

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ

Проректор по учебной работе
и образовательным инновациям

О.Г. Прохоренко

30 июня 2023 г.

Регистрационный № УД-12552/уч.

Web-программирование

**Учебная программа учреждения высшего образования
по учебной дисциплине для специальности:**

1-31 03 07 Прикладная информатика (по направлениям)

Направления специальности:

1-31 03 07-01 прикладная информатика

(программное обеспечение компьютерных систем)

2023 г.

Учебная программа составлена на основе ОСВО 1-31 03 07-2021, типового учебного плана № G 31-1-030/пр-тип от 01.07.2021, учебных планов №G 31-1-034/уч. от 23.07.2021, №G 31-1-023/уч.ин. от 09.08.2021, № G31-1-216/уч. от 22.03.2022

СОСТАВИТЕЛЬ:

Е.Д. Рафеенко, доцент кафедры многопроцессорных систем и сетей Белорусского государственного университета, кандидат физико-математических наук, доцент

РЕЦЕНЗЕНТ:

Н.В. Лапицкая – заведующая кафедрой программного обеспечения информационных систем Учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», кандидат технических наук, доцент.

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой многопроцессорных систем и сетей БГУ
(протокол № 13 от 06.04.2023).

Научно-методическим советом Белорусского государственного университета
(протокол № 9 от 29.06.2023 г.).

Заведующий кафедрой
многопроцессорных систем и сетей



С.В.Марков

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Цели и задачи учебной дисциплины

Цель учебной дисциплины «Web-программирование»: создание представления о современных технологиях проектирования web приложений. Учебная дисциплина «Web-программирование» предполагает изучение языков и технологий программирования клиентских и серверных приложений.

Учебная дисциплина предусматривает изучение языков программирования JavaScript, TypeScript, фреймворков React, Angular для разработки приложений на клиентской стороне и Java Enterprise платформы, фреймворка Spring для создания приложений на серверной стороне. Также в рамках дисциплины рассматриваются вопросы организации защиты web-приложений. Содержание учебного материала ориентировано на подготовку студентов к практическому использованию полученных знаний, формирование у них широкого кругозора в области информационных технологий.

Задачи учебной дисциплины:

- формирование представления об архитектуре web-приложений;
- ознакомление с методами проектирования web-приложений;
- подготовка к самостоятельной разработке web-приложений с использованием современных технологий.

Место учебной дисциплины

В системе подготовки специалиста с высшим образованием для специальности 1-31 03 07 «Прикладная информатика» учебная дисциплина относится к модулю «Прикладное программирование» компонента учреждения высшего образования.

Связи с другими учебными дисциплинами: базовыми для изучения дисциплины «Web-программирование» являются учебные дисциплины модуля «Программирование» государственного компонента.

Требования к компетенциям

Освоение учебной дисциплины «Web-программирование» должно обеспечить формирование следующих **специальных компетенций**:

СК-5. Использовать программные средства и технологии для создания прикладного программного обеспечения.

В результате изучения учебной дисциплины студент должен:

знать:

- концепции, положенные в основу языков для разработки клиентских и серверных приложений;
- методы обеспечения безопасности информационных систем, построенных на основе web-технологий.

уметь:

- анализировать и разрабатывать проекты корпоративных информационных систем (сайтов);

- создавать динамическое содержание web-страниц;
- организовывать доступ к базам данных в web-приложениях;
- обеспечивать безопасность пользователей и защиту информации;

владеть:

- навыками разработки web-систем;
- навыками использования программных комплексов для проектирования, создания и управления web-приложений.

Структура учебной дисциплины

Дисциплина изучается в 4, 5 семестрах. Всего на изучение учебной дисциплины «Web-программирование» отведено:

- для очной формы получения высшего образования– 216 часов, в том числе 132 аудиторных часа, из них: лекции – 66 часов, лабораторные занятия – 58 часов, управляемая самостоятельная работа – 8 часов.

Трудоемкость учебной дисциплины составляет 6 зачетных единиц.

Форма текущей аттестации – зачет в 4 семестре, экзамен в 5 семестре.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Раздел 1. Введение в web разработку

Основы World Wide Web. Общие принципы построения web приложений. Этапы создания web-проекта.

Раздел 2. Язык HTML, каскадные таблицы стилей CSS

Язык HTML - цвет, шрифт, границы, отступы, размеры. Обтекание. Вёрстка макетов. Селекторы, иерархические селекторы. Каскадирование. Медиазапросы. Адаптивная вёрстка.

Раздел 3. Язык JavaScript

Язык JavaScript: типы данных, операторы, функции и объекты. Массивы и хэши. Встроенные классы и объекты. Замыкания. Принципы ООП в JavaScript, прототипирование. Работа с DOM. Работа с окном браузера и с документом. События, всплытие событий.

Раздел 4. Front end разработка

Тема 4.1. JavaScript object notation (JSON). Хранение данных на клиенте

Встроенные методы для работы с JSON. JSON и клонирование. Куки (Cookie). Локальное хранилище данных localStorage.

Тема 4.2. Одностраничные приложения (Single page applications). MVC

Особенности одностраничных приложений, сравнение с традиционными web-приложениями. Паттерны Model View Controller (MVC), Model View Presenter (MVP), Model View ViewModel (MVVM).

Технология AJAX. Асинхронные AJAX-запросы.

Тема 4.3 Модульность. Сборка JavaScript приложений

Модульная организация кода в JavaScript приложении. Инструмент Webpack, встроенный webpack-dev-server. Фреймворк Express. Базовая настройка маршрутизации в приложении.

Тема 4.4 Реактивное программирование. Библиотека RxJS

Потоки событий (Stream). Наблюдатели, наблюдаемые, подписки. Протоколы push и pull. Операторы создания, конвейеризации, трансформации, объединения потоков событий.

Тема 4.5 Язык TypeScript

Основные возможности языка, сравнение с JavaScript. Типы, классы, стрелочные функции. Generic – типы. Модули.

Тема 4.6 Фреймворк React

Технология JavaScript XML (JSX). Настраиваемые компоненты. Жизненный цикл и состояние компонента. Обработка событий, формы.

Тема 4.7 Фреймворк Angular

Модули, компоненты, директивы. Привязка данных. Маршрутизация и навигация (Routing). Получение данных с сервера. Http-клиент.

Раздел 5. Получение данных с сервера

Тема 5.1. Организация работы с базами данных - технология Java DataBase Connectivity (JDBC API)

Соединение с базой данных. Подготовленные запросы и хранимые процедуры. Пул соединений. Выделение ресурсов соединениям. Data Access Object.

Тема 5.2. Доступ к базам данных с использованием ORM фреймворков

Технология Java Persistence API. Hibernate. Сущности, отношения между сущностями. Язык запросов JPQL. CriteriaAPI, MetamodelAPI.

Раздел 6. Серверные web –приложения

Тема 6.1. Протокол HTTP

Схема HTTP-сеанса. Состав HTTP-запроса. Обеспечение безопасности передачи данных HTTP.

Тема 6.2. Web-сервер и серверные технологии

Взаимодействие между браузером и web-сервером. Сценарные языки: классификация по быстродействию. Компонентные технологии разработки web-приложений.

Тема 6.3. Web- компоненты платформы Java Enterprise (Jakarta EE)

Сервлеты (servlets), контекст сервлета, жизненный цикл. Сессии, обработчики web событий, фильтры.

Тема 6.4. Технология шаблонов Thymeleaf

Thymeleaf API, сравнение с другими механизмами создания шаблонов. Контекст Thymeleaf приложения. Выражения в Thymeleaf.

Тема 6.5. Фреймворк Spring

Внедрение зависимостей. Bean-компоненты, области видимости. Spring Boot. Работа с данными в Spring. Spring MVC, контроллеры RESTful. WebSockets в Spring.

Раздел 7. Защита web приложений

Тема 7.1. Авторизация и аутентификация

Типы аутентификации. Использование SSL и сертификата клиента.

Тема 7.2. Spring Security

Служба хранения сведений о пользователях. Провайдеры аутентификации. Защита web - запросов.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Очная форма получения высшего образования с применением дистанционных образовательных технологий (ДОТ)

Номер раздела, темы,	Название раздела, темы	Количество аудиторных часов					Количество часов УСР	Формы контроля знаний
		лекции	практические занятия	семинарские занятия	лабораторные занятия	Иное		
1	2	3	4	5	6	7	8	9
1	Введение в web разработку	2						Устный опрос
2	Язык HTML, каскадные таблицы стилей CSS	4			4			Контрольная работа №1. Устная защита лабораторных работ
3	Язык JavaScript	8			10			Контрольная работа №2 Устная защита лабораторных работ
4	Front end разработка	24			20		4	Коллоквиум
4.1	JavaScript object notation (JSON). Хранение данных на клиенте	2			2			Устная защита лабораторных работ
4.3	Одностраничные приложения (Single page applications). MVC	2			2			Устная защита лабораторных работ

4.3	Модульность. Сборка приложений	JavaScript	2			2			Устная защита лабораторных работ
4.4	Реактивное программирование. Библиотека RxJS		4			4			Контрольная работа №3. Устная защита лабораторных работ
4.5	Язык TypeScript		2					2	Собеседование по заданиям УСР
4.6.	Фреймворк React		6			4		2	Контрольная работа №4. Устная защита лабораторных работ
4.7	Фреймворк Angular		6			6			Контрольная работа №5. Устная защита лабораторных работ
5	Получение данных с сервера		8			8			
5.1	Организация работы с базами данных - технология Java DataBase Connectivity (JDBC API)		4			4			Контрольная работа №6. Устная защита лабораторных работ
5.2	Доступ к базам данных с использованием ORM фреймворков		4			4			Устная защита лабораторных работ
6	Серверные web –приложения		16			12		4	Коллоквиум

6.1	Протокол HTTP	1					Устный опрос по теме 6.1
6.2	Web-сервер и серверные технологии	1					Устный опрос по теме 6.2
6.3	Web-компоненты платформы Java Enterprise (Jakarta EE)	4			4	2	Контрольная работа №7 Устная защита лабораторных работ
6.4	Технология шаблонов Thymeleaf	4			4		Устная защита лабораторных работ
6.5	Фреймворк Spring	6			4	2	Устная защита лабораторных работ. Собеседование по заданиям УСП
7	Защита web приложений	4			4		
7.1	Авторизация и аутентификация	2					Собеседование
7.2	Spring Security	2			4		Контрольная работа №8 Устная защита лабораторных работ

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Перечень основной литературы

1. Робсон, Э. Изучаем HTML, XHTML и CSS / Элизабет Робсон, Эрик Фримен ; [пер. с англ. В. Черник]. - 2-е изд. - Москва ; Санкт-Петербург ; Минск : Питер, 2022. - 720 с. - URL: <https://ibooks.ru/bookshelf/377028/reading>.
2. Симпсон, К. Область видимости и замыкания / Кайл Симпсон ; [пер. с англ. Е. Матвеев]. - 2-е междунар. изд. - Санкт-Петербург ; Москва ; Минск : Питер, 2022. - 240 с. - URL: <https://ibooks.ru/reading.php?short=1&productid=379921>
3. Фримен, А. Angular для профессионалов = Pro Angular / Адам Фримен ; [пер. с англ. Е. Матвеева]. - Санкт-Петербург [и др.] : Питер, 2018. - 800 с. - URL: <https://ibooks.ru/reading.php?short=1&productid=356434>.
4. Бэнкс, А. React. Современные шаблоны для разработки приложений / Алекс Бэнкс, Ева Порселло ; [пер. с англ. С. Черников]. - 2-е изд. - Санкт-Петербург ; Москва ; Минск : Питер, 2022. - 317 с. - URL: <https://ibooks.ru/bookshelf/379917/reading>.
5. Дронов, В. А. React 17. Разработка веб-приложений на JavaScript / Владимир Дронов. - Санкт-Петербург : БХВ-Петербург, 2022. - 384 с. - URL: <https://ibooks.ru/bookshelf/385763/reading>.
6. Хеклер, М. Spring Boot по-быстрому. Создаем облачные приложения на Java и Kotlin / М. Хеклер ; [пер. с англ. И. Пальти]. - Санкт-Петербург ; Москва ; Минск : Питер, 2022. - 351 с. - URL: <https://ibooks.ru/reading.php?short=1&productid=385989>.

Перечень дополнительной литературы

1. Черный, Б. Профессиональный TypeScript. Разработка масштабируемых JavaScript -приложений – Санкт-Петербург [и др.] : Питер, 2021. – 351 с.
2. Современный учебник JavaScript [Электронный ресурс]. – Режим доступа : <https://learn.javascript.ru/>. Дата доступа: 21.01.2023
3. TypeScript [Электронный ресурс]. – Режим доступа : <https://www.typescriptlang.org/>. Дата доступа: 20.01.2023
4. Фреймворк Angular [Электронный ресурс]. – Режим доступа : <https://angular.io/>. Дата доступа: 24.02.2023
5. Фреймворк React [Электронный ресурс]. – Режим доступа : <https://react.dev/>. Дата доступа: 06.02.2023
6. Фреймворк Spring [Электронный ресурс]. – Режим доступа : <https://spring.io/>. Дата доступа: 15.02.2023
7. Thymeleaf [Электронный ресурс]. – Режим доступа : <https://www.thymeleaf.org/>. Дата доступа: 20.02.2023
8. Блинов, И. Н. Java from EPAM : учеб.-метод. пособие / И. Н. Блинов, В. С. Романчик. – Минск: Четыре четверти, 2021. — 560 с

Электронные ресурсы

1. Образовательный портал БГУ [Электронный ресурс]. – Режим доступа: <https://edufpmi.bsu.by/course/view.php?id=9>. – Дата доступа: 06.02.2023.
2. Образовательный портал БГУ [Электронный ресурс]. – Режим доступа: <https://edufpmi.bsu.by/course/view.php?id=351>. – Дата доступа: 06.02.2023.

Перечень рекомендуемых средств диагностики и методика формирования итоговой оценки

Объектом диагностики компетенций студентов являются знания, умения, полученные ими в результате изучения учебной дисциплины. Выявление учебных достижений студентов осуществляется с помощью мероприятий текущего контроля и промежуточной аттестации.

Для диагностики компетенций в рамках учебной дисциплины рекомендуется использовать следующие формы:

1. Устная форма: опросы, устная защита лабораторных работ.

2. Письменная форма: контрольные работы для оценивания на основе модульно-рейтинговой системы.

В качестве рекомендуемых технических средств диагностики используется обучение, организованное на платформе Moodle (<https://edufpmi.bsu.by>).

Используется рейтинговая оценка знаний студента, дающая возможность проследить и оценить динамику процесса достижения целей обучения. Рейтинговая оценка предусматривает использование весовых коэффициентов для текущего контроля знаний и текущей аттестации студентов по дисциплине.

Примерные весовые коэффициенты, определяющие вклад текущего контроля знаний и текущей аттестации в рейтинговую оценку:

- защита лабораторных работ – 40 %;

- контрольные работы – 40 %;

- опросы – 10 %.

- собеседования по заданиям УСР – 10%

Рейтинговая оценка по дисциплине рассчитывается на основе оценки текущей успеваемости и экзаменационной оценки с учетом их весовых коэффициентов. Вес оценка по текущей успеваемости составляет 40 %, экзаменационная оценка – 60 %.

Формой промежуточной аттестации по дисциплине «Web программирование» учебным планом предусмотрены зачет, экзамен.

Примерная тематика лабораторных занятий

1. Разработка html-страниц, таблицы, стили
2. Табличная и блочная верстка
3. Введение в JavaScript. Функции, хэши.
4. Введение в JavaScript. Функции, хэши – ООП подход
5. JavaScript. Работа с DOM.
6. JavaScript. Обработка событий.
7. JavaScript. Хранение данных на клиенте. MVC. Модульность
8. Создание приложения с использованием RxJS.
9. Создание приложения с использованием RxJS. Обработка событий.
10. Настройка Express .

11. Разработка React компонентов.
12. Создание архитектуры Angular приложения.
13. Создание Angular приложения. Обработка данных.
14. Создание Http-клиента в Angular приложении.
15. Создание Angular приложения – навигация.
16. Разработка форм в Angular приложении.
17. Развертывание Angular приложения на Firebase.
18. Создание приложения, имеющего доступ к базе данных, использование API JDBC. Connection Wrapper.
19. Создание приложения, имеющего доступ к базе данных, использование API JDBC. Шаблон DAO.
20. Организация пула соединений.
21. Создание приложения, имеющего доступ к базе данных, с использованием технологии Java Persistence API.
22. Создание приложения, имеющего доступ к базе данных, с использованием технологии Java Persistence API. Criteria API, Metamodel API.
23. Программирование сервлетов. Использование Cookies.
24. Проектирование приложения с использованием Spring Boot, Spring MVC.
25. Проектирование приложения с использованием Spring Boot, Spring Data.
26. Проектирование приложения с использованием Spring Boot, Spring Security.
27. Проектирование пользовательского интерфейса
28. Проектирование RESTful web-сервиса.
29. Проектирование клиента RESTful web-сервиса.
30. Проектирование клиента RESTful web-сервиса. Взаимодействие с Angular приложением

Примерный перечень заданий для управляемой самостоятельной работы студентов

1. Язык TypeScript: Разработка React и TypeScript компонентов. Форма контроля – собеседование.
2. React - добавление типа состояния. Форма контроля – собеседование.
3. Организация поддержки сессии. Форма контроля – собеседование.
4. Проектирование Spring репозитория. Форма контроля – собеседование.

Описание инновационных подходов и методов к преподаванию учебной дисциплины

При организации образовательного процесса используются следующие методы:

- метод учебной дискуссии, который предполагает участие студентов в целенаправленном обмене мнениями, идеями для предъявления и/или согласования существующих позиций по определенной проблеме. Использование метода обеспечивает появление нового уровня понимания изучаемой темы, применение

знаний (теорий, концепций) при решении проблем, определение способов их решения.

- метод группового обучения, который представляет собой форму организации учебно-познавательной деятельности обучающихся, предполагающую функционирование разных типов малых групп, работающих как над общими, так и специфическими учебными заданиями.

В качестве технических средств для организации работы в рамках учебной дисциплины рекомендуется использовать Образовательный портал БГУ (<https://edufpmi.bsu.by>) – инструмент с эффективной функциональностью контроля, тренинга и самостоятельной работы.

Используются методы и приемы развития критического мышления, которые представляют собой систему, формирующую навыки работы с информацией; понимания информации как отправного, а не конечного пункта критического мышления.

При организации образовательного процесса используется практико-ориентированный подход, который предполагает освоение содержания дисциплины через выполнение практических заданий, имеющих элементы учебно-исследовательской деятельности.

Методические рекомендации по организации самостоятельной работы обучающихся

Для организации самостоятельной работы студентов по учебной дисциплине следует использовать современные информационные ресурсы: разместить на образовательном портале комплекс учебных и учебно-методических материалов (учебно-программные материалы, учебное издание для теоретического изучения дисциплины, методические указания к лабораторным занятиям, материалы текущего контроля и текущей аттестации, позволяющие определить соответствие учебной деятельности обучающихся требованиям образовательного стандарта высшего образования и учебно-программной документации, в т.ч. вопросы для подготовки к зачету, задания, тесты, вопросы для самоконтроля, тематика рефератов и др., список рекомендуемой литературы, информационных ресурсов и др.).

Темы реферативных работ

1. Языки разработки сценариев.
2. Компонентные технологии разработки web-приложений.

Рекомендуемая тематика контрольных работ и коллоквиума:

Контрольная работа № 1

Язык HTML, каскадные таблицы стилей CSS. Язык JavaScript – встроенный объект window.

Контрольная работа № 2

Язык JavaScript – работа с тегами на web странице

Контрольная работа № 3
Реактивное программирование с использованием библиотеки RxJS

Контрольная работа № 4
React компоненты с состоянием.

Коллоквиум

Раздел 4 - Front end разработка.

Контрольная работа № 5
Фреймворк Angular – маршрутизация в приложении.

Контрольная работа № 6
Взаимодействие с базами данных - JDBC API.

Контрольная работа № 7
Сервлеты и фильтры в web- приложении.

Контрольная работа № 8
Программная аутентификация.

Коллоквиум

Раздел 6 - Серверные web –приложения.

Примерный вариант контрольной работы № 1

Создать страницу Task.html с кнопкой, по которой вызывается JS скрипт, позволяющий выполнить программный переход на другой URL.

Примерный вариант контрольной работы № 2

Создать страницу Task.html с формой, в которой должны быть: текстовое поле, переключатель и кнопка Submit. Написать JS скрипт, который валидирует форму (правила валидации – непустое поле ввода и отмеченный переключатель).

Примерный вариант контрольной работы № 3

Создать страницу Task.html с формой, в которой должны быть: текстовое поле и кнопка Submit. Щелчок по кнопке запускает поток, в котором счетчик увеличивается на единицу каждую секунду. Каждый новый щелчок останавливает предыдущий поток и запускает новый, счетчик обнуляется.

Примерный вариант контрольной работы № 4

Создать React компонент, который будет отображать названия всех областей Беларуси, содержащих заданную подстроку.

Примерный вариант контрольной работы № 5

Создать Angular модуль с названием TransportCompany, в этом модуле разместить компоненты TransportCenter и TransportList. При открытии проекта отображается компонент TransportCenter, для которого дочерним подгружается компонент TransportList.

Примерный вариант контрольной работы № 6

Создать БД с одной таблицей, в которой хранится информация о товаре (название, стоимость, наличие). Приложение получает из БД список товаров, которых нет в наличии.

Примерный вариант контрольной работы № 7

Разработать web –приложение, состоящее из сервлета, двух thymeleaf страниц и dao-класса UserDao. Первая страница содержит форму с полем ввода возраста и кнопкой, по которой вызывается сервлет, перенаправляющий на вторую страницу.

Примерный вариант контрольной работы № 8

Разработать web –приложение, состоящее из сервлета, двух thymeleaf страниц и dao-класса UserDao. Первая страница содержит форму с полем ввода имени и пароля и кнопкой, по которой происходит аутентификация пользователя .

Примерный перечень вопросов к зачету

1. Общие принципы построения web приложений. Этапы создания web-проекта.
2. Клиентские сценарии и приложения
3. Программы, выполняющиеся на клиент-машине.
4. Язык JavaScript.
5. Замыкания.
6. Принципы ООП в JavaScript, прототипирование.
7. Работа с DOM.
8. Работа с окном браузера и с документом.
9. События, всплытие событий.
10. Особенности одностраничных приложений,
11. Локальное хранилище данных localStorage.
12. Модульная организация кода в JavaScript приложении. Инструмент Webpack,
13. Реактивное программирование. Библиотека RxJS
14. Язык TypeScript
15. Фреймворк React

Примерный перечень вопросов к экзамену

1. Архитектура Angular приложения.
2. Связывание данных.
3. Работа с формами.
4. Навигация в Angular приложении.
5. Язык TypeScript. Observables в Angular приложении.
6. Java DataBase Connectivity (JDBC API).
7. Соединение с базой данных.
8. Подготовленные запросы и хранимые процедуры.
9. Шаблон Data Access Object.
10. Пул соединений. Выделение ресурсов соединениям.
11. Транзакции.
12. Технология Java Persistence API.

13. Описание сущностей и отношений между ними.
14. Преобразование наследования.
15. Управление сущностями.
16. Java Persistence Query Language.
17. Использование Criteria API и Metamodel API для создания запросов.
18. Протокол HTTP. Схема HTTP-сеанса. Состав HTTP-запроса, ответа.
19. Web- компоненты Сервлеты.
20. Динамические страницы Thymeleaf.
21. Фильтры.
22. Сессии, куки.
23. Spring Framework
24. Внедрение зависимостей. Dispatcher servlet.
25. Репозитории.
26. REST - контроллер.
27. RESTful service клиент.
28. Безопасность в Spring.

ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ

Название учебной дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола)

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ

на ____ / ____ учебный год

№№ ПП	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры
многопроцессорных систем и сетей (протокол № ____ от _____ 200_ г.)

Заведующий кафедрой

УТВЕРЖДАЮ

Декан факультета
