — всемирная система объединенных компьютерных сетей, построенная на базе IP и маршрутизации IP-пакетов. Интернет образует глобальное информационное пространство, служит физической основой для Всемирной паутины (World Wide Web, WWW) и множества других систем передачи данных.

Интернет можно рассматривать как «сеть сетей», каждая из которых управляется независимым оператором – поставщиком услуг Интернета (ISP, Internet Service Provider – интернет провайдер).

Сегодняшний Интернет обязан своему появлению объединенной сети ARPANET, которая начиналась как скромный эксперимент в новой тогда технологии коммутации пакетов (табл. 1). Сеть ARPANET была развернута в 1969 г. и состояла поначалу всего из четырех узлов с коммутацией пакетов, используемых для взаимодействия нескольких хостов и терминалов. Первые линии связи, соединявшие узлы, работали на скорости всего 50 Кбит/с. Сеть ARPANET финансировалась управлением перспективного планирования научно-исследовательских работ ARPA (Advanced Research Projects Agency) министерства обороны США и предназначалась для изучения технологии и протоколов коммутации пакетов, которые могли бы использоваться для кооперативных распределенных вычислений.

Таблица 1. *Хронология развития Интернета (с 1966 по 2000 г.)*

| Год | Событие |
|------|---|
| 1966 | Эксперимент с коммутацией пакетов управления ARPA |
| 1969 | Первые работоспособные узлы сети ARPANET |
| 1972 | Изобретение распределенной электронной почты |
| 1973 | Первые компьютеры, подключенные к сети ARPANET за пределами США |
| 1975 | Сеть ARPANET передана в ведение управления связи министерства обороны США |
| 1980 | Начинаются эксперименты с TCP/IP |
| 1981 | Каждые 20 дней к сети добавляется новый хост |
| 1983 | Завершен переход на TCP/IP |
| 1986 | Создана магистраль NSFnet |
| 1990 | Сеть ARPANET прекратила существование |
| 1991 | Появление Gopher |

| | |
|------|--|
| 1991 | Изобретение Всемирной паутины. Выпущена система PGP. Появление Mosaic |
| 1995 | Приватизация магистрали Интернета |
| 1996 | Построена магистраль OC-3 (155 Мбит/с) |
| 1998 | Число зарегистрированных доменных имен превысило 2 млн. |
| 2000 | Количество индексируемых web-страниц превысило 1 млрд. |

На рисунке 1.1 представлен график, показывающий динамику роста числа хостов (как формально зарегистрированных, так и активно функционирующих).

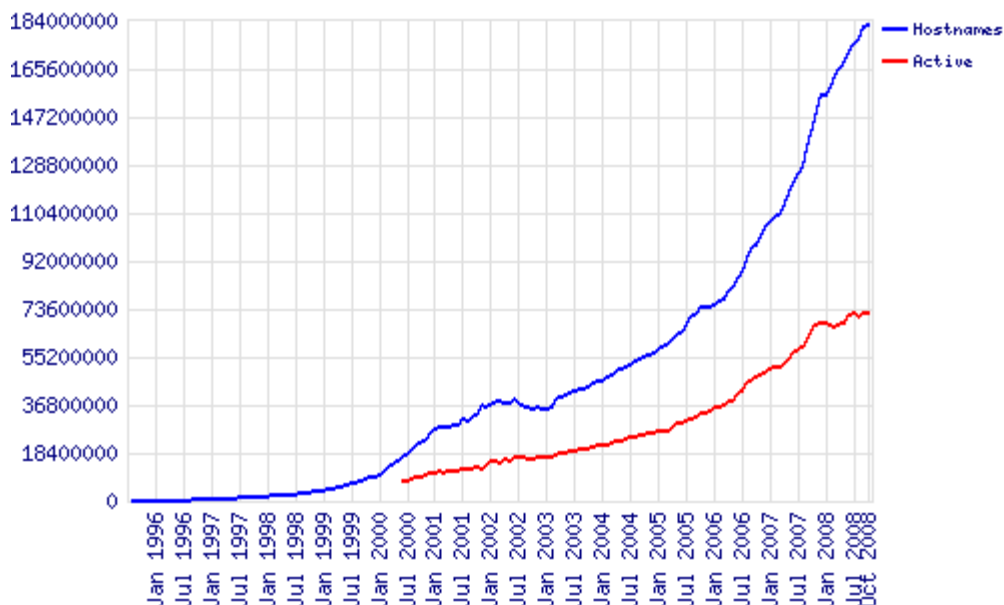


Рис. 1.1. Динамика роста числа хостов в Интернет (взято с сайта www.netcraft.com).

Возможно ли централизованное управление в такой глобальной сети? Ответ на данный вопрос будет отрицательным, поскольку, во-первых, данная сеть является транснациональной и, во-вторых, в силу исторических предпосылок ее формирования.

Тем не менее, в Интернете могут проявляться опосредованные формы централизации в форме единой технической политики, согласованном наборе технических стандартов, назначении имен и адресов компьютеров и сетей, входящих в Интернет.

То есть Интернет является децентрализованной сетью, что имеет свои достоинства и недостатки.

1. Достоинства:

- Легкость наращивания Интернета.

2. Недостатки:

- Сложность модернизации технологий и услуг Интернета, поскольку требуются согласованные усилия всех поставщиков услуг.
- Невысокая надежность услуг Интернета.
- Ответственность за работоспособность отдельных сегментов этой сети возлагается на поставщиках услуг Интернета.

Существуют различные типы поставщиков услуг Интернета:

- просто поставщик услуг Интернета выполняет транспортную функцию для конечных пользователей – передачу их трафика в сети других поставщиков услуг Интернета;
- поставщик интернет-контента имеет собственные информационно- справочные

ресурсы, предоставляя их содержание в виде web-сайтов;

- поставщик услуг хостинга предоставляет свои помещения, каналы связи и серверы для размещения внешнего контента;
- поставщик услуг по доставке контента занимается только доставкой контента в многочисленные точки доступа с целью повышения скорости доступа пользователей к информации;
- поставщик услуг по поддержке приложений предоставляет клиентам доступ к крупным универсальным программным продуктам, например SAP R3;
- поставщик биллинговых услуг обеспечивает оплату счетов по Интернету;

Стандартизация в интернет

Интернет является очень сложной сетью, и соответственно такой же сложной является задача организации взаимодействия между устройствами сети. Для решения такого рода задач используется *декомпозиция*, т.е. разбиение сложной задачи на несколько более простых задач-модулей. Одной из концепций, реализующих декомпозицию, является многоуровневый подход. Такой подход дает возможность проводить разработку, тестирование и модификацию каждого отдельного уровня независимо от других уровней. *Иерархическая декомпозиция* позволяет, перемещаясь в направлении от более низких к более высоким уровням переходить к более простому представлению решаемой задачи.

Специфика многоуровневого представления сетевого взаимодействия состоит в том, что в процессе обмена сообщениями участвуют как минимум две стороны, для которых необходимо обеспечить согласованную работу двух иерархий аппаратно-программных средств. Каждый из уровней должен поддерживать интерфейс с выше- и нижележащими уровнями собственной иерархии средств и интерфейс со средствами взаимодействия другой стороны на том же уровне иерархии. Данный тип интерфейса называется протоколом (см. рисунок 2).

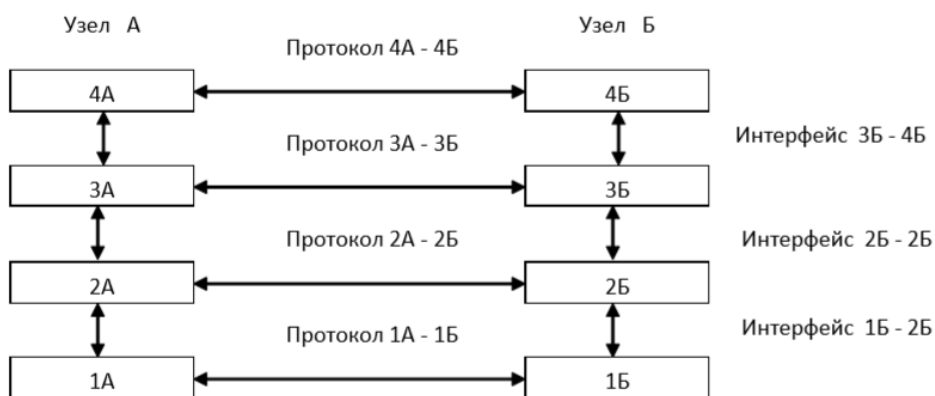


Рисунок 2. Организация взаимодействия между уровнями иерархии при иерархической декомпозиции в сети Интернет.

Иерархически организованный набор протоколов, достаточный для организации взаимодействия узлов в сети, называется *стеком протоколов*.

В начале 80-х годов международные организации по стандартизации ISO (International Organization for Standardization), ITU (International Telecommunications Union) и другие разработали стандартную модель взаимодействия открытых систем OSI (Open System Interconnection). Назначение данной модели состоит в обобщенном представлении средств сетевого взаимодействия. Ее также можно рассматривать в качестве универсального языка сетевых специалистов (*справочной модели*).

Поскольку сеть – это соединение разнородного оборудования, актуальной является проблема совместимости, что в свою очередь, требует согласования всеми производителями общепринятых стандартов. *Открытой* является система, построенная в соответствии с *открытыми*

спецификациями.

Спецификация представляет собой формализованное описание аппаратных (программных) компонентов, способов их функционирования, взаимодействия с другими компонентами, условий эксплуатации, особых характеристик. Под *открытыми спецификациями* понимаются опубликованные, общедоступные спецификации, соответствующие стандартам и принятые в результате достижения согласия после всестороннего обсуждения всеми заинтересованными сторонами. Использование открытых спецификаций при разработке систем позволяет третьим сторонам разрабатывать для этих систем аппаратно-программные средства расширения и модификации, а также создавать программно-аппаратные комплексы из продуктов разных производителей.

Если две сети построены с соблюдением принципов открытости, это дает следующие преимущества:

- Возможность построения сети из аппаратных и программных средств различных производителей, придерживающихся стандарта;
- Безболезненная замена отдельных компонентов сети другими, более совершенными;
- Легкость сопряжения одной сети с другой.

В рамках модели OSI средства взаимодействия делятся на семь уровней: прикладной, представления, сеансовый, транспортный, сетевой, канальный и физический. В распоряжение программистов предоставляется прикладной программный интерфейс, позволяющий обращаться с запросами к самому верхнему уровню, а именно, - уровню приложений.

Сеть Интернет строилась в полном соответствии с принципами открытых систем. В разработке стандартов этой сети принимали участие тысячи специалистов-пользователей сети из вузов, научных организаций и компаний. Результат работы по стандартизации воплощается в документах *RFC*.

RFC (англ. Request for Comments) — документ из серии пронумерованных информационных документов Интернета, содержащих технические спецификации и Стандарты, широко применяемые во Всемирной сети. В настоящее время первичной публикацией документов *RFC* занимается *IETF* под эгидой открытой организации Общество Интернет (*ISOC*). Правами на *RFC* обладает именно Общество Интернет. Формат *RFC* появился в 1969 г. при обсуждении проекта *ARPANET*. Первые *RFC* распространялись в печатном виде на бумаге в виде обычных писем, но уже с декабря 1969 г., когда заработали первые сегменты *ARPANET*, документы начали распространяться в электронном виде. В таблице 2 приведены некоторые из наиболее известных документов *RFC*.

Таблица 2. Примеры популярных *RFC*-документов.

| Номер <i>RFC</i> | Тема |
|------------------|--|
| RFC 768 | UDP |
| RFC 791 | IP |
| RFC 793 | TCP |
| RFC 822 | Формат электронной почты, заменен RFC 2822 |
| RFC 959 | FTP |
| RFC 1034 | DNS — концепция |
| RFC 1035 | DNS — внедрение |
| RFC 1591 | Структура доменных имен |

| | |
|----------|-------------------------------------|
| RFC 1738 | URL |
| RFC 1939 | Протокол POP версии 3 (POP3) |
| RFC 2026 | Процесс стандартизации в Интернете |
| RFC 2045 | MIME |
| RFC 2231 | Кодировка символов |
| RFC 2616 | HTTP |
| RFC 2822 | Формат электронной почты |
| RFC 3501 | IMAP версии 4 издание 1 (IMAP4rev1) |

Основным организационным подразделением, координирующим работу по стандартизации Интернет, является *ISOC* (Internet Society), объединяющее порядка 100 тысяч участников, которые занимаются различными аспектами развития данной сети. *ISOC* курирует работу *IAB* (Internet Architecture Board), включающую две группы:

- *IRTF* (Internet Research Task Force). Координирует долгосрочные исследовательские проекты, относящиеся к TCP/IP;
- *IETF* (Internet Engineering Task Force). Инженерная группа, определяющая спецификации для последующих стандартов Интернет.

Разработкой стандартов для сети Web, начиная с 1994 года, занимается Консорциум W3C (World Wide Web Consortium), основанный и до сих пор возглавляемый Тимом Бернерсом-Ли.

Консорциум W3C — организация, разрабатывающая и внедряющая технологические стандарты для Интернета и WWW. Миссия W3C формулируется следующим образом: «Полностью раскрыть потенциал Всемирной паутины путём создания протоколов и принципов, гарантирующих долгосрочное развитие Сети». Две другие важнейшие задачи Консорциума — обеспечить полную «интернационализацию Сети» и сделать ее доступной для людей с ограниченными возможностями.

W3C разрабатывает для WWW единые принципы и стандарты, называемые «*Рекомендациями*», которые затем внедряются разработчиками программ и оборудования. Благодаря *Рекомендациям* достигается совместимость между программными продуктами и оборудованием различных компаний, что делает сеть WWW более совершенной, универсальной и удобной в использовании.

Все *Рекомендации* W3C открыты, то есть, не защищены патентами и могут внедряться любым человеком без каких-либо финансовых отчислений Консорциуму.

Для удобства пользователей Консорциумом созданы специальные *программы-валидаторы* (англ. Online Validation Service), которые доступны по сети и могут за несколько секунд проверить документы на соответствие популярным *Рекомендациям* W3C. Консорциумом также созданы многие другие утилиты для облегчения работы web-мастеров и программистов. Большинство утилит — это программы с открытым исходным кодом, все они бесплатные. В последнее время, повинуясь мировым тенденциям, Консорциум, в целом, гораздо больше внимания уделяет проектам с открытым исходным кодом.

Стек протоколов TCP/IP

Эти протоколы изначально ориентированы на глобальные сети, в которых качество соединительных каналов не идеально. Он позволяет создавать глобальные сети, компьютеры в которых соединены друг с другом самыми разными способами от высокоскоростных оптоволоконных кабелей и спутниковых каналов до коммутируемых телефонных линий. TCP/IP соответствует модели OSI достаточно условно и содержит 4 уровня. Прикладной

уровень стека соответствует трем верхним уровням модели OSI: прикладному, представления и сеансовому.

В сети данные всегда передаются блоками относительно небольшого размера. Каждый блок имеет префиксную часть (заголовок), описывающую содержимое блока, и суффиксную, содержащую, например, информацию для контроля целостности передаваемого блока данных.

Название стека протоколов TCP/IP состоит из названий двух разных протоколов. Протокол IP (Internet Protocol) представляет собой протокол нижнего (сетевого) уровня и отвечает за передачу пакетов данных в сети. Он относится к так называемым протоколам *датаграмм* и работает без подтверждений. Последнее означает, что при его использовании доставка пакетов данных не гарантируется и не подтверждается. Не гарантируется также и то, что пакеты достигнут пункта назначения в той последовательности, в которой они были отправлены.

К протоколам сетевого уровня относится также протокол межсетевых управляющих сообщений *ICMP* (Internet Control Message Protocol), предназначенный для передачи маршрутизатором источнику информации об ошибках при передаче пакета.

Очевидно, что намного удобнее передавать данные по каналу, который работает корректно, доставляя все пакеты по порядку. Поэтому поверх протокола IP работает протокол передачи данных более высокого (транспортного) уровня — TCP (Transmission Control Protocol). Посылая и принимая пакеты через протокол IP, протокол TCP гарантирует доставку всех переданных пакетов данных в правильной последовательности.

Следует отметить, что при использовании протокола IP обеспечивается более быстрая передача данных, так как не тратится время на подтверждение приема каждого пакета. Есть и другие преимущества. Одно из них заключается в том, что он позволяет рассылать пакеты данных в широковещательном режиме, при котором они достигают всех компьютеров физической сети. Что же касается протокола TCP, то для передачи данных с его помощью необходимо создать канал связи между компьютерами. Он и создается с использованием протокола IP.

Для идентификации сетевых интерфейсов используются 3 типа адресов:

- аппаратные адреса (или MAC-адреса);
- сетевые адреса (IP-адреса);
- символьные (доменные) имена.

В рамках IP протокола для создания глобальной системы адресации, не зависящей от способов адресации узлов в отдельных сетях, используется пара идентификаторов, состоящая из номера сети и номера узла. При этом IP-адрес идентифицирует не отдельный компьютер или маршрутизатор, а одно сетевое соединение в составе сети, в которую он входит; то есть конечный узел может входить в несколько IP-сетей.

DNS – система доменных имен

Несмотря на то, что аппаратное и программное обеспечение в рамках TCP/IP сетей для идентификации узлов использует IP-адреса, пользователи предпочитают *символьные имена* (*доменные имена*).

Первоначально в локальных сетях из небольшого числа компьютеров применялись плоские имена, состоящие из последовательности символов без разделения их на отдельные части, например *MYSOMP*. Для установления соответствия между символьными именами и числовыми адресами использовались широковещательные запросы. Однако для больших территориально распределенных сетей, работающих на основе протокола TCP/IP такой способ оказался неэффективным. Поэтому для установления соответствия между доменным именем и IP-адресом используется специальная система доменных имен (DNS, Domain Name System), которая основана на создаваемых администраторами сети таблиц соответствия.

В сетях TCP/IP используется доменная система имен, имеющая иерархическую (в виде дерева) структуру. Данная структура имен напоминает иерархию имен, используемую во многих файловых системах. Запись доменного имени начинается с самой младшей составляющей, затем после точки следует следующая по старшинству символьная часть имени и так далее. Последовательность заканчивается корневым именем, например: *company.yandex.ru*.

Построенная таким образом система имен позволяет разделять административную ответственность по поддержке уникальности имен в пределах своего уровня иерархии между различными людьми или организациями.

Совокупность имен, у которых несколько старших составных частей совпадают, образуют *домен* имен.

Корневой домен управляется центральными органами Интернета: *IANA* и *Internic*.

Домены верхнего уровня назначаются для каждой страны, а также для различных типов организаций. Имена этих доменов должны следовать международному стандарту *ISO 3166*. Для обозначения стран используются двухбуквенные аббревиатуры, например *ru* (Российская Федерация), *us* (США), *it* (Италия), *fr* (Франция).

Для различных типов организаций используются трехбуквенные аббревиатуры:

- *net* – сетевые организации;
- *org* – некоммерческие организации;
- *com* – коммерческие организации;
- *edu* – образовательные организации;
- *gov* – правительственные организации.

Администрирование каждого домена возлагается на отдельную организацию, которая делегирует администрирование поддоменов другим организациям.

Для получения доменного имени необходимо зарегистрироваться в соответствующей организации, которой организация *InterNIC* делегировала свои полномочия по распределению доменных имен.

С 1997 года компания «**АзияИнфо**» является официальным регистратором доменов в зоне KG. Данное право было делегировано компании Международной Корпорацией по Регистрации Имен и Номеров в Интернет (ICANN).

В TCP/IP сетях соответствие между доменными именами и IP-адресами может устанавливаться как локальными средствами, так и централизованными службами. Первоначально соответствие задавалось с помощью создаваемого вручную на хосте файла *hosts.txt*, состоящего из строк, содержащих пару вида

«доменное имя – IP-адрес». Однако с активным ростом Интернета такое решение оказалось немасштабируемым.

Альтернативное решение – централизованная служба DNS, использующая распределенную базу отображений «доменное имя – IP-адрес». Сервер домена хранит только имена, которые заканчиваются на следующем ниже по дереву уровне. Это позволяет распределять более равномерно нагрузку по разрешению имен между всеми DNS-серверами. Каждый DNS-сервер помимо таблицы отображения имен содержит ссылки на DNS-серверы своих поддоменов.

Существуют две схемы разрешения DNS-имен.

Нерекурсивная процедура:

1. DNS-клиент обращается к корневому DNS-серверу с указанием полного доменного имени;
2. DNS-сервер отвечает клиенту, указывая адрес следующего DNS-сервера, обслуживающего домен верхнего уровня, заданный в следующей старшей части имени;
3. DNS-клиент делает запрос следующего DNS-сервера, который отсылает его к DNS-

серверу нужного поддомена и т.д., пока не будет найден DNS-сервер, в котором хранится соответствие запрошенного имени IP-адресу. Сервер дает окончательный ответ клиенту.

Рекурсивная процедура:

1. DNS-клиент запрашивает локальный DNS-сервер, обслуживающий поддомен, которому принадлежит клиент;
2. Далее
3. Если локальный DNS-сервер знает ответ, он возвращает его клиенту
4. Если локальный сервер не знает ответ, то он выполняет итеративные запросы к корневому серверу. После получения ответа сервер передает его клиенту.

Таким образом, при рекурсивной процедуре клиент фактически перепоручает работу своему серверу. Для ускорения поиска IP-адресов DNS-серверы широко применяют кэширование (на время от часов до нескольких дней) проходящих через них ответов.

World Wide Web (WWW)

Сеть WWW образуют миллионы *web-серверов*, расположенных по всему миру. *Web-сервер* является программой, запускаемой на подключённом к сети компьютере и передающей данные по протоколу HTTP.

Для идентификации ресурсов (зачастую файлов или их частей) в WWW используются идентификаторы ресурсов *URI* (Uniform Resource Identifier). Для определения местонахождения ресурсов в этой сети используются локаторы ресурсов *URL* (Uniform Resource Locator). Такие URL-локаторы представляют собой комбинацию URI и системы DNS.

Доменное имя (или IP-адрес) входит в состав URL для обозначения компьютера (его сетевого интерфейса), на котором работает программа web-сервер.

На клиентском компьютере для просмотра информации, полученной от web-сервера, применяется специальная программа — *web-браузер*. Основная функция web-браузера - отображение гипертекстовых страниц (web-страниц). Для создания гипертекстовых страниц в WWW изначально использовался язык HTML. Множество web-страниц образуют *web-сайт*.

Понятие прокси-сервер

Прокси-сервер (proxy-server) — служба в компьютерных сетях, позволяющая клиентам выполнять косвенные запросы к другим сетевым службам.

Сначала клиент подключается к прокси-серверу и запрашивает какой-либо ресурс, расположенный на другом сервере. Затем прокси-сервер либо подключается к указанному серверу и получает ресурс у него, либо возвращает ресурс из собственного *кэша* (если имеется). В некоторых случаях запрос клиента или ответ сервера может быть изменен прокси-сервером в определенных целях. Также прокси-сервер позволяет защищать клиентский компьютер от некоторых сетевых атак.

Чаще всего прокси-серверы применяются для следующих целей:

- обеспечение доступа с компьютеров локальной сети в Интернет;
- кэширование данных: если часто происходят обращения к одним и тем же внешним ресурсам, то можно держать их копию на прокси-сервере и выдавать по запросу, снижая тем самым нагрузку на канал во внешнюю сеть и ускоряя получение клиентом запрошенной информации.
- сжатие данных: прокси-сервер загружает информацию из Интернета и передаёт информацию конечному пользователю в сжатом виде.
- защита локальной сети от внешнего доступа: например, можно настроить прокси-сервер так, что локальные компьютеры будут обращаться к внешним ресурсам только через него, а внешние компьютеры не смогут обращаться к локальным вообще (они «видят» только прокси-сервер).

— ограничение доступа из локальной сети к внешней: например, можно запретить доступ к определённым web-сайтам, ограничить использование интернета каким-то локальным пользователям, устанавливать квоты на трафик или полосу пропускания, фильтровать рекламу и вирусы.

— анонимизация доступа к различным ресурсам. Прокси-сервер может скрывать сведения об источнике запроса или пользователе. В таком случае целевой сервер видит лишь информацию о прокси-сервере, например, IP-адрес, но не имеет возможности определить истинный источник запроса. Существуют также искажающие прокси-серверы, которые передают целевому серверу ложную информацию об истинном пользователе.

Протоколы Интернет прикладного уровня

Самый верхний уровень в иерархии протоколов Интернет занимают следующие протоколы прикладного уровня:

- *DNS* - распределённая система доменных имён, которая по запросу, содержащему доменное имя хоста сообщает IP адрес;
- *HTTP* - протокол передачи гипертекста в Интернет;
HTTPS - расширение протокола HTTP, поддерживающее шифрование;
- *FTP* (File Transfer Protocol - RFC 959) - протокол, предназначенный для передачи файлов в компьютерных сетях;
- *Telnet* (TELEcommunication NETwork - RFC 854) - сетевой протокол для реализации текстового интерфейса по сети;
- *SSH* (Secure Shell - RFC 4251) - протокол прикладного, позволяющий производить удалённое управление операционной системой и передачу файлов. В отличие от Telnet шифрует весь трафик;
- *POP3* – протокол почтового клиента, который используется почтовым клиентом для получения сообщений электронной почты с сервера;
- *IMAP* - протокол доступа к электронной почте в Интернет;
- *SMTP* – протокол, который используется для отправки почты от пользователей к серверам и между серверами для дальнейшей пересылки к получателю;
- *LDAP* - протокол для доступа к службе каталогов X.500, является широко используемым стандартом доступа к службам каталогов;
- *XMPP* (Jabber) - основанный на XML расширяемый протокол для мгновенного обмена сообщениями в почти реальном времени;
- *SNMP* - базовый протокол управления сети Internet.

Рассмотрим более подробно некоторые из этих протоколов.

FTP

FTP позволяет подключаться к серверам FTP, просматривать содержимое каталогов и загружать файлы с сервера или на сервер; кроме того, возможен режим передачи файлов между серверами; FTP позволяет обмениваться файлами и выполнять операции над ними через TCP-сети. Данный протокол работает независимо от операционных систем. Исторически протокол FTP предложил открытую функциональность, обеспечивая прозрачный перенос файлов с одного компьютера на другой по сети. Это не так тривиально, как может показаться, так как у разнотипных компьютеров могут различаться размеры слов, биты в словах могут храниться в неодинаковом порядке или использоваться разные форматы слов.

Telnet

Название «telnet» имеют также некоторые утилиты, реализующие клиентскую часть

протокола. Протокол *telnet* работает в соответствии с принципами архитектуры «клиент-сервер» и обеспечивает эмуляцию алфавитно-цифрового терминала, ограничивая пользователя режимом командной строки. Приложение *telnet* предоставило язык для общения терминалов с удаленными компьютерами. Когда появилась сеть ARPANET, для каждой компьютерной системы требовались собственные терминалы. Приложение *telnet* стало общим знаменателем для терминалов. Достаточно было написать для каждого компьютера программное обеспечение, поддерживающее «терминал *telnet*», чтобы один терминал мог взаимодействовать с компьютерами всех типов.

SSH

Сходен по функциональности с протоколами *telnet* и *rlogin*, но, в отличие от них, шифрует весь трафик, включая и передаваемые пароли. SSH-клиенты и SSH-серверы имеются для большинства операционных систем.

Почтовые протоколы.

Хотя *telnet* и FTP были (и остаются) полезными, первым приложением, совершившим переворот в сознании пользователей компьютеров сети ARPANET, стала электронная почта. До сети ARPANET существовали системы электронной почты, но все они были однокомпьютерными системами. В 1972 г. Рэй Томлинсон (Ray Tomlinson) из компании BBN написал первый пакет, предоставляющий распределенные почтовые услуги в компьютерной сети из нескольких компьютеров. Уже к 1973 г. исследования управления ARPA показали, что три четверти всего трафика сети ARPANET составляла электронная почта. Польза электронной почты оказалась столь велика, что все больше пользователей стремилось подключиться к сети ARPANET, в результате чего возрастала потребность в добавлении новых узлов и использовании высокоскоростных линий. Таким образом, появилась тенденция, сохраняющаяся и по сей день.

- *POP3* (Post Office Protocol Version 3 - RFC 1939) — протокол, который используется почтовым клиентом для получения сообщений электронной почты с почтового сервера;

- *IMAP* (Internet Message Access Protocol - RFC 3501) — протокол доступа к электронной почте. Аналогичен POP3, однако предоставляет пользователю богатые возможности для работы с почтовыми ящиками, находящимися на центральном сервере. Электронными письмами можно манипулировать с компьютера пользователя (клиента) без необходимости постоянной пересылки с сервера и обратно файлов с полным содержанием писем.

- *SMTP* (Simple Mail Transfer Protocol — RFC 2821) — протокол, предназначенный для передачи электронной почты. Используется для отправки почты от пользователей к серверам и между серверами для дальнейшей пересылки к получателю. Для приёма почты почтовый клиент должен использовать протоколы POP3 или IMAP.

Контрольные вопросы:

Общее устройство сети интернет.

Понятие домена и управление доменами.

Протоколы интернет.

План СРМ:

1. Понятие прокси-сервер

2. Протоколы Интернет прикладного уровня

VIII. ЛАБОРАТОРНЫЕ МАТЕРИАЛЫ

Лабораторная работа №1: Служебные утилиты для работы в Интернет. Изучение протокола HTTP

Аннотация: Цель работы: изучение структуры IP-адреса; ознакомление с наиболее популярными утилитами для диагностики сетевой конфигурации и сетевых соединений; ознакомление с основами протокола HTTP. (ОК-2, ПК-13)

Структура IP адреса

IP-адрес состоит из двух частей: номера сети и номера узла в сети.

Самой распространенной является *запись IP-адреса* в виде четырех чисел, разделенных точками, каждое из которых представляет *значение* байта в десятичной форме, например: **213.180.204.11**. *Запись* адреса не предусматривает специального разграничительного знака между номером сети и номером узла.

Для разделения этих частей обычно используется 2 подхода:

- С помощью маски (RFC 950, RFC 1518), представляющей собой число в паре с IP-адресом. С помощью операции "логическое И" над этими двумя числами выделяется номер сети.
- С помощью классов адресов (RFC 791).

Вводится пять классов адресов: A, B, C, D, E.

A, B, C – используются для адресации сетей, D и E – имеют специальное назначение. Признаком, на основании которого *IP-адрес* относят к тому или иному классу, являются значения нескольких первых битов адреса.

Таблица 3.1. Распределение адресов в IP сетях.

| Класс | Первые биты | Наименьший номер сети | Наибольший номер сети | Максимальное число узлов в сети |
|-------|-------------|-------------------------------|--------------------------------|---------------------------------|
| A | 0 | 1.0.0.0 (0 - не используется) | 126.0.0.0 (127-зарезервирован) | 2^{24} (3 байта) |
| B | 10 | 128.0.0.0 | 191.255.0.0 | 2^{16} (2 байта) |
| C | 110 | 192.0.0.0 | 223.255.255.0 | 2^8 (1 байт) |
| D | 1110 | 224.0.0.0 | 239.255.255.255 | групповые адреса |
| E | 11110 | 240.0.0.0 | 247.255.255.255 | зарезервировано |

В рамках *IP* протокола существуют ограничения при назначении *IP-адресов*, а именно

- номера сетей и номера узлов не могут состоять из двоичных нулей или единиц;
- если IP-адрес состоит только из двоичных нулей, то он называется неопределенным адресом и обозначает адрес того узла, который сгенерировал этот пакет;
- если в поле номера сети стоят только нули, то по умолчанию считается, что узел назначения принадлежит той же самой сети, что и узел, который отправил пакет; такой адрес может быть использован только в качестве адреса отправителя;
- если все двоичные разряды IP-адреса равны 1, то пакет с таким адресом назначения должен рассылаться всем узлам, находящимся в той же сети, что и источник этого пакета; такой адрес называется *ограниченным широковещательным*, поскольку пакет не сможет выйти за границы сети;
- если в поле адреса назначения в разрядах, соответствующих номеру узла, стоят только единицы, то пакет рассылается всем узлам сети, номер которой указан в адресе назначения; такой тип адреса называется *широковещательным*;
- если первый октет адреса равен 127, то такой адрес называется внутренним адресом стека протоколов; он используется для тестирования программ, организации клиентской и серверной частей приложений, установленных на одном компьютере;
- групповые адреса, относящиеся к классу D, предназначены для экономичного распространения в Интернете, большой корпоративной сети аудио- или видеопрограмм.

Стандартным *классам сетей* можно поставить в соответствие следующие значения маски:

- класс A – 255.0.0.0 ;
- класс B – 255.255.0.0 ;
- класс C – 255.255.255.0.

Рассмотрим следующий пример:

| | | |
|---------------------|-------------------------|---------------|
| Исходные данные | <i>IP адрес</i> | 62.76.167.21 |
| | <i>Маска сети</i> | 255.255.255.0 |
| Логическая операция | И | |
| Результат | <i>Адрес сети</i> | 62.76.167.0 |
| | <i>Номер компьютера</i> | 21 |

Для определения сетевых настроек компьютера и сетевого оборудования, диагностики и получения другой информации, относящейся к интернет-протоколам, широко используются специальные утилиты.

Утилита `ipconfig`

`Ipconfig` - это утилита командной строки для вывода деталей текущего соединения компьютера с сетью и контроля над клиентским сервисом *DHCP*. *DHCP* (*Dynamic Host Configuration Protocol*) - это сетевой протокол, позволяющий компьютерам автоматически получать *IP-адрес* и другие параметры, необходимые для работы в сети *TCP/IP*.

Синтаксис команды:

`ipconfig/ключи`

Команда `ipconfig/all` - отображает полную информацию по всем сетевым адаптерам.

Пример вывода для *Windows*:

```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Версия 5.1.2600]
(C) Корпорация Майкрософт, 1985-2001.

Z:\>ipconfig/all

Настройка протокола IP для Windows

Имя компьютера . . . . . : C1R385N16
Основной DNS-суффикс . . . . . : cs.vsu.ru
Тип узла . . . . . : неизвестный
IP-маршрутизация включена . . . . . : нет
WINS-прокси включен . . . . . : нет
Порядок просмотра суффиксов DNS . . . . . : cs.vsu.ru
                                                    cs.vsu.ru
                                                    vsu.ru

Подключение по локальной сети 3 - Ethernet адаптер:
UMnet8
DNS-суффикс этого подключения . . . . . :
Описание . . . . . : VMware Virtual Ethernet Adapter for
Физический адрес . . . . . : 00-50-56-C0-00-08
Dhcp включен . . . . . : нет
IP-адрес . . . . . : 192.168.111.1
Маска подсети . . . . . : 255.255.255.0
Основной шлюз . . . . . :

Подключение по локальной сети 2 - Ethernet адаптер:
UMnet1
DNS-суффикс этого подключения . . . . . :
Описание . . . . . : VMware Virtual Ethernet Adapter for
Физический адрес . . . . . : 00-50-56-C0-00-01
Dhcp включен . . . . . : нет
IP-адрес . . . . . : 192.168.61.1
Маска подсети . . . . . : 255.255.255.0
Основной шлюз . . . . . :

Подключение по локальной сети - Ethernet адаптер:
on #2
DNS-суффикс этого подключения . . . . . : cs.vsu.ru
Описание . . . . . : Intel(R) PRO/100 VE Network Connecti
Физический адрес . . . . . : 00-14-85-17-CD-6C
Dhcp включен . . . . . : да
Автонастройка включена . . . . . : да
IP-адрес . . . . . : 10.16.1.146
Маска подсети . . . . . : 255.255.255.0
Основной шлюз . . . . . : 10.16.1.2
DHCP-сервер . . . . . : 62.76.220.205
DNS-серверы . . . . . : 62.76.220.205
                        62.76.220.204
Аренда получена . . . . . : 12 ноября 2008 г. 8:08:31
Аренда истекает . . . . . : 12 ноября 2008 г. 23:08:31

Z:\>_

```

Утилита ping

Ping (*Packet InterNet Grouper*) - это системная программа, предназначенная для проверки соединений в сетях на основе *TCP/IP*. Она отправляет *Echo-Request* запросы протокола *ICMP* указанному узлу сети и фиксирует поступающие ответы (*ICMP Echo-Reply*). Время между отправкой запроса и получением ответа (*RTT, Round Trip Time*) позволяет определять двусторонние задержки (*RTT*) по маршруту и частоту потери пакетов. Что позволяет косвенно определять загруженность каналов передачи данных и промежуточных устройств. Полное отсутствие *ICMP*-ответов может также означать, что удалённый узел (или какой-либо из промежуточных маршрутизаторов) блокирует *ICMP Echo-Reply* или игнорирует *ICMP Echo-Request*.

Синтаксис:

ping -параметры конечное_имя

Конечное имя – это доменное имя или *IP-адрес* хоста

Пример:

```
C:\>ping www.mail.ru

Обмен пакетами с www.mail.ru [194.67.57.26] по 32 байт:
Ответ от 194.67.57.26: число байт=32 время=11мс TTL=118
Ответ от 194.67.57.26: число байт=32 время=11мс TTL=118
Ответ от 194.67.57.26: число байт=32 время=11мс TTL=118
Ответ от 194.67.57.26: число байт=32 время=11мс TTL=118

Статистика Ping для 194.67.57.26:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0 (0% потерь),
Приблизительное время приема-передачи в мс:
    Минимальное = 11мсек, Максимальное = 11 мсек, Среднее = 11 мсек
```