Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Песцов Дмитрий Николаевич
Должность: Проректор ТГАСУ по учебной работе
Дата подписания: 16.11.2026 15:52:44
Уникальный программный ключ:
377c65ba4b077ae829074a6e0578e9ca83c05cf0

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ АРХИТЕКТУРНО-СТРОИТЕЛЬНЫЙ УНИВЕРСИТЕТ»

Институт непрерывного образования

СОГЛАСОВАНО
Директор ИНО

_____ Н.Р.Шадейко
(подпись)

"____"_____20___ г.

УТВЕРЖДАЮ
Проректор по учебной работе

_____ Д.Н. Песцов
(подпись)

"____"_____20___ г.

ПРОГРАММА МОДУЛЯ

Основы web-технологий
(наименование дисциплины)

ДОПОЛНИТЕЛЬНАЯ ПРОФЕССИОНАЛЬНАЯ ПРОГРАММА

«Web-программирование»
(наименование программы)

Томск 2025

Составитель _____*профессор кафедры ПМ, Г.А. Онопенко*_____
(Должность, Ф.И.О., подпись)

Рецензент _____*профессор кафедры ПМ, Б.М. Шумилов*_____
(Должность, Ф.И.О., подпись)

Введена в действие с «___» _____ 20__ г.

СОДЕРЖАНИЕ

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ 4
1.1 Учебный план 4
1.2 Планируемые результаты обучения 4
2. СОДЕРЖАНИЕ ПРОГРАММЫ 5
2.1 Учебно-тематический план модуля 5
3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ 7
3.1 Материально-технические условия реализации программы 7
3.2 Учебно-методическое и информационное обеспечение программы 7
3.2.1 Литература 7
3.2.2 Методические разработки 8
3.3 Процедура зачета 9
3.4 Кадровое обеспечение 9
ПРИЛОЖЕНИЕ 1 10
ПРИЛОЖЕНИЕ 2 12

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ

1.1 Учебный план

Наименование дисциплины	Общая трудоем-кость, час.	Всего, ауд. час.	Аудиторные занятия, час.			СРС, час.	Текущий контроль*, (шт.)			Промежуточная аттестация	
			лекции	лабораторные работы	прак. занятия, семинар		РК, РГР	КР	КП	Зачет	Экзамен
Модуль 1. Основы web-технологий											
Основы цифровой культуры. Информационные технологии и информационная безопасность	2	1	1			1					
Структура и принципы WWW. Web-редакторы	12	7	3		4	5					
Язык гипертекстовой разметки HTML	28	20	6		14	8					
Каскадные таблицы стилей CSS	44	28	8		20	16					
Итого	**86**	**56**	**18**		**38**	**30**				**Зачет**	

1.2 Планируемые результаты обучения

Слушатель программы переподготовки будет обладать следующими профессиональными компетенциями:

Матрица компетенций дисциплины

Профессиональные компетенции	ID-28	ID-35
	Применяет языки программирования для решения профессиональных задач	Разрабатывает различные веб-архитектуры

Слушатель программы переподготовки будет способен осуществлять следующие трудовые действия:

- Выявление первоначальных требований заказчика к информационным ресурсам (ИР) (ID-28);
- Определение возможности достижения соответствия ИР первоначальным требованиям заказчика (ID-28);
- Проектирование структуры разделов ИР (ID-28);
- Разработка интерфейса пользователя для ИР с использованием стандартов в области web-разработки (ID-35).

Слушатель программы переподготовки будет обладать следующими необходимыми умениями:

- умеет применять выбранные языки web-программирования и технологии проектирования сайтов (ID-28, У1);
- умеет применять стандартные алгоритмы и использовать типовые решения задач программирования (ID-28, У2);

- умеет использовать стандартные средства программирования для моделирования объектов и процессов (ID-28, У3).
- умеет определять возможности отображения web-страниц в размерах рабочего пространства устройств для разных видов дизайн-макетов (ID-35, У1);
- умеет применять специализированное программное обеспечение для верстки страниц ИР (ID-35, У2).

Слушатель программы переподготовки будет обладать следующими необходимыми знаниями:
- знает основы web-технологий как инструмента создания информационных систем и сервисов (ID-28, З1);
- знает базовые технологии разработки алгоритмов и программ (ID-28, З2);
- знает основные синтаксические конструкции языков программирования высокого уровня, необходимых для разработки web-сайта (ID-28, З3).
- знает особенности отображения элементов ИР в различных браузерах, синтаксис языка web-программирования (ID-35, З1);
- знает особенности web-программирования на выбранном языке, стандартные библиотеки языка программирования (ID-35, З2).

2. СОДЕРЖАНИЕ ПРОГРАММЫ

2.1 Учебно-тематический план модуля

№ п/п	Наименование разделов/тем	Общая трудоемкость, ч	Всего ауд. ч	По учебному плану с использованием дистанционных образовательных технологий, ч						СРС, ч
				Аудиторные занятия, ч			Дистанционные занятия, ч.			
				лекции	лабораторные работы	практические и семинарские занятия	лекции	лабораторные работы	практические и семинарские занятия	
1	2	3	4	5	6	7	8	9	10	11
1	**Раздел 1. Основы цифровой культуры. Информационные технологии и информационная безопасность**	**2**	**1**	**1**						**1**
2	1.1. Основы цифровой культуры. Классификация цифровых устройств. Информационно-поисковые и справочные системы. Основы защиты информации. Установка и настройка антивирусного ПО.	*2*	*1*	*1*						*1*
3	**Раздел 2. Структура и принципы WWW. Web-редакторы**	**12**	**7**	**3**		**4**				**5**
4	2.1. Основы создания информационных ресурсов. Основы разработки веб-сайтов, интернет-магазинов, сервисов и т.д.	*1,5*	*0,5*	*0,5*						*1*

5	2.2. Организационная структура сети Интернет. Хостинг. Протоколы TCP/IP. Система доменных имен DNS.	*1,5*	*0,5*	*0,5*						*1*
6	2.3. Системы управления контентом - CMS. Возможности CMS. Применение CMS в различных областях деятельности. Принципы, на основе которых разрабатываются CMS.	*3*	*2*			*2*				*1*
7	2.4. Структура и принципы WWW. Клиентские технологии. Серверные технологии. Языки разметки и структурирования информации.	*2*	*1*	*1*						*1*
8	2.5. Современная модель веб-приложения. Подход разделения данных, логики и представления в веб-приложении («Модель – Вид - Поведение» - MVC). Системы управления контентом - CMS. Системы контроля версий (CVS). Системы управления проектами: Jira и другие.	*4*	*3*	*1*		*2*				*1*
9	**Раздел 3. Язык гипертекстовой разметки HTML**	**28**	**20**	**6**		**14**				**8**
10	3.1 Структура Web-технологий. Основные понятия и термины. Принципы проектирования и реализации Web-приложений. Обзор языков и технологий разметки.	*2*	*2*	*2*						
11	3.2 Создание веб-страницы. Структура, теги, атрибуты. Типографика. Цвет. Вставка изображения, видео, звука на страницу. Гиперссылки. Списки. Таблицы	*12*	*8*			*8*				*4*
12	3.3 HTML5 – новый подход. Классификация новых элементов. Семантика. Адаптивный (отзывчивый) дизайн.	*4*	*4*	*4*						
13	3.4 Структура документа. Семантическая модель сайта формата «блог». Элементы формы. Адаптивная разметка. Условные комментарии. Медиазапросы.	*10*	*6*			*6*				*4*
14	**Раздел 4. Каскадные таблицы стилей CSS**	**44**	**28**	**8**		**20**				**16**

15	4.1 Причины и необходимость возникновения технологии CSS. Структура стилевой таблицы. Стилевое правило. Селекторы	*4*	*4*	*4*						
16	4.2 Стилевое правило. Основные селекторы. Блок, свойства блочного элемента. Позиционирование. Разработка и реализация основных элементов страницы («резиновые» элементы, блочные меню, многоуровневые меню и т.п.). Блочные модели страницы (абсолютная, плавающая, комбинированная). Вёрстка страниц, создание сайта.	*18*	*10*			*10*				*8*
17	4.4 Причины и необходимость дополнения стандарта CSS. Новый подход к развитию стилевых таблиц	*4*	*4*	*4*						
18	4.5 Новые CSS-свойства (скруглённые уголки, цветовые функции, тени блоков и текста, трансформации, анимация переходов, колонки, CSS-градиенты, CSS-анимация). Псевдоклассы, псевдоэлементы. Модели FlexBox, Grid Layout. Новый подход к разметке страницы – технология БЭМ.	*18*	*10*			*10*				*8*
19	**Всего**	**86**	**56**	**18**		**38**				**30**

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

3.1 Материально-технические условия реализации программы

Наименование специализированных аудиторий, кабинетов, лабораторий	Вид занятий	Наименование оборудования, программного обеспечения
Компьютерный класс 304/1	лекции, практические занятия	Компьютеры, проектор, экран, доска, Интернет, Интернет-браузеры.

3.2 Учебно-методическое и информационное обеспечение программы

3.2.1 Литература

Основная литература

1. Гуриков С.Р. Интернет-технологии: Учебное пособие / С.Р. Гуриков. - М.: Форум: НИЦ ИНФРА-М, 2015. - 184 с. – Режим доступа: http://znanium.com/go.php?id=488074.

2. Панфилов К. Создание веб-сайта от замысла до реализации. [Электронный ресурс] – М.: ДМК Пресс, 2009. – 440 с. – Режим доступа: http://znanium.com/catalog.php?item= booksearch&code = HTML&page=2

3. Дунаев В. HTML, скрипты и стили / Дунаев В. – 3-е изд., переб. и доп. – СПб.: БХВ-

Петербург, 2011. – 810 с. – Режим доступа: http://znanium.com/catalog.php?bookinfo=350371

4. Дронов В. А. HTML 5, CSS 3 и Web 2.0. Разработка современных Web-сайтов. – СПб.: БХВ-Петербург, 2011. – 414 с. Режим доступа: http://znanium.com/catalog.php?bookinfo=351455

5. Пилгрим М. Погружение в HTML5/ Пилгрим М.- перев. с англ. – СПб.: БХВ-Петербург, 2011. – 294 с. – Режим доступа: http://znanium.com/catalog.php?bookinfo=355295

6. Информационная безопасность и защита информации: Учебное пособие/ Е.К. Баранова, А.В. Бабаш. – 3-е изд. - М.: ИЦ РИОР, НИЦ ИНФРА-М, 2016. - 322 с. Режим доступа: http://znanium.com/go.php?id=495249.

Дополнительная литература

1. Заяц А. М. Основы WEB технологий. Разработка WEB-приложений современными инструментальными средствами: учебно-методическое пособие для бакалавров по направлению подготовки 09.03.02 информационные системы и технологии / Заяц А.М. - Санкт-Петербург: СПбГЛТУ, 2021. - 116 с. URL: https://e.lanbook.com/book/191164.

2. Загуменов, А.П. Как раскрутить и разрекламировать Web-сайт в сети Интернет [Электронный ресурс] / А. П. Загуменов. – М.: ДМК Пресс, 2007. – 384 с. – Режим доступа: http://znanium.com/catalog.php?item=booksearch&code=HTML

3. Попов Е.Б. Профессиональный иностранный язык: английский язык [Электронный ресурс] / Е.Б. Попов. – М.: НИЦ ИНФРА - М, 2016. – 150 с. Режим доступа: http://znanium.com/catalog.php?bookinfo=760140

4. Башлы, П.Н. Информационная безопасность и защита информации [Электронный ресурс] : Учебник / П. Н. Башлы, А. В. Бабаш, Е. К. Баранова. - М.: РИОР, 2013. - 222 с. Режим доступа с сервера ТГАСУ: http://znanium.com/catalog.php?bookinfo=405000

Ресурсы информационно-коммуникационной сети «Интернет»

1. Электронная научно-техническая библиотека ТГАСУ [Электронный ресурс]. – Режим доступа http://tsuab.ru/ru/struktura-tgasu/nt-library/, свободный.

2. Курсы дистанционного обучения ТГАСУ [Электронный ресурс] – Режим доступа: http://ido.tsuab.ru/, сервер ТГАСУ- требуется регистрация.

3. Figma (Фигма) - графический онлайн-редактор для совместной работы для создания веб-дизайна [Электронный ресурс]. – Режим доступа: https://www.figma.com – требуется регистрация.

4. Информационная безопасность (Information Security) – http://www.infosecuritymag.com

5. CMS-платформа создания сайтов, конструктор сайтов Tilda Publishing [Электронный ресурс]. – Режим доступа: https://tilda.cc/ru/ – требуется регистрация.

6. Академия HTML. Сервис интерактивного онлайн-обучения [электронный ресурс] / Режим доступа: http://www.htmlacademy.ru/

7. Справочник CSS. Изучаем с нуля [электронный ресурс] / Режим доступа: http://www.eltisbook.ru/category/css

8. Электронные книги и учебники по веб-дизайну и веб-программированию для начинающих [электронный ресурс] / Режим доступа: https://html5beginner.github.io/books/index.html

9. Справочные и учебные материалы для создания сайтов [электронный ресурс] / Режим доступа: https://basicweb.ru/

10. Электронно-библиотечная система «ZNANIUM.COM» (электронный ресурс; режим доступа http:// http://znanium.com/);

11. Электронно-библиотечная система РУКОНТ (электронный ресурс; режим доступа https://lib.rucont.ru/);

12. Библиотека программиста [электронный ресурс] / Режим доступа: https://hhrd.ru/library/243

3.2.2 Методические разработки

1. Защита информации / сост.: А.Е. Петелин, Ю.П. Еремина; Том. гос. архит.- строит. ун-т. – Томск: Изд-во ТГАСУ, 2012. – 50 с. Режим доступа: http://portal.tsuab.ru/Met_2012_1/m40.pdf

2. Технология создания web-страниц: методические указания к самостоятельной работе и задания к контрольной работе / сост. Э.А. Эшаров, Г.А. Онопенко, Е.Ю. Корниенко. – Томск: Изд-во Том. гос. архит.- строит. ун-та, 2015. – 44 с.

3. Основы каскадных таблиц стилей: методические указания к самостоятельной работе и задания к контрольной работе / сост. Э.А. Эшаров, Г.А. Онопенко, Б.М. Шумилов. – Томск: Изд-во Том. гос. архит.- строит. ун-та, 2016. – 46 с.

3.3 Процедура зачета

Формой промежуточной аттестации является зачет, который проводится в форме теста. К зачету допускаются студенты, прослушавшие лекции и выполнившие практические работы.

Зачтено	Выставляется студенту в случае, если уровень выполнения, предусмотренной зачетом работы, отвечает большинству требований, теоретическое содержание дисциплины освоено полностью или частично, но пробелы не носят существенного характера, необходимые навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено. По результатам итогового теста студент набрал более, чем 70% правильных ответов.
Не зачтено	Выставляется студенту в случае, если уровень выполнения, предусмотренной зачетом работы слабый, уровень выполнения не отвечает большинству требований, теоретическое содержание курса освоено частично, некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения задания не выполнены, либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному. По результатам итогового теста студент набрал менее 70% правильных ответов.

В случае неудовлетворительной оценки студент имеет право пересдать зачет в установленном порядке.

3.4 Кадровое обеспечение

Авторы и преподаватели:

№ п/п	ФИО, должность	Модули (темы, лекции)	Часов, всего
1.	Кульков Д.Н., руководитель отдела ИТ ООО «ТН Диджитал»	Модуль 1. **Раздел 1.** Основы цифровой культуры. Информационные технологии и информационная безопасность. **Раздел 2.** Структура и принципы WWW. Web-редакторы.	14
2.	Онопенко Г.А., зав. кафедрой прикладной математики, профессор	Модуль 1. **Раздел 3.** Язык гипертекстовой разметки HTML	28
3.	Кульков Д.Н., руководитель отдела ИТ ООО «ТН Диджитал»	Модуль 1. **Раздел 3.** Язык гипертекстовой разметки HTML	28
4.	Воронкова И.А., старший преподаватель кафедры прикладной математики	Модуль 1. **Раздел 4.** Каскадные таблицы стилей CSS	44

ПРИЛОЖЕНИЕ 1

АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ МОДУЛЯ

Основы web-технологий

(название модуля)

1. Цели и задачи изучения разделов модуля:

Ознакомление студентов с основами создания информационных ресурсов, разработки веб-сайтов, структуры и принципов работы сети Интернет, возможностями системы управления контентом – *CMS* и её применение. Освоение основных концепций и принципов web-дизайна и технологий проектирования сайтов.

Формирование у слушателей знаний о программировании интернет-приложений, освоение базовых возможностей языка гипертекстовой разметки *HTML* и каскадных таблиц стилей *CSS* для создания веб-сайтов.

ID-28. Применяет языки программирования для решения профессиональных задач
ID-35. Разрабатывает различные веб-архитектуры

3. Планируемые результаты обучения по модулю.

- умеет применять выбранные языки web-программирования и технологии проектирования сайтов (ID-28, У1);
- умеет применять стандартные алгоритмы и использовать типовые решения задач программирования (ID-28, У2);
- умеет использовать стандартные средства программирования для моделирования объектов и процессов (ID-28, У3).
- умеет определять возможности отображения web-страниц в размерах рабочего пространства устройств для разных видов дизайн-макетов (ID-35, У1);
- умеет применять специализированное программное обеспечение для верстки страниц ИР (ID-35, У2).
- знает основы web-технологий как инструмента создания информационных систем и сервисов (ID-28, З1);
- знает базовые технологии разработки алгоритмов и программ (ID-28, З2);
- знает основные синтаксические конструкции языков программирования высокого уровня, необходимых для разработки web-сайта (ID-28, З3).
- знает особенности отображения элементов ИР в различных браузерах, синтаксис языка web-программирования (ID-35, З1);
- знает особенности web-программирования на выбранном языке, стандартные библиотеки языка программирования (ID-35, З2).

4. Тематическое содержание разделов модуля

Основы цифровой культуры. Классификация цифровых устройств. Информационно-поисковые и справочные системы. Основы защиты информации. Установка и настройка антивирусного ПО.

Основы создания информационных ресурсов. Основы разработки веб-сайтов, интернет-магазинов, сервисов и т.д. Организационная структура сети Интернет. Хостинг. Протоколы *TCP/IP*. Система доменных имен *DNS*.

Структура и принципы *WWW*. Клиентские технологии. Серверные технологии. Языки разметки и структурирования информации. Современная модель веб-приложения. Подход разделения данных, логики и представления в веб-приложении («Модель – Вид - Поведение» - *MVC*). Системы управления контентом - *CMS*. Системы контроля версий (CVS). В практикуме дисциплины изучаются основы web-дизайна с помощью веб-редактора *Figma* и технологий проектирования сайтов на CMS-конструкторе *Tilda Publishing*.

Структура web-технологий. Основные понятия и термины. Принципы проектирования и реализации web-приложений. Обзор языков и технологий разметки. Создание веб-страницы. Структура, теги, атрибуты. Типографика. Цвет. Вставка изображения, видео, звука на страницу. Гиперссылки. Списки. Таблицы. *HTML5* – новый подход. Классификация новых элементов. Семантика. Адаптивный (отзывчивый) дизайн. Структура документа. Семантическая модель сайта формата «блог». Элементы формы. Адаптивная разметка. Условные комментарии. Медиазапросы.

Причины и необходимость возникновения технологии *CSS*. Структура стилевой таблицы. Стилевое правило. Селекторы. Стилевое правило. Основные селекторы. Блок, свойства блочного элемента. Позиционирование. Разработка и реализация основных элементов страницы («резиновые» элементы, блочные меню, многоуровневые меню и т.п.). Блочные модели страницы (абсолютная, плавающая, комбинированная). Вёрстка страниц, создание сайта.

Причины и необходимость дополнения стандарта *CSS*. Новый подход к развитию стилевых таблиц. Новые *CSS*-свойства (скруглённые уголки, цветовые функции, тени блоков и текста, трансформации, анимация переходов, колонки, *CSS*-градиенты, *CSS*-анимация). Псевдоклассы, псевдоэлементы. Модели *FlexBox, Grid Layout*. Новый подход к разметке страницы – технология БЭМ.

ПРИЛОЖЕНИЕ 2

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО МОДУЛЮ

Основы web-технологий

Составитель профессор, Г.А. Онопенко
(должность, ФИО, подпись)

Рецензент профессор, Б.М. Шумилов
(должность, ФИО, подпись)

Приложение 2.1. Практические работы.

Раздел 1. Цифровая культура. Информационные технологии и информационная безопасность.

Самостоятельная работа 1. Безопасная работа в интернете: анализ веб-сайтов, безопасный поиск, надежные пароли, электронная почта, платежи.

Цель работы:

- Ознакомиться с правилами безопасной работы в Интернете: как организовать безопасный поиск в Интернете, правила работы с веб-сайтами, надежность паролей и о повышении безопасности работы с браузером
- Изучить основные правила безопасности при работе с электронной почтой и с системами обмена сообщениями, правила безопасности платежей в Интернете. Порядок выполнения работы.

1. Ознакомиться с теоретическими вопросами работы с веб-сайтами в Интернете. Изучить признаки потенциально опасных и мошеннических сайтов.
2. Провести анализ нескольких сайтов в Интернете, пользуясь случайным выбором, подготовить отчет по следующей схеме:

а) внешний вид сайта: качество изображения, отображение кнопок и полей ввода, адресная строка

б) необычное поведение при переходе по страницам сайта

в) настойчивое предложение скачать приложение

г) требования ввести адрес электронной почты, SMS-отправления и пр.

3. Изучить вопрос безопасного поиска в Интернете на примере поисковых систем yandex и google.

а) настройки безопасной работы в поисковой системе google

б) настройки поисковой системы yandex

4. Настройка безопасной работы браузера на примере Mozilla Firefox.

а) работа на собственном и чужом (общественном) компьютере, файлы cookie.

б) настройки безопасной работы, конфиденциальности, защищенного режима.

5. Работа с электронной почтой.

а) регистрация e-mail почтового ящика, требования системы к надежности пароля.

б) восстановление доступа к почтовому ящику, привязка мобильного телефона.

6. Работа с электронными платежными системами.

а) изучите теоретические вопросы и правила использования пластиковых банковских карт при электронных платежах.

б) системы электронных денег: регистрация, правила безопасной работы, надежность паролей, SMS-сообщения, сертификаты пользователей. На примере электронных платежных систем Яндекс.Кошелек и WebMoney.

7. Предъявите результаты работы преподавателю.

Самостоятельная работа 2. Установка, настройка и работа с антивирусом. Диагностика, обновление антивирусных баз, настройка уведомлений в антивирусной программе.

Цель работы:

- Ознакомиться с правилами работы с антивирусом Касперского
- Изучить настройки, обновление антивирусных баз и диагностики в антивирусной программе.

Порядок выполнения работы.

1. Знакомство с аппаратными и системными требованиями антивируса Касперского, проверьте наличие свободного места на компьютере перед установкой программы. Убедитесь, что на компьютере нет других антивирусных программ.
2. Установка и настройка антивирусной программы.

а) используйте для установки Мастер установки

б) изучите интерфейс программы: главное окно, окно настроек, окна статистики и отчетов, окно справочной системы
3. Структура программы и настройки.
а) используйте настройки программы Защита, Поиск вирусов и Сервис
б) настройте постоянную защиту: настройка, управление и обслуживание
в) проведите поиск вирусов: выбор объекта проверки, лечение, удаление вирусов
г) работа с обновлением антивирусных баз, карантином и создание резервного хранилища
4. Работа с антивирусными утилитами

Раздел 2. Структура и принципы WWW. Web-редакторы

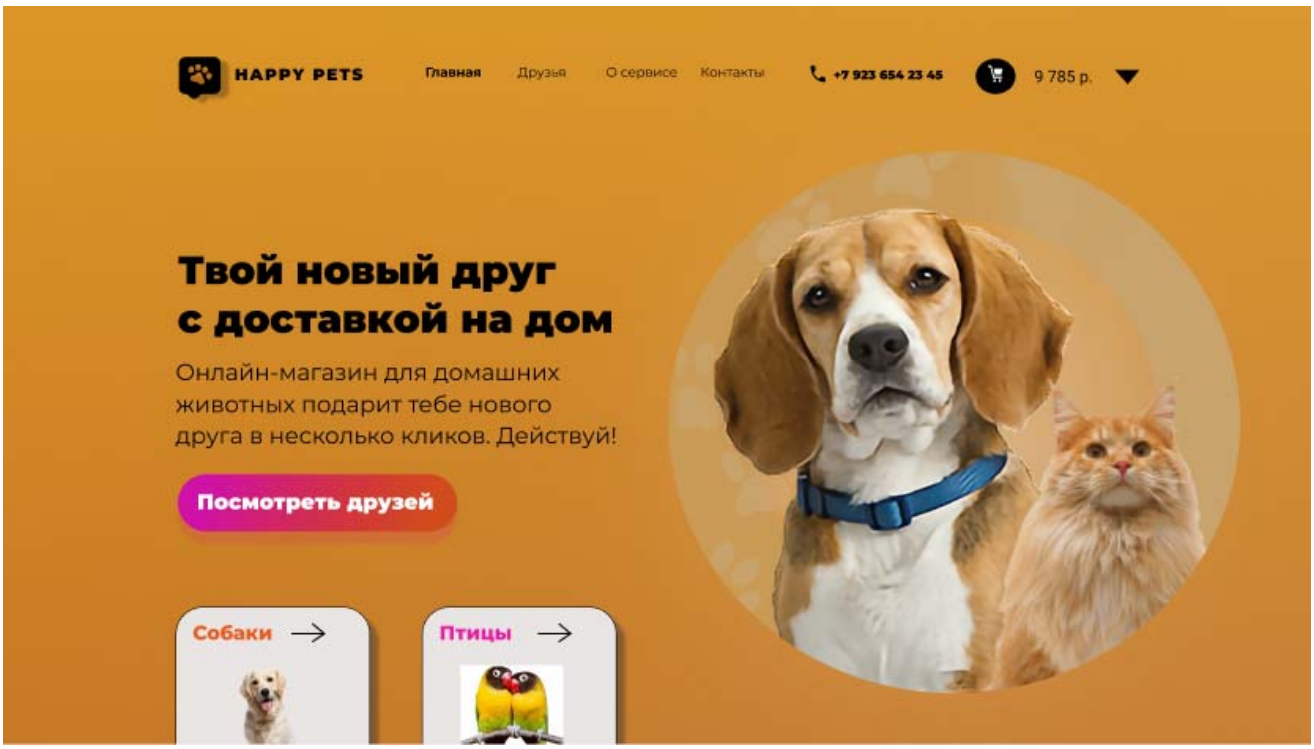
Создание веб-дизайна главного экрана лендинга.

Практическое задание 1. Работа в визуальном редакторе Фигма.

1. Зарегистрировать аккаунт на сайте https://www.figma.com/
2. Посмотреть видео, повторяя дизайн главного экрана сайта о животных за преподавателем.
   Если вы считаете, что можете создать свой дизайн по другой теме, подбору цвета и т.д., то можете сделать это.
3. В текстовой области задания прислать ссылку на ваш дизайн (В конце видео показано, как создать ссылку).

Если в изображении требуется удалить фон, используйте следующие сервисы:
https://www.photoroom.com/ru/instrumenty/ubrat-fon-onlayn
https://www.erase.bg/ru (бесплатно можно делать ограниченное количество файлов).
Если ссылка получается длинная, используйте сокращение ссылок https://clck.ru/.

Пример типового варианта



Создание веб-сайта «Экскурсии по Томску».

Практическое задание 2. Создание лендинга по шаблону в CMS Tilda.

1. Зарегистрировать аккаунт на сайте https://tilda.cc/ru/
2. Посмотреть видео, повторяя создание веб-страницы экскурсий за преподавателем. Вы можете создать лендинг по другой теме и шаблону.
3. В текстовой области задания прислать ссылку на ваш лендинг (опубликовать веб-страницу и прикрепить ссылку в текстовой области задания).

Для работы потребуется открыть следующие сайты:
1. Фотосток https://unsplash.com/
2. Иконки и графика: https://icons8.ru/ или https://www.iconfinder.com/
3. Яндекс картинки: https://yandex.ru/images

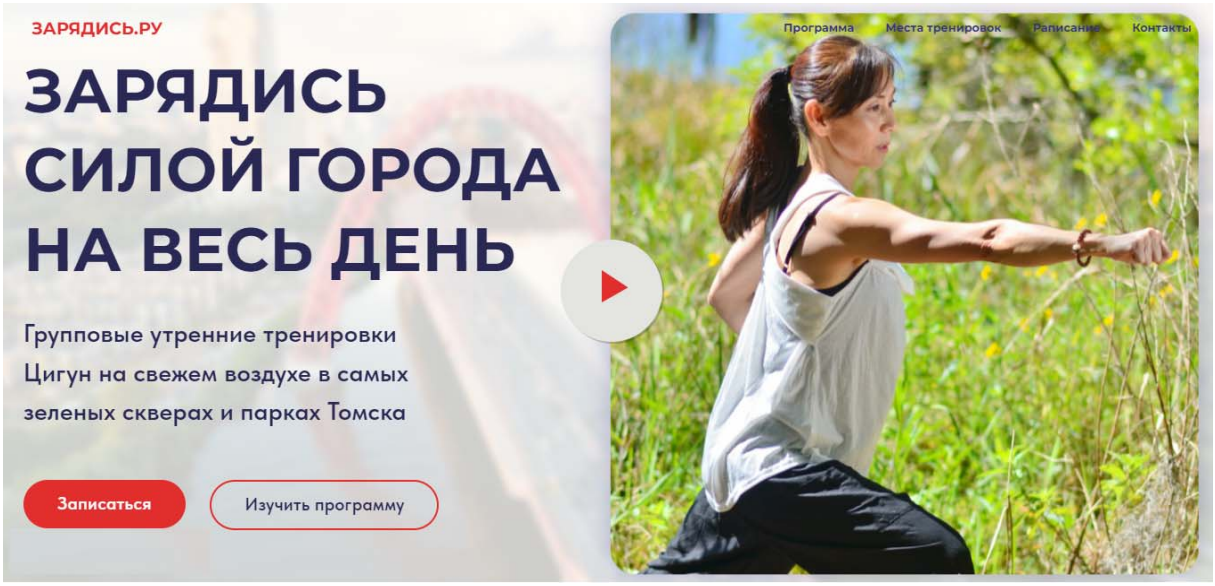
Создание главного экрана веб-сайта «Зарядка в Томске».

Практическое задание 3. Создание главного экрана веб-сайта на Zero Block в CMS Tilda.

1. Зайти в свой аккаунт на сайте https://tilda.cc/ru/
2. Посмотреть видео, повторяя создание главного экрана веб-страницы за преподавателем. Вы можете создать главный экран по другой теме.
3. В текстовой области задания прислать ссылку на вашу работу (опубликовать веб-страницу и прикрепить ссылку в текстовой области задания).

Для работы потребуется открыть следующие сайты:
1. Фотосток https://unsplash.com/
2. Видео-ролики: https://www.youtube.com/ или https://rutube.ru/

Пример типового варианта



Добавление анимации для «Зарядка в Томске» и его адаптации под разные разрешения экрана.

Практическое задание 4. Адаптация главного экрана под разные размеры гаджетов в CMS Tilda.

1. Зайти в свой аккаунт на сайте https://tilda.cc/ru/
2. Открыть свою страницу созданного на практике 3 главного экрана «Зарядка на свежем воздухе в Томске».
3. Посмотреть видео, повторяя все действия за преподавателем.
   Вы можете использовать другие виды анимации и настройки шаблона главного экрана на разных гаджетах.
4. В текстовой области задания прислать ссылку на вашу работу (опубликовать веб-страницу и прикрепить ссылку в текстовой области задания).

Раздел 3. Язык гипертекстовой разметки HTML

Практические задания

Практические задания приведены в методических материалах:
1. Технология создания web-страниц: методические указания к самостоятельной работе и задания к контрольной работе / сост. Э.А. Эшаров, Г.А. Онопенко, Е.Ю. Корниенко. – Томск: Изд-во Том. гос. архит.- строит. ун-та, 2015. – 44 с.

Раздел 4. Каскадные таблицы стилей CSS

Практические задания

Практические задания приведены в методических материалах:
1. Основы каскадных таблиц стилей: методические указания к самостоятельной работе и задания к контрольной работе / сост. Э.А. Эшаров, Г.А. Онопенко, Б.М. Шумилов. – Томск: Изд-во Том. гос. архит.- строит. ун-та, 2016. – 46 с.

Приложение 2.2. Примерные вопросы итогового теста.

Раздел 1. Цифровая культура. Информационные технологии и информационная безопасность.

1. К правовым методам, обеспечивающим информационную безопасность, относятся:
a) Разработка аппаратных средств обеспечения правовых данных
b) Разработка и установка во всех компьютерных правовых сетях журналов учета действий
c) Разработка и конкретизация правовых нормативных актов обеспечения безопасности
2. Основными источниками угроз информационной безопасности являются все указанное в списке:
a) Хищение жестких дисков, подключение к сети, инсайдерство
b) Перехват данных, хищение данных, изменение архитектуры системы
c) Хищение данных, подкуп системных администраторов, нарушение регламента работы
3. Виды информационной безопасности:
a) Персональная, корпоративная, государственная
b) Клиентская, серверная, сетевая
c) Локальная, глобальная, смешанная
4. Цели информационной безопасности – своевременное обнаружение, предупреждение:
a) несанкционированного доступа, воздействия в сети
b) инсайдерства в организации - чрезвычайных ситуаций
5. Основные объекты информационной безопасности:
a) Компьютерные сети, базы данных
b) Информационные системы, психологическое состояние пользователей
c) Бизнес-ориентированные, коммерческие системы
6. Основными рисками информационной безопасности являются:
a) Искажение, уменьшение объема, перекодировка информации
b) Техническое вмешательство, выведение из строя оборудования сети
c) Потеря, искажение, утечка информации
7. К основным принципам обеспечения информационной безопасности относится:
a) Экономической эффективности системы безопасности
b) Многоплатформенной реализации системы
c) Усиления защищенности всех звеньев системы
8. Основными субъектами информационной безопасности являются:
a) руководители, менеджеры, администраторы компаний
b) органы права, государства, бизнеса - сетевые базы данных, фаерволлы
9. К основным функциям системы безопасности можно отнести все перечисленное:
a) Установление регламента, аудит системы, выявление рисков
b) Установка новых офисных приложений, смена хостинг-компании
c) Внедрение аутентификации, проверки контактных данных пользователей тест
10. Принципом информационной безопасности является принцип недопущения:
a) Неоправданных ограничений при работе в сети (системе)
b) Рисков безопасности сети, системы
c) Презумпции секретности
11. Принципом политики информационной безопасности является принцип:
a) Невозможности миновать защитные средства сети (системы)
b) Усиления основного звена сети, системы
c) Полного блокирования доступа при риск-ситуациях
12. Принципом политики информационной безопасности является принцип:
a) Усиления защищенности самого незащищенного звена сети (системы)
b) Перехода в безопасное состояние работы сети, системы
c) Полного доступа пользователей ко всем ресурсам сети, системы
13. Принципом политики информационной безопасности является принцип:
a) Разделения доступа (обязанностей, привилегий) клиентам сети (системы)
b) Одноуровневой защиты сети, системы
c) Совместимых, однотипных программно-технических средств сети, системы
14. К основным типам средств воздействия на компьютерную сеть относится:
a) Компьютерный сбой
b) Логические закладки («мины»)

c) Аварийное отключение питания
15. Когда получен спам по e-mail с приложенным файлом, следует:
a) Прочитать приложение, если оно не содержит ничего ценного - удалить
b) Сохранить приложение в парке «Спам», выяснить затем IP-адрес генератора спама
c) Удалить письмо с приложением, не раскрывая (не читая) его
16. Принцип Кирхгофа:
a) Секретность ключа определена секретностью открытого сообщения
b) Секретность информации определена скоростью передачи данных
c) Секретность закрытого сообщения определяется секретностью ключа
17. ЭЦП – это:
a) Электронно-цифровой преобразователь
b) Электронно-цифровая подпись
c) Электронно-цифровой процессор
18. Наиболее распространены угрозы информационной безопасности корпоративной системы:
a) Покупка нелицензионного ПО
b) Ошибки эксплуатации и неумышленного изменения режима работы системы
c) Сознательного внедрения сетевых вирусов
19. Наиболее распространены угрозы информационной безопасности сети:
a) Распределенный доступ клиент, отказ оборудования
b) Моральный износ сети, инсайдерство
c) Сбой (отказ) оборудования, нелегальное копирование данных
20. Наиболее распространены средства воздействия на сеть офиса:
a) Слабый трафик, информационный обман, вирусы в интернет
b) Вирусы в сети, логические мины (закладки), информационный перехват
c) Компьютерные сбои, изменение администрирования, топологии
21. Утечкой информации в системе называется ситуация, характеризуемая:
a) Потерей данных в системе
b) Изменением формы информации
c) Изменением содержания информации
22. Свойствами информации, наиболее актуальными при обеспечении информационной безопасности являются:
a) Целостность
b) Доступность
c) Актуальность
23. Угроза информационной системе (компьютерной сети) – это:
a) Вероятное событие - Детерминированное (всегда определенное) событие
b) Событие, происходящее периодически
24. Информация, которую следует защищать (по нормативам, правилам сети, системы) называется: - Регламентированной
a) Правовой
b) Защищаемой
25. Разновидностями угроз безопасности (сети, системы) являются все перечисленное в списке:
a) Программные, технические, организационные, технологические
b) Серверные, клиентские, спутниковые, наземные
c) Личные, корпоративные, социальные, национальные
26. Окончательно, ответственность за защищенность данных в компьютерной сети несет:
a) Владелец сети
b) Администратор сети
c) Пользователь сети
27. Политика безопасности в системе (сети) – это комплекс:
a) Руководств, требований обеспечения необходимого уровня безопасности
b) Инструкций, алгоритмов поведения пользователя в сети
c) Нормы информационного права, соблюдаемые в сети
28. Наиболее важным при реализации защитных мер политики безопасности является:
a) Аудит, анализ затрат на проведение защитных мер
b) Аудит, анализ безопасности
c) Аудит, анализ уязвимостей, риск-ситуаций

Раздел 2. Структура и принципы WWW. Web-редакторы

1. Тег – это:
- фрагмент текста на Web-странице
- адрес Web-документа
- адрес компьютера в сети
- команда языка программирования HTML

2. Выравнивание текста осуществляется при помощи тега:
- valign
- right
- left
– align

3. Тег используется для:
- задания параметров фона
- установления параметров шрифта во фрагменте текста
- настройки параметров стиля
- установления параметров основного шрифта на Web-странице

4. Для того, чтобы установить на Web-странице отображение текста определенным шрифтом, используется атрибут тега:
- kbd
- code
- type
- cite

5. При добавлении ячейки в строки таблиц используется тег:
- td
- tr
- table
- add

6. Добавление бегущей строки в Web-документ осуществляется тегом:
- runstring
- stroke
- marquee
- target

7. Вставка рисунка в Web-документ осуществляется тегом:
- photo
- img
- picture
- videoimg

8. К редакторам, в рамках которых может формироваться текст Web-документа, относятся:
- WordPad, FrontPage Express, Notepad, MS Word
- FrontPage Express, HomeSite, PowerPoint
- MS Word, MS Excel, CorelDraw
- MS Windows, MS Word, HomeSite

9. Для выделения активного участка карты используется тег:
- domain
- shape
- square
- area

10.Базовый протокол Интернета
a) TCP/IP
b) HTTP
c) FTP

11.URL – это ...

a) прикладной протокол
b) название языка, на котором создаются Web-страницы
c) адрес ресурса в сети Интернет
12. Какая программа из пакета MS Internet Explorer предназначена для общения по электронной почте
a) Обозреватель
b) Outlook Express
c) NetMeeting
13.Отметьте правильные адреса электронной почты
a) moymail@rambler.ru
b) comp.os@manager
c) znanie.info.news
d) 2002km@mail.net.ru
e) kom_23@nsk.su
14.Обозреватель Internet Explorer может работать по протоколу
a) POP3
b) SMTP
c) HTTP
d) IMAP
15. Гиперссылки – это ...
a) активные элементы Web-страниц
b) пассивные элементы Web-страниц
c) элементы не Web-страниц
16.Отметьте правильные IP-адреса
a) 125.34.2
b) 125.34.12.1
c) 168.257.89.11
d) 11.0.0.0
e) 157.255.45.7
17. Для сохранения графического элемента с Web-страницы в файл следует воспользоваться командой
a) Файл/Сохранить как...
b) Сохранить на панели инструментов
c) Сохранить рисунок как... из контекстного меню
18.Web-страница – это файл с расширением
a) htm
b) doc
c) ftp
d) exe
19. После отправки сообщения его копия помещается в папку
a) Входящие
b) Исходящие
c) Отправленные
20. С помощью ссылки на Web-странице можно обратиться к
a) любому ресурсу Интернет
b) только к другим Web-страницам
21. Страница, загружаемая при запуске браузера является
a) Домашней страницей сайта
b) Домашней страницей браузера
c) Последней страницей, с которой осуществлялась работа в предыдущем сеансе
22.Рисунки из Интернета загружаются
a) быстрее текста

b) медленнее текста
c) одинаково
23. Организация, предоставляющая доступ к Интернету, называется
a) Обозревателем
b) РосНИИРос
c) Провайдером
d) Сервис-центром
24. Приходящая корреспонденция помещается в папку
a) Отправленные
b) Входящие
c) Исходящие
25. При сохранении Web-страницы на диск с помощью команды
a) Файл/Сохранить как... в Internet Explorer
b) сохранится только текст
c) сохранится текст и рисунки
d) сохранятся все элементы
26. Отметьте правильные адреса Web-страниц
a) http://www.home.dom.ru/index.htm
b) http:\\www.narod.ru\default.htm
c) www://yandex.ru/main.htm
d) http://groups.com
e) http://ftp://www.mail.ru/index.htm
27. FTP – это протокол
a) передачи почты на сервер
b) передачи файлов
c) передачи гипертекстовых документов
28. Браузер – программа-клиент для работы с
a) World Wide Web
b) Internet Explorer
c) Outlook Express
d) электронной почтой
29. Протокол – это
a) программа для общения с помощью электронной почты
b) список доступных серверов
c) правила передачи данных в сети
30.Сервер – это
a) компьютер клиента сети Интернет
b) компьютер, управляющий сетевыми ресурсами
c) почтовый робот

Раздел 3. Язык гипертекстовой разметки HTML

1. Верный синтаксис комментария?
1. <!-- комментарий -->
2. /* комментарий */
3. # комментарий

2. Что заключается между тегами <title></title>?
1. Всплывающая подсказка.
2. Заголовок изображения.
3. Заголовок страницы.
3. Наименование тега, с помощью которого создаётся форма для ввода данных пользователем?
1. body

2. form
3. html

4. Являются ли идентичными для браузера теги <DIV> <Div>?
1. Да
2. Нет
3. Теги не существуют.

5. Наименование тега для подключения внешней таблицы стилей.
1. script
2. link
3. style

6. Что означает тег ?
1. Пункт маркированного списка.
2. Пункт нумерованного списка.
3. Все ответы верные.

7. Выбирите тег для создания раскрывающегося списка.
1. <select>
2. <input>
3. <label>

8. Назначение тега <fieldset>?
1. Создаёт поле для ввода текста.
2. Группирует связанные элементы в форме.
3. Определяет заголовок для формы ввода.

9. Что обозначает запись <input required />?
1. Создание поля для ввода пароля.
2. Создание обязательного для заполнения поля для ввода текста.
3. Создание поля для ввода текста, которое временно будет отключенным.

10. Выберите тег для создания гиперссылки.
1. <link>
2. <label>
3. <a>

11. Что означает тег <noscript>?
1. Данная страница не поддерживает выполнение скриптов.
2. Выполнение скриптов отключено.
3. Код выполняемый при отключенном скрипте.

12. Выбирите теги для добавления звукового контента.
1. <src></src>
2. <audio></audio>
3. <source></source>

13. Что означает тег
?
1. Небольшая цитата
2. Фрагмент текста не имеющий важности
3. Перевод строки в тексте

14. Укажите верный способ отправки формы методом GET.
1. <form name="get"></form>
2. <form action="get"></form>
3. <form method="get"></form>

15. Что означает атрибут form?
1. Элемент является формой для отправки данных
2. Указывает на имя формы, с которой связан элемент
3. Указывает на действие формы, которое будет заменено

16. С помощью какого атрибута указывается языковая принадлежность элемента?
1. lang
2. media
3. language

17. Что означает selected?
1. Ничего не обозначает, такого тега или атрибута нет
2. Атрибут тега <option>, указывающий на выбранный элемент
3. Тег содержащий раскрывающийся список элементов для выбора

18. Значение тега <textarea>?
1. Логически выделяет область документа, как содержащую текстовое содержимое
2. Предназначен для вывода большого объёмы текста
3. Определяет многострочный элемент управления для ввода текста

19. Каково значение colspan?
1. Определяет количество строк в теге <textarea>
2. Указывает на количество ячеек таблицы, объединяемых по горизонтали
3. Определяет количество столбцов в таблице, создаваемой с помощью <table>

20. Какой тег определяет текст, как заголовок?
1. <h2>
2. <h1>
3. <h4>

21. Какой тег определяет заключенный в него текст, как цитату?
1. <abbr>
2. <dl>
3. <blockquote>

22. Назначение тега <nav>?
1. Определяет секцию документа, содержащую элементы навигации
2. Представляет элемент с выпадающим списком, содержащий элементы навигации
3. Определяет место в документе, куда перейдёт пользователь, по связанной с ним ссылке навигации

23. Что означает тег <figcaption>?
1. Подпись или заголовок, описывающий содержимое тега <figure>
2. Указывает на заголовок таблицы, заключенной в <table>
3. Определяет описание изображения заключенное в <img>, которое показывается при неудачной загрузки изображения

Раздел 4. Каскадные таблицы стилей CSS

1. Что устанавливает свойство background-attachment?
1. Будет ли фоновое изображение прокручиваемым
2. Будет ли фоновое изображение повторяться
3. Устанавливает как будет обрезано фоновое изображение

2. Каким свойством устанавливается насыщенность (жирность) шрифта?
1. font-style
2. font-weight

3. font-size

3. Назначение свойства text-indent?
1. Устанавливает расстояние между буквами
2. Устанавливает расстояние между словами
3. Задаёт отступ «красной строки»

4. С помощью какого свойства можно изменить регистр букв в слове?
1. text-transform
2. text-overflow
3. text-decoration

5. В каком из ответов создаётся зачеркнутый текст?
1. text-overflow: ellipsis
2. text-transform: capitalize
3. text-decoration: line-through

6. Каким свойством задаётся ширина (толщина) границ элемента?
1. border-width
2. width
3. line-width

7. Какое влияние на элемент окажет добавление border-style: dashed?
1. Граница элемента отобразится линией из точек
2. Граница элемента отобразится пунктирной линией
3. Никакого, так как такого значения не существует

8. Какое описание соответствует назначению элемента :last-child?
1. Определяет стиль последнего дочернего элемента родителя
2. Определяет стиль первого дочернего элемента родителя
3. Определяет стиль нечётных дочерних элементов родителя

9. Укажите верный ответ применения правила div > p { color: red; } ... Параграф1 Параграф2
1. Текст Параграф1 и Параграф2 будет красного цвета
2. Фон текста Параграф1 и Параграф2 будет красного цвета
3. Текст Параграф1 будет красного цвета

10. Назначение псевдокласса :disabled?
1. Отключает элемент
2. Применяет стиль к элементу, когда тот находится в отключенном состоянии
3. Делает элемент невидимым

11. Укажите верный ответ применения стиля p + span { color: red; } ... span1 paragraph1 span2 paragraph2
1. Текст span1 и span2 будет красного цвета
2. Текст paragraph2 будет красного цвета
3. Текст span2 будет красного цвета

12. Укажите верный синтаксис определения правила.
1. div { color: red; }
2. div = [color=>red]
3. div { var color = red; }

13. Укажите верный способ включения стиля на страницу.
1. <head> <link href="style/colors.css" rel="stylesheet" type="text/css"> </head>
2.

3. <head> <style> div { color: red; } </style> </head>

14. Назначение свойства background-clip?
1. Определяет как фоновое изображение или цвет фона будут выводиться под границами блока
2. Определяет как фоновое изображение будет подстраиваться (растягиваться, сжиматься, повторяться) под размеры блока
3. Определяет количество повторений фонового изображения, которое полностью не заполнило размеры блока

15. Назначение свойства list-style-type?
1. Определяет будет ли списочный элемент, например <ol>, являться блочным или встраиваемым типом
2. Определяет стиль маркера списочного элемента
3. Свойства не существует

16. Что означает свойство overflow-x?
1. Выравнивание по горизонтали (по оси x)
2. Задаёт поведение блока, когда содержимое не помещается в его размеры
3. Предназначен для изменения алгоритма расчёта ширины элемента

17. Допустимые значения свойства display?
1. block
2. table
3. flex

18. Укажите верный ответ применения правила p ~ span { color: red; } ... span1 Параграф1 span2 span3
1. Красным текстом отобразится span2 и span3
2. Красным текстом отобразится span1
3. Красным текстом отобразится span2

19. Стиль какого элемента определяет псевдокласс :only-child?
1. Применяется только к дочерним элементам родителя первого уровня вложенности
2. Применяется к дочернему элементу, если он единственный у родителя
3. Применяется к родительскому элементу, имеющему хотя бы одно потомка

20. Назначение свойства width?
1. Устанавливает ширину элемента
2. Выравнивает элемент по ширине
3. Устанавливает высоту элемента

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ АРХИТЕКТУРНО-СТРОИТЕЛЬНЫЙ УНИВЕРСИТЕТ»

Институт непрерывного образования

СОГЛАСОВАНО
Директор ИНО

_____ Н.Р.Шадейко
(подпись)

"____"_____20___ г.

УТВЕРЖДАЮ
Проректор по учебной работе

_____ Д.Н. Песцов
(подпись)

"____"_____20___ г.

ПРОГРАММА МОДУЛЯ

Введение в web-программирование
(наименование дисциплины)

ДОПОЛНИТЕЛЬНАЯ ПРОФЕССИОНАЛЬНАЯ ПРОГРАММА

«Web-программирование»
(наименование программы)

Томск 2025

Составитель *доцент кафедры ПМ, А.В. Титов*
(Должность, Ф.И.О., подпись)

Рецензент *профессор, Б.М. Шумилов*
(Должность, Ф.И.О., подпись)

Введена в действие с «___»_____20 г.

СОДЕРЖАНИЕ

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ 4
1.1 Учебный план 4
1.2 Планируемые результаты обучения 4
2. СОДЕРЖАНИЕ ПРОГРАММЫ 5
2.1 Учебно-тематический план 5
3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ 6
3.1 Материально-технические условия реализации программы 6
3.2 Учебно-методическое и информационное обеспечение программы 6
3.2.1 Литература 6
3.2.2 Методические разработки 7
3.3 Процедура зачета 7
3.4 Кадровое обеспечение 8
ПРИЛОЖЕНИЕ 1 9
ПРИЛОЖЕНИЕ 2 10

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ

1.1 Учебный план

Наименование дисциплины	Общая трудоем-кость, час.	Всего, ауд. час.	Аудиторные занятия, час.			СРС, час.	Текущий контроль*, (шт.)			Промежуточная аттестация	
			Лекции	Лабораторные работы	Прак. занятия, семинар		РК, РГР, реф.	КР	КП	Зачет	Экзамен
Модуль 2. Введение в web-программирование											
Основы алгоритмизации и программирования	14	8	4		4	6					
Программирование на JavaScript	52	34	10		24	18					
Итого	**66**	**42**	**14**		**28**	**24**				**Зачёт**	

1.2 Планируемые результаты обучения

Слушатель программы переподготовки будет обладать следующими профессиональными компетенциями:

Матрица компетенций дисциплины

Профессиональные компетенции	ID-28	ID-35
	Применяет языки программирования для решения профессиональных задач	Разрабатывает различные веб-архитектуры

Слушатель программы переподготовки будет способен осуществлять следующие трудовые действия:

- применять языки программирования для решения профессиональных задач (ID-28);
- разрабатывать различные веб-ориентированные решения (ID-35).

Слушатель программы переподготовки будет обладать следующими необходимыми знаниями:

- знает базовые технологии разработки алгоритмов и программ (ID-28, З2);
- знает основные синтаксические конструкции языков программирования высокого уровня, необходимых для разработки web-сайта (ID-28, З3);
- знает особенности отображения элементов информационных ресурсов (ИР) в различных браузерах, синтаксис языка web-программирования (ID-35, З1);
- знает особенности web-программирования на выбранном языке, стандартные библиотеки языка программирования (ID-35, З2).

Слушатель программы переподготовки будет обладать следующими необходимыми умениями:

- умеет применять стандартные алгоритмы и использовать типовые решения задач программирования (ID-28, У2);
- умеет использовать стандартные средства программирования для моделирования объектов и процессов (ID-28, У3);
- умеет определять возможности отображения web-страниц в размерах рабочего пространства устройств для разных видов дизайн-макетов (ID-35, У1);
- умеет применять специализированное программное обеспечение для верстки страниц информационных ресурсов (ИР) (ID-35, У2).

2. СОДЕРЖАНИЕ ПРОГРАММЫ

2.1 Учебно-тематический план

№ п/п	Наименование разделов/тем	Общая трудоемкость, ч	Всего ауд. ч	По учебному плану с использованием дистанционных образовательных технологий, ч						СРС, ч
				Аудиторные занятия, ч			Дистанционные занятия, ч.			
				лекции	лабораторные работы	практические и семинарские занятия	лекции	лабораторные работы	практические и семинарские занятия	
1	2	3	4	5	6	7	8	9	10	11
1	**Раздел 5. Основы алгоритмизации и программирования**									
2	Понятие алгоритма, его свойства и основные характеристики	*3*	*2*	*1*		*1*				*1*
3	Способы записи алгоритмов	*3*	*2*	*1*		*1*				*1*
4	Основные виды алгоритмов и способы их представления	*4*	*2*	*1*		*1*				*2*
5	Знакомство с языком JavaScript. Решение задач.	*4*	*2*	*1*		*1*				*2*
6	**Итого**	**14**	**8**	**4**	**-**	**4**	**-**	**-**	**-**	**6**
7	**Раздел 6. Программирование на JavaScript**									
8	Назначение и возможности языка. Основные алгоритмические конструкции. DOM. События, обработчики событий. Динамический HTML	*4*	*4*	*4*						
9	Динамическое построение веб-страницы. Обработка событий. Простейшие клиентские приложения, игры. Управление элементами страницы.	*16*	*10*			*10*				*6*
10	Основные понятие ООП: объект, свойства, методы; инкапсуляция, наследование. Способы реализации ООП в JavaScript. Классы.	*2*	*2*	*2*						

11	Создание и использование объекта в JS. Описание класса. Примеры создания и использования классов.	*10*	*4*			*4*				*6*
12	Назначение и возможности JS-библиотек и фреймворков	*4*	*4*	*4*						
13	Синтаксис библиотеки/фреймворка. Управление элементами страницы (поиск, удаление, добавление, изменение содержимого и оформления). Обработчики событий: создание и удаление. Элементы пользовательского интерфейса.	*16*	*10*			*10*				*6*
14	**Итого**	**52**	**34**	**10**	**-**	**24**	**-**	**-**	**-**	**18**
15	**Всего**	**66**	**42**	**14**		**28**				**24**

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

3.1 Материально-технические условия реализации программы

Наименование специализированных аудиторий, кабинетов, лабораторий	Вид занятий	Наименование оборудования, программного обеспечения
Компьютерный класс 304/1	лекции, практические занятия	Компьютеры, проектор, экран, доска, Интернет, Интернет-браузеры. Текстовые редакторы.

3.2 Учебно-методическое и информационное обеспечение программы

3.2.1 Литература

Основная литература

1. Семакин, И.Г. Основы алгоритмизации и программирования: Учебник / И.Г. Семакин. - М.: Academia, 2017. - 384 с.
2. Серкова, Е.Г. Основы алгоритмизации и программирования: практикум / Е.Г. Серкова. - РнД: Феникс, 2019. - 189 с.
3. Янцев В.В. JavaScript и PHP. Content management system: учебное пособие для вузов / Янцев В.В. - Санкт-Петербург: Лань, 2022. - 192 с. URL: https://e.lanbook.com/book/266657.

Дополнительная литература

1. Канцедал, С.Е. Алгоритмизация и программирование: учебное пособие /С.Е. Канцедал. – Москва: Издательский Дом "ФОРУМ": ООО "Научно-издательский центр ИНФРА-М", 2018. – 352 с. Режим доступа: http://znanium.com.
2. Колдаев, В.Д. Основы алгоритмизации и программирования: Учебное пособие / В.Д. Колдаев. - М.: Форум, 2015. - 352 с.
3. Парфилова, Н.И. Программирование: Основы алгоритмизации и программирования: Учебник / Н.И. Парфилова; Под ред. Трусова Б.Г. - М.: Academia, 2018. - 32 с.

4. Брокшмидт, К., К. Введение в разработку приложений для Windows 8 с использованием HTML, CSS и JavaScript [Электронный ресурс]. - Национальный Открытый Университет «ИНТУИТ», Москва; 2016; http://biblioclub.ru/index.phppage=book&id=428973
5. Сычев, А. В; Перспективные технологии и языки веб-разработки [Электронный ресурс]. - Национальный Открытый Университет «ИНТУИТ», Москва; 2016; http://biblioclub.ru/index.phppage=book&id=429078
6. Зудилова, Т. В; Web-программирование JavaScript [Электронный ресурс]. - Университет ИТМО, Санкт-Петербург; 2012; http://www.iprbookshop.ru/65749.html
7. Кулькова, Л. И; Задачи и упражнения по JavaScript: учебное пособие [Электронный ресурс]. - Российский университет дружбы народов, Москва; 2018; http://www.iprbookshop.ru/104199.html
8. Рындин, Н. А. Технологии разработки клиентских WEB-приложений на языке JavaScript: учебное пособие [Электронный ресурс]. - Воронежский государственный технический университет, ЭБС АСВ, Воронеж; 2020; http://www.iprbookshop.ru/108188.html

Ресурсы информационно-коммуникационной сети «Интернет»

1. Электронная научно-техническая библиотека ТГАСУ [Электронный ресурс]. – Режим доступа http://tsuab.ru/ru/struktura-tgasu/nt-library/, свободный.
2. Курсы дистанционного обучения ТГАСУ [Электронный ресурс] – Режим доступа: http://ido.tsuab.ru/, сервер ТГАСУ- требуется регистрация.
3. Академия HTML. Сервис интерактивного онлайн-обучения [электронный ресурс] / Режим доступа: http://www.htmlacademy.ru/
4. Электронные книги и учебники по веб-дизайну и веб-программированию для начинающих [электронный ресурс] / Режим доступа: https://html5beginner.github.io/books/index.html
5. Справочные и учебные материалы для создания сайтов [электронный ресурс] / Режим доступа: https://basicweb.ru/

3.2.2 Методические разработки

1. Технология создания web-страниц: методические указания к самостоятельной работе и задания к контрольной работе / сост. Э.А. Эшаров, Г.А. Онопенко, Е.Ю. Корниенко. – Томск: Изд-во Том. гос. архит.- строит. ун-та, 2015. – 44 с. Режим доступа: http://portal.tsuab.ru/Metodi_/2015/met038_2015.pdf.

3.3 Процедура зачета

Формой промежуточной аттестации является зачет, который проводится в форме теста. К зачету допускаются студенты, прослушавшие лекции, выполнившие практические работы.

Зачтено	Выставляется студенту в случае, если уровень выполнения, предусмотренной зачетом работы, отвечает большинству требований, теоретическое содержание дисциплины освоено полностью или частично, но пробелы не носят существенного характера, необходимые навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено. По результатам итогового теста студент набрал более, чем 70% правильных ответов.
Не зачтено	Выставляется студенту в случае, если уровень выполнения, предусмотренной зачетом работы слабый, уровень выполнения не отвечает большинству требований, теоретическое содержание курса освоено частично, некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения задания не выполнены, либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному. По результатам итогового теста студент набрал менее 70% правильных ответов.

В случае неудовлетворительной оценки студент имеет право пересдать зачет в установленном порядке.

3.4 Кадровое обеспечение

Авторы и преподаватели:

№ п/п	ФИО, должность	Модули (темы, лекции)	Часов, всего
1.	Титов А.В., доцент кафедры прикладной математики	Модуль 2. **Раздел 5.** Основы алгоритмизации и программирования **Раздел 6.** Программирование на JavaScript	66
2.	Кульков Д.Н., руководитель отдела ИТ ООО «ТН Диджитал»	Модуль 2. **Раздел 6.** Программирование на JavaScript	52

ПРИЛОЖЕНИЕ 1

АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ МОДУЛЯ

Введение в web-программирование

(название модуля)

1. Цели и задачи изучения модуля:

Формирование у слушателей знаний по построению алгоритмов и разработке веб-программ при решении задач профессиональной деятельности, освоение принципов алгоритмизации и создание программ на языке JavaScript.

Изучение особенностей языка JavaScript, его сильных и слабых сторон, наиболее востребованных задач, решаемых при помощи языка JavaScript. Изучение основных понятий языка, операций с ними, базовых концепций. Получение базовых практических навыков разработки программного обеспечения на языке JavaScript.

2. Содержание компетенций, формируемых при изучении модуля:

ID-28. Применяет языки программирования для решения профессиональных задач

ID-35. Разрабатывает различные веб-ориентированные решения

3. Планируемые результаты обучения по модулю:

- знает базовые технологии разработки алгоритмов и программ (ID-28, З2);
- знает основные синтаксические конструкции языков программирования высокого уровня, необходимых для разработки web-сайта (ID-28, З3);
- знает особенности отображения элементов информационных ресурсов (ИР) в различных браузерах, синтаксис языка web-программирования (ID-35, З1);
- знает особенности web-программирования на выбранном языке, стандартные библиотеки языка программирования (ID-35, З2).
- умеет применять стандартные алгоритмы и использовать типовые решения задач программирования (ID-28, У2);
- умеет использовать стандартные средства программирования для моделирования объектов и процессов (ID-28, У3);
- умеет определять возможности отображения web-страниц в размерах рабочего пространства устройств для разных видов дизайн-макетов (ID-35, У1);
- умеет применять специализированное программное обеспечение для верстки страниц информационных ресурсов (ИР) (ID-35, У2).

4. Тематическое содержание модуля:

Понятие алгоритма, его свойства и основные характеристики. Виды алгоритмов и их реализация. Способы и средства записи алгоритмов. Алгоритмы и структуры данных. Методы сортировки. Методы поиска данных. Знакомство с языком JavaScript. Элементы математической логики. Программирование с использованием массивов. Работа со структурами.

Назначение и возможности языка. Основные алгоритмические конструкции. DOM. События, обработчики событий. Динамический HTML Динамическое построение веб-страницы. Обработка событий. Простейшие клиентские приложения, игры. Управление элементами страницы. Основные понятие ООП: объект, свойства, методы; инкапсуляция, наследование. Способы реализации ООП в JavaScript. Классы. Создание и использование объекта в JS. Описание класса. Примеры создания и использования классов. Назначение и возможности JS-библиотек и фреймворков. Синтаксис библиотеки/фреймворка. Управление элементами страницы (поиск, удаление, добавление, изменение содержимого и оформления). Обработчики событий: создание и удаление. Элементы пользовательского интерфейса.

ПРИЛОЖЕНИЕ 2

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО МОДУЛЮ

Введение в web-программирование

Составитель доцент, А.В. Титов
(должность, ФИО, подпись)

Рецензент профессор, Б.М. Шумилов
(должность, ФИО, подпись)

Приложение 2.1. Практические задания и методические указания.

Раздел 1. Основы алгоритмизации и программирования

Линейный алгоритм

Линейным называется алгоритм, в котором всегда выполняются все действия строго последовательно.

Как правило, алгоритмы линейной структуры состоят из трех частей: ввод исходных данных, вычисления результатов по формулам, вывод значений результатов. Это самые простые алгоритмы.

ПРИМЕР 1. Найти сторону и диагональ квадрата, если известна его площадь.

Математическая постановки задачи. Введем математические обозначения величин: *a* - сторона квадрата, *d* - его диагональ, s - площадь.

Исходными данными задачи является только величина s (она должна быть известна, т.е. введена в начале алгоритма). Величины *a, d* являются результатом решения задачи.

Теперь надо вспомнить формулы, связывающие эти величины:

$s = a\,2$ - формула площади квадрата,

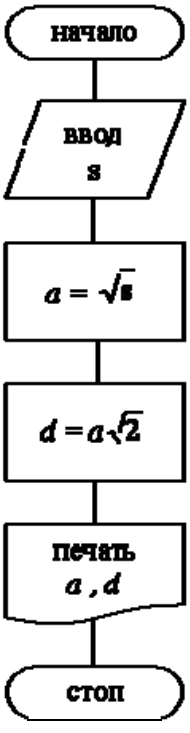
$a\,2 + a\,2 = d\,2$ - теорема Пифагора.

Выбор метода решения задачи. Метод очень простой - прямое вычисление сначала стороны квадрата, а затем его диагонали по формулам:

$a = \sqrt{s}$, $d = a\sqrt{2}$.

Разработка алгоритма. Схема алгоритма решения задачи приведена на рисунке. Каждое элементарное действие обозначено в схеме отдельным символом - блоком.

В дальнейшем условимся понятие "символ" графической схемы называть более привычным словом "блок".



Порядок выполнения блоков, как уже было сказано, определяют линии потока. По схеме видно, что при вводе совершенно различных исходных данных (по смыслу задачи, конечно, s>0) все действия этого алгоритма будут выполняться всегда, и порядок их выполнения никогда не изменяется.

Для отладки линейного алгоритма достаточно сравнить результаты его исполнения с результатами ручного решения задачи.

Разветвляющаяся структура

Разветвляющимся называется алгоритм, при выполнении которого каждый раз последовательность действий может быть разная, т.е. каждый раз выбирается один из нескольких путей прохождения схемы алгоритма. Конкретный путь прохождения алгоритма называется ветвью алгоритма. Схема подобного алгоритма обязательно содержит хотя бы один блок (символ) "решение", который и обеспечивает *разветвление* вычислительного процесса.

ПРИМЕР 2. Вычислить значение функции *y* по формуле

$$y = \frac{a^2 + 3}{x - 1}$$.

Математическая постановка задачи. Из условия задачи ясно, что все величины имеют математические обозначения, известна формула для вычислений.

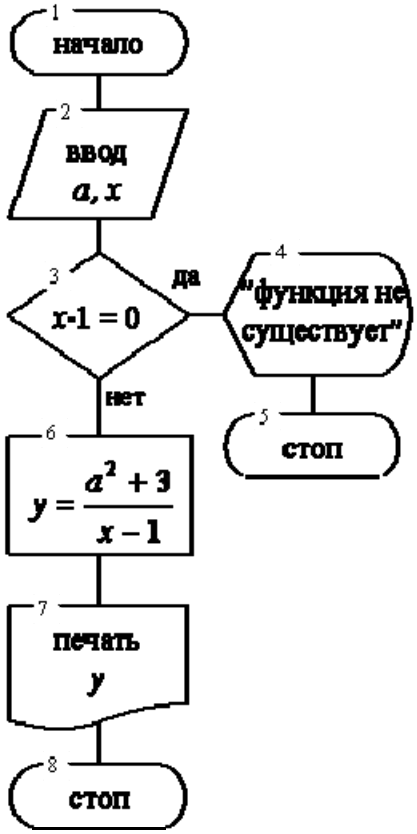
Исходными данными задачи являются переменные *a, x.*

Результат задачи - переменная *y*.

Остается выяснить ограничения для исходных данных. Область определения переменных *a, x* - вся числовая ось. Но есть одно значение переменной *x*, при котором функция *y* не может быть вычислена (при *x*=1 знаменатель обращается в нуль, делить на нуль нельзя).

Выбор метода решения задачи. Прежде чем вычислять функцию *y*, выясним, не обращается ли в нуль знаменатель дроби.

Разработка алгоритма. Схема алгоритма решения задачи приведена на рисунке ниже.



В блоке 3 проверка:

- если знаменатель равен нулю, то на экран выдается сообщение, что при заданных исходных данных функция не существует (блок 4), и решение задачи заканчивается (это одна ветка алгоритма);
- если знаменатель не равен нулю, вычисляем и печатаем значение функции *y* (блоки 6, 7, 8 - это вторая ветка алгоритма).

Особенность отладки разветвляющихся алгоритмов состоит в следующем: для проверки правильности всех ветвей алгоритма тест должен включать несколько наборов исходных данных - по числу ветвей алгоритма.

Одним из методов проверки правильности алгоритма является его ***трассировка*** (*trace* - след). Она заключается в тщательном, скрупулезном выполнении алгоритма вручную на примере конкретных исходных данных из всей области их определения. В ходе такой проверки должны быть установлены по крайней мере два факта:

- при выборе исходных данных из одной и той же области определения ход выполнения алгоритма всегда один и тот же;
- при выборе исходных данных из разных областей определения ход выполнения алгоритма разный.

Такие утверждения дают право предполагать, что алгоритм составлен правильно. При трассировке схемы удобно записывать пути ее прохождения для последующего анализа - будем последовательно записывать номера блоков, которые выполняются фактически.

Выполним трассировку схемы (рис.6). Выберем два разных набора исходных данных:

a=1, *x*=1. Путь: блоки 1, 2, 3, 4, 5.

a=1, *x*=2. Путь: блоки 1, 2, 3, 6, 7, 8.

От значения переменной *a* путь прохождения схемы не зависит, поэтому мы взяли одно и то же число. Выбор разных значений переменной *x* привел к получению разных путей прохождения схемы. Если взять любое другое значение (например, *x*=3), то мы получим второй вариант пути. Итак, схема алгоритма имеет 2 разные ветки.

Конечно, эти рассуждения могут показаться элементарными и ненужными, но умение выполнять трассировку схемы поможет выявить ошибки при разработке схемы алгоритма решения большой и сложной задачи.

ПРИМЕР 3. Найти наибольшее из трех чисел.

Математическая постановки задачи. Введем математические обозначения величин: пусть *A, B, C* - заданные числа (это исходные данные задачи, их значения будем вводить), *M* - наибольшее (максимальное) из чисел *A, B, C* (это искомая величина, результат решения задачи, его значение будем печатать).

Выбор метода решения задачи.

Первый вариант. Сначала сравним первые два числа, а затем большее из них сравним с третьим числом. То число, которое окажется большим после второго сравнения, и есть максимальное.

Разработка алгоритма.

Схема алгоритма решения задачи приведена на рис.1.

В блоке 3 происходит сравнение чисел *A* и *B*. Если *A* окажется больше *B*, то управление передается по ветке "да" и в блоке 4 число *A* сравнивается с числом *C*.

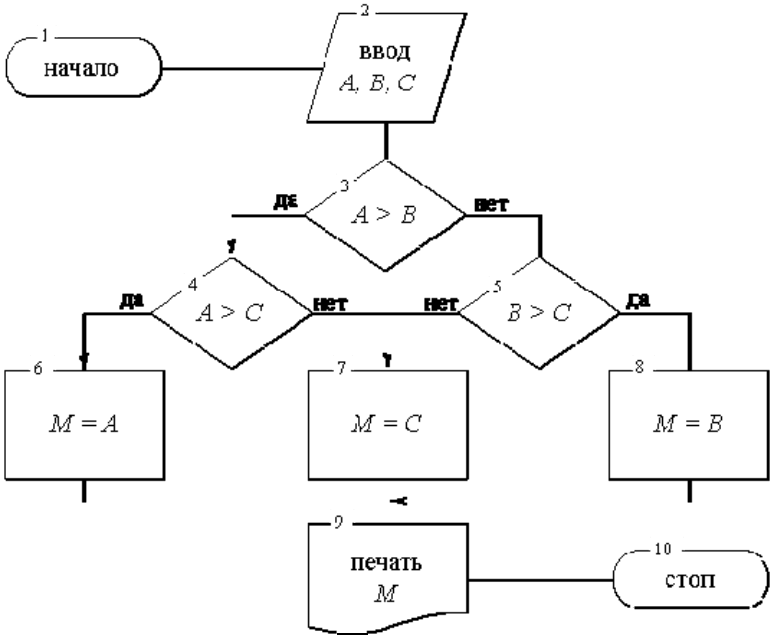


Рис. 1.

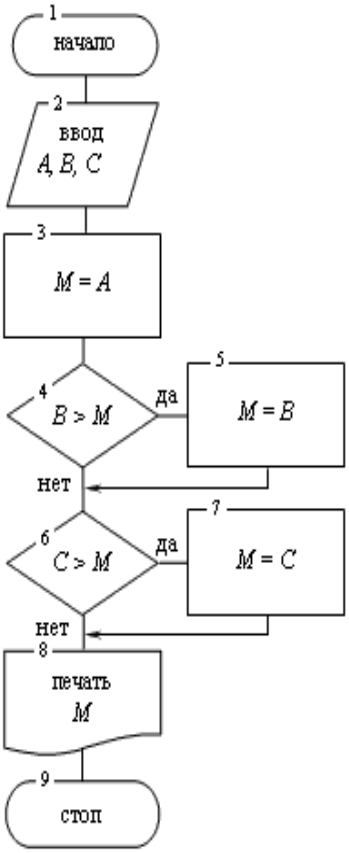


Рис. 2

Для полной трассировки схемы подберем 4 набора исходных данных, т.к. на схеме прослеживается 4 разных ветки. Проверим их:

A=1, *B*=2, *C*=3.	Путь:	1,	2,	3,	5,	7,	9,	10.
A=1, *B*=3, *C*=2.	Путь:	1,	2,	3,	5,	8,	9,	10.
A=2, *B*=1, *C*=3.	Путь:	1,	2,	3,	4,	7,	9,	10.

A=3, *B*=2, *C*=1. Путь: 1, 2, 3, 4, 6, 9, 10.

Действительно, все пути прохождения схемы разные, а какой-либо другой набор исходных данных даст один из вышеуказанных путей.

Второй вариант. Как правило, любая задача имеет не один способ решения. Даже для этой, на первый взгляд, простой задачи можно предложить еще один вариант. Предположим, что первое число и есть максимальное (присвоим его значение числу *M*). Затем проверим, не окажется ли следующее число больше числа *M*. Если это так, то заменим значение *M* на это число. Такую же проверку проведем и для третьего числа.

Разработка алгоритма

Схема алгоритма решения задачи приведена на рис.2. В блоках 4, 5 происходит возможное уточнение *M* по результату сравнения со вторым числом *B*. В блоках 6, 7 происходит возможное уточнение *M* по результату сравнения с третьим числом *C*.

Схема имеет 4 разные ветки. Попробуйте самостоятельно подобрать 4 разных набора исходных данных для полной трассировки схемы.

Идея, лежащая в основе этого алгоритма, позволяет найти максимальное из любого количества чисел. Для этого достаточно действия 4 и 5 повторять для каждого числа. Как организовать повторение действий? Надо построить цикл. Переходим к рассмотрению следующей структуры алгоритма - циклической.

Циклическая структура

Циклическим называется алгоритм, который содержит участок, выполняющийся многократно, каждый раз с новыми значениями переменных, изменяющихся по одним и тем же законам.
По способу организации циклы делятся на два основных вида:

- циклы с известным заранее числом повторений (*классические*);
- циклы с неизвестным числом повторений (*итерационные*).

Классический цикл организуется с помощью специальной переменной, которая называется параметром цикла. *Параметр цикла* - это числовая переменная, которая управляет работой цикла. Она изменяется по закону арифметической прогрессии, что обеспечивает повторение цикла нужное количество раз. Для этого заранее должны быть известны:

начальное значение параметра (обозначим его $t_{нач}$);

конечное значение параметра (обозначим его $t_{кон}$);

- шаг изменения параметра (обозначим его Δt).

Зная эти 3 величины, можно вычислить количество повторений цикла по формуле:

$$k = \left[\frac{t_{кон} - t_{нач}}{\Delta t} \right] + 1.$$

В этой формуле квадратные скобки обозначают, что после деления берется только целая часть числа (дробная часть всегда отбрасывается, а не округляется), т.к. количество повторений цикла - это целая величина. Классический цикл имеет 4 части:

- *подготовка цикла* - параметру цикла присваивается начальное значение;
- *тело цикла* - основные действия, которые повторяются каждый раз, на каждом витке цикла;
- *изменение параметра* цикла на величину шага;
- *условие выхода* из цикла (или, напротив - *условие повторения* цикла) - проверка параметра на конечное значение.

От взаимного расположения этих частей зависит вид типовой схемы одиночного цикла. На рис.3 представлен цикл типа "до" (с условием выхода из него), а на рис.4 - цикл типа "пока" (с условием повтора). Оба типа абсолютно равнозначны. На этапе алгоритмизации задачи безразлично, какой из них Вы выберете. Разница между ними обнаруживается только на этапе программирования, т.к. каждому типу цикла соответствуют разные циклические операторы.

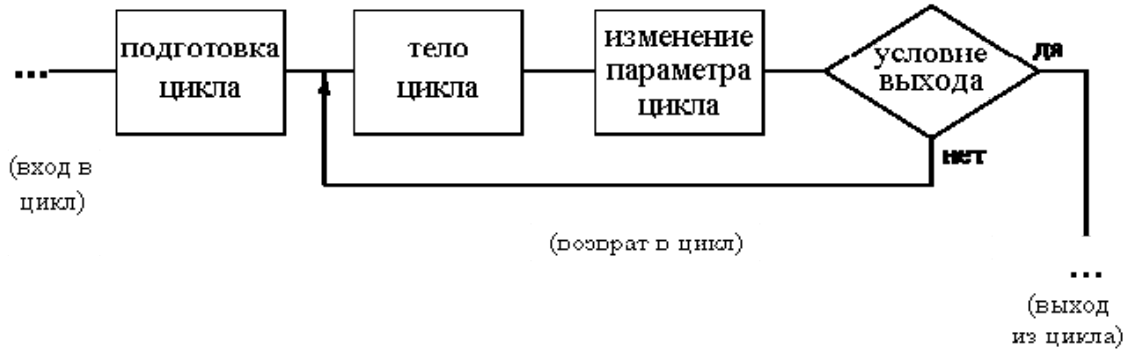


Рис. 3.

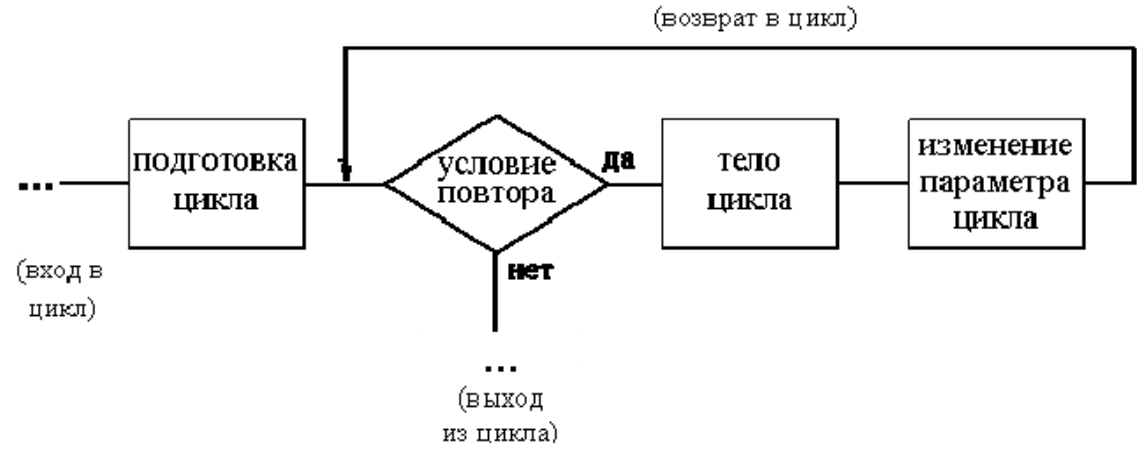


Рис. 4.

ПРИМЕР 4. Построить таблицу значений функции $y = 2sin(x2\text{-}1)$ на заданном интервале изменения аргумента.

Математическая постановка задачи. Введем недостающие обозначения:

(a,b) - интервал;

h - шаг изменения аргумента функции.

Все три величины являются исходными данными задачи. Результат задачи - значения функции, вычисляемые на каждом витке цикла.

Выбор метода решения задачи. На каждом витке цикла будем выбирать точку (значение аргумента), вычислять в ней функцию и печатать оба значения в таблицу.

Разработка алгоритма. В качестве параметра цикла удобно использовать сам аргумент функции, т.к. он изменяется на интервале *(a,b)* с постоянным шагом *h* (т.е. по закону арифметической прогрессии). Тогда начальным значением параметра будет число *a*, а конечным

значением - число *b.* В теле цикла - 2 действия: вычисление функции и печать. Схема алгоритма решения задачи приведена на рис. 5. Определим части цикла на схеме:

- подготовка цикла - блок 3,
- изменение параметра - блок 6,
- условие выхода из цикла - блок 7.
- тело цикла легче всего определять в последнюю очередь (это повторяющиеся в цикле действия, которые не касаются организации цикла). В этом алгоритме тело цикла составляют блоки 4, 5.

Знать принципы организации циклического алгоритма и уметь определять части цикла на схеме (даже "незнакомой" задачи) очень полезно. Это значительно облегчит Вам в дальнейшем процесс программирования алгоритма. Как правило, части цикла, касающиеся его организации (связанные с параметром цикла), описываются во всех языках программирования специальными операторами организации цикла. А действия, составляющие тело цикла, выделяются в отдельную группу операторов. Так что Вы сможете избежать ошибок программирования, если сейчас научитесь определять тело цикла на схеме алгоритма.

Для обозначения циклического процесса есть специальный блок - это символ "подготовка".

Внутри блока указываются параметр цикла, его начальное и конечное значения и шаг изменения в виде

$$t = t_{нач}, t_{кон}, \Delta t$$

Тело цикла обычно располагается ниже и более четко просматривается на схеме.

На рис.5 приведена типовая схема организации одиночного цикла с помощью блока "подготовка".

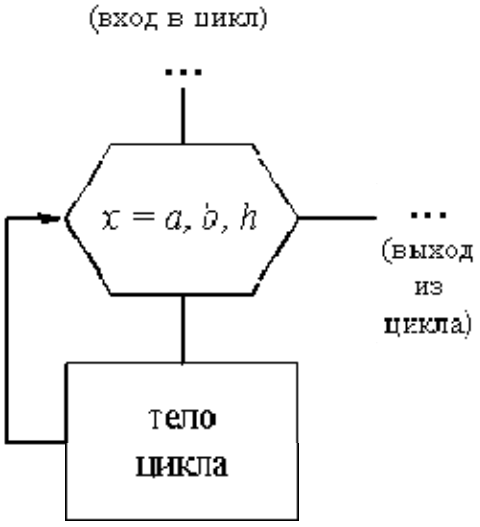


Рис. 5

На рис.7 приведена схема решения задачи табулирования функции с помощью блока "подготовка". Сравните с предыдущей, "развернутой" схемой (рис.6).

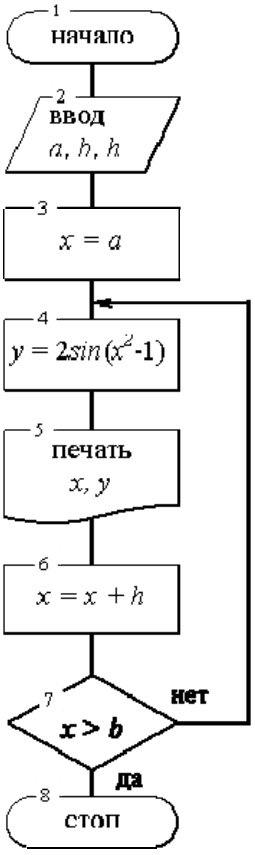


Рис. 6

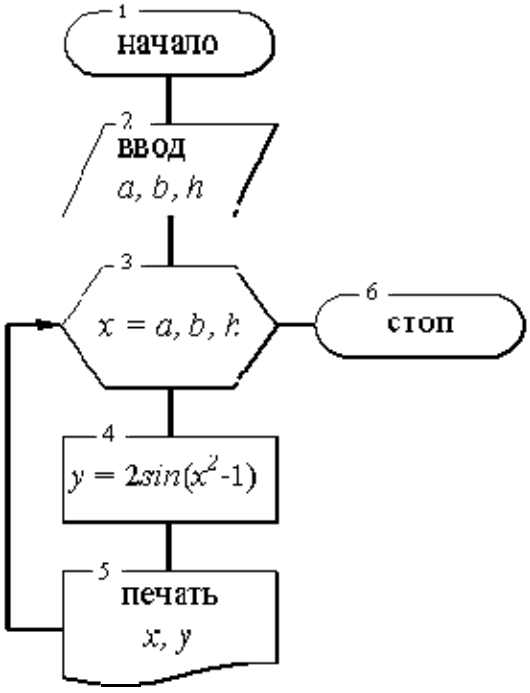


Рис. 7

Тело цикла на этой схеме составляют блоки 4, 5.

Количество повторений цикла (оно определяет количество строк в таблице значений функции) вычислим по формуле

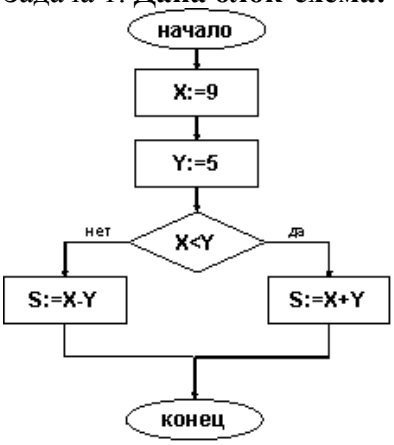
$$k - \left[\frac{b-a}{h}\right] + 1$$

после задания конкретных значений исходных данных.

Конечно, использование символа "подготовка" значительно сокращает размер циклической схемы и упрощает ее восприятие, но говорить о преимуществе того или иного варианта схемы не стоит, иногда при программировании более полезным оказывается именно "развернутый" вариант схемы. Все зависит от того, какими операторами описания цикла будет реализована конкретная схема.

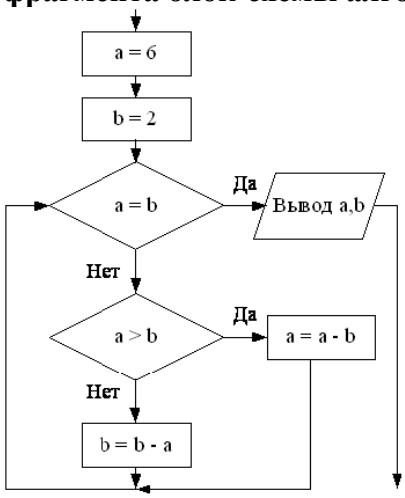
Задачи на алгоритмы.

Задача 1. **Дана блок-схема:**



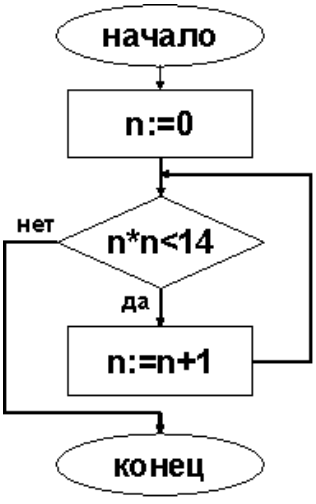
чему равно значение переменной S ?

Задача 2. **В результате работы фрагмента блок-схемы алгоритма**



какие значения примут a и b?

Задача 3. **Дана блок-схема:**

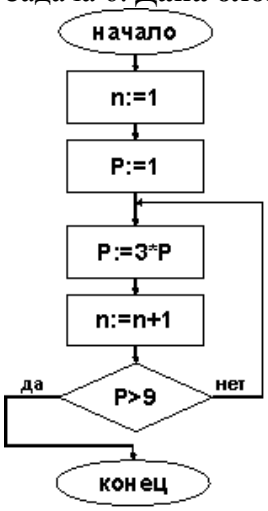


чему равно значение переменной n?

Задача 4. **Вася действует по следующему алгоритму:**
Шаг 1. Пройти 5 м прямо.
Шаг 2. Повернуть направо.
Шаг 3. Повторять шаги 1-2, пока не будет пройдено 40 м.
Шаг 4. Остановиться.
Какое расстояние после выполнения шага 4 будет от начальной точки до точки, из которой Вася начал свое движение?

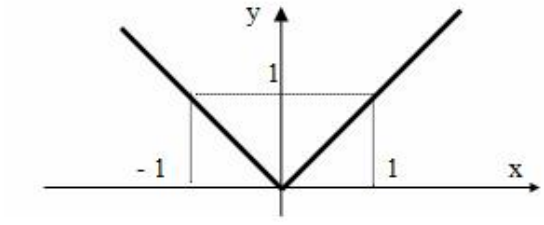
Задача 5. **Вася действует по следующему алгоритму:**
Шаг 1. Купить стакан сока за 20 рублей.
Шаг 2. Выпить сок и подождать 1 час.
Шаг 3. Повторять шаги 1-2, пока остается более 10 рублей.
Шаг 4. Пойти домой.
Если у Васи изначально было 30 руб., то домой он пойдет через ___ часа

Задача 6. **Дана блок-схема:**



какое значение примет переменная n?

Задача 7. Составьте алгоритм для вычисления значения функции, график которой изображён на рисунке:



Вопросы и задания

21. Логические задачи:

а) Фальшивая монета.

Имеется 9 монет одинакового достоинства. Известно, что 8 из них имеют одинаковый вес, а одна, фальшивая, немного легче остальных. Требуется за 2 взвешивания на чашечных весах определить без гирь фальшивую монету.

б) Бочонок с квасом.

Двум приятелям привезли 8-и литровый бочонок с квасом. Как им разделить квас поровну, если в их распоряжении имеются только две пустые банки ёмкостью 5 и 3 литра?

в) Три поджаренных ломтика.

Чтобы поджарить ломтик хлеба, приходится тратить 30 сек. на одну сторону, а затем столько же на вторую. На моей сковороде умещается лишь два ломтика. Как поджарить три ломтика хлеба только за 1,5 минуты, а не за 2 минуты, как получилось бы, если не предпринять некоторую хитрость?

г) Распределите воду.

В ведре 10 литров воды. Напишите алгоритм, как с помощью 3-х литровой банки и 7-и литрового бидона распределить воду так, чтобы в ведре и в бидоне оказалось по пять литров воды. Составьте алгоритм с наименьшим числом переливаний.

д) Песочные часы.

Имеются двое песочных часов на 8 мин. и на 3 мин. Чтобы приготовить эликсир бессмертия, его требуется варить ровно 7 мин. Как это сделать?

е) Тупик.

Два встречных поезда, в каждом из которых локомотив и 21 вагон, встретились на дороге с одним тупиком. Тупик вмещает 11 вагонов или 10 вагонов и локомотив. Как поездам разъехаться (то есть, как должны маневрировать машинисты, чтобы каждый поезд продолжал движение в своём направлении)? Составьте графический алгоритм.

ж) Переправа.

Папа с двумя сыновьями отправились в поход. На их пути встретилась река. У берега плот. Он выдержит на воде одного папу или двоих ребят. Как переправиться на другой берег папе с сыновьями?

з) Отгадай число.

Задумано натуральное пятизначное число. Опишите алгоритм, позволяющий за позволяющий за минимальное число вопросов отгадать его, если ответы на вопросы даются в форме «да» и «нет».

22. Какова характерная форма организации действий в линейных алгоритмах?

23. Даны величины A, B, C, D. Составьте алгоритм для перемещения значений этих величин по такому принципу: B должно получить значение A; C- значение B, D - значение C.

24. Дана последовательность чисел в1, в2, в3, в4. Составьте алгоритм перемещения этих чисел по правилу: каждому из них присвоить значение соседнего с ним справа, а последнему - значение первого числа.

25. В обувной магазин поступила партия обуви различных размеров и различной цены. Данные представлены в таблице:

Размер обуви	Количество пар	Цена одной пары (в рублях)
38	120	30
39	50	40
41	93	90

Составьте алгоритм определения новых цен на обувь и стоимости всей партии обуви после повышения цен в N раз.

26. Используя следующие данные меню обеда в столовой:

Блюдо	Цена в рублях
Борщ	3,5
Котлета	4
Каша	2
Чай	0,3

Составьте алгоритм определения общей стоимости обеда в столовой, и во сколько раз возрастёт стоимость обеда, если цена котлеты увеличится вдвое.

27. Какая форма организации действий называется ветвлением?

28. В чём сходства и различия между полной и неполной формами ветвлений?

29. Запишите алгоритм решения квадратного уравнения на естественном языке и с помощью блок-схемы.

30. Заданы два числа ***к*** и ***m*** . Переменной n присвоить значение ***1***, если***k<m***; значение ***0***, если ***k=m***; значение ***-1***, если ***k>m***. Алгоритм составьте с помощью блок-схемы.

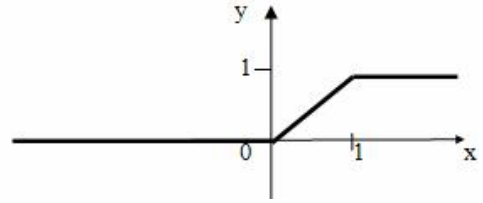
31. Задано число ***p***. Если ***p<0***, то ***y=p+5***, если ***p>0***, то ***y=p-5***, в противном случае решения нет. Составьте алгоритм решения этой задачи помощью блок-схемы.

32. Запишите в виде алгоритмов правила определения знака:

а) произведения двух действительных чисел;

б) суммы двух действительных чисел.

33. Составьте алгоритм для вычисления значения функции, график которой изображён на рисунке:



34. Составьте алгоритм на естественном языке и на языке блок-схем для вычисления значения y при некоторых значениях A и C:

$$y=\begin{cases} A, \text{ если } A=C \\ A+C, \text{ если } A>C \\ C-A, \text{ если } A<C \end{cases}$$

35. Построить алгоритм решения следующей задачи: “Ракета запускается с точки на экваторе Земли со скоростью y (км/сек) в направлении движения Земли по орбите вокруг Солнца. Каков будет результат этого запуска, если он зависит от скорости y по таким правилам:

если $y \le 7{,}8$ км/сек, то ракета упадёт на землю;
если $7{,}8 < y \le 11{,}2$ км/сек, то ракета станет спутником Земли;
если $11{,}2 < y \le 16{,}4$ км/сек, то ракета станет спутником Солнца;
если скорость ракеты больше 16,4 км/сек,
то она покинет солнечную систему.

36. Составьте алгоритм для вычисления значения выражения: $\sqrt{\frac{1+\cos y}{1-\cos y}} + \sqrt{\frac{1+\cos x}{1-\cos x}}$, где $y=tgx$, $x=60^0$

37. Составьте алгоритм для вычисления значения объёма куба с произвольной стороной.

38. Составьте алгоритм вычисления площади поверхности S и объёма V конуса, заданного диаметром основания D и длиной образующей L.

39. Составьте алгоритмы для задач:

а) Даны три стороны треугольника A,B и C. Как определить, является ли треугольник равносторонним.

б) Даны два числа A и B. Как определить максимум из этих двух чисел. Учтите и вариант, когда нет максимума.

40. Составьте алгоритм для вычисления У при некоторых значениях A,B и X:

$$y=\begin{cases} \sin(2A)^2+B, & \text{если } X<A \\ \cos^2(A+B), & \text{если } X>=A \end{cases}$$

Составьте алгоритм для вычисления У при некоторых значениях A,C и X:

$$y=\begin{cases} \sqrt{A+C}, & \text{если } X<C \\ A, & \text{если } X>C \\ |C-A|, & \text{если } X=C \end{cases}$$

42. Составьте алгоритм, с помощью которого можно определить, поместится ли круг произвольного радиуса в квадрат с произвольной стороной.

43. Составьте алгоритм, определяющий в какой четверти на плоскости находится точка с координатами (x, y).

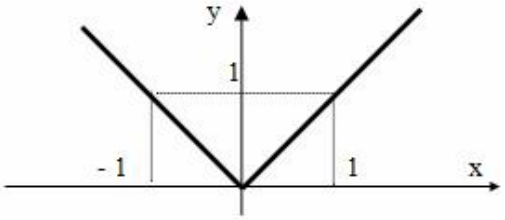
44. Составьте алгоритм для нахождения числа корней уравнения вида:
$x^2 = a+1$ в зависимости от значения параметра a.

45. Составьте алгоритм для нахождения числа корней уравнения вида:
$x^2 = a$ в зависимости от значения параметра a.

46. Составьте алгоритм к задаче:
Определить, взяв за исходные данные стороны треугольника, острый, тупой или прямой один из углов (например, угол C) треугольника ABC.

47. Составьте алгоритм, который проверял бы у ученика таблицу умножения по трём вопросам, причём, номер вопроса обозначьте переменной N, а количество правильных ответов - переменной K.

48. Составьте алгоритм для вычисления значения функции, график которой изображён на рисунке:



49. Составьте алгоритмы для решения задач:

а) Даны три произвольных числа. Определите все чётные из них.

б) Даны пять произвольных чисел. Определите только отрицательные и кратные четырём.

в) Дано 10 чисел. Из двух первых нечётных чисел меньшее замените произведением, а большее - разностью первого и второго. В случае, если нечётных чисел среди них нет, выдайте сообщение об этом.

50. Назовите известные вам виды циклов.

51. Чем отличаются структуры циклов ***до*** и ***пока***?

52. Что такое параметр цикла?

53. Чем отличаются входной язык и объектный?

54. Каковы назначения компилятора и интерпретатора?

55. Перечислите этапы решения задачи на ЭВМ.

56. Где находится программа во время её исполнения?

57. В чём идея визуального программирования?

58.Чем отличаются функции системных и прикладных программистов?

59. Составьте алгоритмы для решения задач:

а) Спортсмен пробежал в первый день 10 км. Каждый следующий день он увеличивал дневную норму на 10% от нормы предыдущего дня. Какой путь он пробежит на седьмой день?

б) Найдите произведение 25 произвольных вещественных чисел;

в) В ЭВМ вводятся роста учеников класса из 30 человек. Найдите средний рост класса.

60. Составьте алгоритм для вычисления значений функции

$y=2\sin^2 x - \cos x^2$, если $x \in [0, 2pi]$ с шагом pi/2.

61. Составьте алгоритм для вычисления значений функции

$y=Ax^2+\cos(x-B)$, если $x \in [0, 3.5]$ с шагом 0.5,
а A и B - произвольные натуральные числа.

62. Составьте алгоритм для определения среднемесячной температуры апреля.

63. Составьте алгоритм для решения задач:

а) Найдите произведение любых двух натуральных чисел а и в, если исполнитель её не умеет выполнять умножение, а умеет лишь складывать.

б) Продавщица Иванова просит составить алгоритм, в результате которого при вводе цены весового продукта выводилась бы таблица стоимости 50,100,150,200,...10000 грамм.

в) Одноклеточная амёба каждые 3 часа делится на 2 клетки. Определите, сколько клеток будет через 3, 6, 9, 12...24 часа.

64. Составьте алгоритм для вычисления значений функции

$y=50x^2 + x$, если $x \in [-20, 120]$ с шагом 5.

65. Составьте алгоритм, определяющий площади полной поверхности каждого из двух цилиндров, если диаметры их оснований соответственно равны 0.6 и 2 см, а их высоты равны 1 и 1.5 см.

66. Составьте алгоритм для вычисления значения y, если

$$y = 1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{6} + \ldots + \frac{1}{20}$$

67. Составьте алгоритм для определения количества различных сигналов, если их может быть подано *m* флажками различных цветов. Отличие сигналов заключается в порядке расположения разноцветных флажков. Решить при *m=6* (задача сводится к нахождению *m!*).

68. Составьте алгоритм для решения задач:

а) На каждом следующем дне рождения Винни Пух съедает столько же пищи, сколько на двух предыдущих. На двух первых днях рождения у Пятачка и Кролика он съел по 100 грамм пищи.

Сколько же пищи Винни Пух съест на пятнадцатом дне рождения?

б) В старину изобретателю игры в шахматы была предложена любая награда. Мудрец попросил на первую клетку шахматной доски положить одно зерно пшеницы, на вторую - в 2 раза больше, чем на первую, на третью - в 2 раза программу, подсчитывающую, сколько зёрен получил мудрец, и сколько это килограмм пшеницы, если 10 зёрен весит 1 грамм?

в) В ЭВМ вводятся фамилии и роста 400 человек. Выведите на экран фамилии тех, чей рост не меньше 170 см.

г) Среди 15 произвольных вещественных чисел найдите количество отрицательных.

д) Среди 100 целых чисел подсчитайте количество нечётных.

69. Составьте алгоритм для нахождения количества чётных, нечётных и равных нулю чисел из списка 15 значений.

70. Составьте алгоритм для вычисления суммы всех двузначных нечётных чисел.

71. Составьте алгоритм, вычисляющий сумму 30 членов последовательности, заданной рекуррентно:

$$a_1=2,\ a_{n+1}=2a_n-1$$

72. Составьте алгоритм для нахождения числа дней в месяце, если даны:*номер месяца N* - целое число от 1 до 12, *целое число A*, если год високосный A=1, если не високосный, то A=0.

73. В обращении имеются монеты 1-, 5-, 10-, 50- копеечного достоинства. Составьте алгоритм, определяющий сдачу наименьшим количеством монет.

74. Составьте алгоритм, определяющий, какое число чаще всего выпадало в 10 тиражах игры «Спортлото 5 из 36». Покажите само число и частоту его появления.

75. Составьте алгоритм, выводящий в результате

а) все простые числа в диапазоне от 1 до 100.

б) все простые числа в диапазоне от A до B.

в) все делители натурального числа N.

г) сумму цифр числа N.

76. Составьте алгоритм для вычисления суммы 100 слагаемых:

$$y=\sin(1\pi/2) - \sin(3\pi/4)+ \sin(5\pi/8) -...$$

77. Составьте алгоритм нахождения наибольшего общего делителя (НОД) двух целых положительных чисел.

Раздел 2. Программирование на JavaScript

Примерное практическое задание. Доработайте веб-игру "Арканоид", согласно своему варианту.

Варианты задания:

1) Добавьте режим автоигры в программу. Запуск того или иного режима должен начинаться с нажатия соответствующих кнопок на клавиатуре.

2) Добавьте в программу возможность масштабирования игрового поля и игровых элементов (это могут быть кнопки управления масштабированием).

3) Добавьте в программу возможность изменения количества кирпичиков, как по горизонтали, так и по вертикали (это могут быть кнопки управления количеством M и N).

4) Добавьте в программу цветовую палитру. Раскрасьте кирпичики разными цветами, с формированием узора.

5) Добавьте в программу бронированные кирпичики (т. е. чтобы его разбить нужно по нему ударить мячом несколько раз, например три).

6) Добавьте в программу механизм получения бонусов (т. е. удар мяча по кирпичику приводит к падению предмета, поймав который изменяется характеристика платформы-ракетки, например ее скорость движения).

7) Добавьте в программу больше уровней. От уровня к уровню изменяется скорость, количество кирпичиков, или характеристика платформы-ракетки.

8) Добавьте в программу задержку старта начала игры. Старт игры нужно начинать, нажав соответствующую кнопку. После каждого промаха, старт начинается по нажатию на кнопку.

9) Добавьте в программу случайную логически связную расстановку кирпичиков.

10) Исправьте "баг" отскока мяча от платформы-ракетки (мяч отскакивает от нижней поверхности платформы, а должен от верхней).

11) Добавьте свойство "крученого" мяча (мяч при ударе подкручивается, а значит меняется его поведение при отражении от поверхности).

12) Раскрасьте мяч и платформу-ракетку. Добавьте возможность изменения скорости движения мяча путем установки свойств.

Исходный код программы

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <style>* { padding: 0; margin: 0; } canvas { background: #eee; display: block; margin: 0 auto; }</style>
</head>
<body>

<canvas id="myCanvas" width="480" height="320"></canvas>

<script>
  var canvas = document.getElementById("myCanvas");
  var ctx = canvas.getContext("2d");
  var ballRadius = 10;
  var x = canvas.width/2;
  var y = canvas.height-30;
  var dx = 2;
  var dy = -2;
```

```
var paddleHeight = 10;
var paddleWidth = 75;
var paddleX = (canvas.width-paddleWidth)/2;
var rightPressed = false;
var leftPressed = false;
var brickRowCount = 5;
var brickColumnCount = 3;
var brickWidth = 75;
var brickHeight = 20;
var brickPadding = 10;
var brickOffsetTop = 30;
var brickOffsetLeft = 30;
var score = 0;
var lives = 3;

var bricks = [];
for(var c=0; c<brickColumnCount; c++) {
    bricks[c] = [];
    for(var r=0; r<brickRowCount; r++) {
        bricks[c][r] = { x: 0, y: 0, status: 1 };
    }
}

document.addEventListener("keydown", keyDownHandler, false);
document.addEventListener("keyup", keyUpHandler, false);
document.addEventListener("mousemove", mouseMoveHandler, false);

function keyDownHandler(e) {
    if(e.code  == "ArrowRight") {
        rightPressed = true;
    }
    else if(e.code == 'ArrowLeft') {
        leftPressed = true;
    }
}
function keyUpHandler(e) {
    if(e.code == 'ArrowRight') {
        rightPressed = false;
    }
    else if(e.code == 'ArrowLeft') {
        leftPressed = false;
    }
}
function mouseMoveHandler(e) {
    var relativeX = e.clientX - canvas.offsetLeft;
    if(relativeX > 0 && relativeX < canvas.width) {
        paddleX = relativeX - paddleWidth/2;
    }
}
function collisionDetection() {
    for(var c=0; c<brickColumnCount; c++) {
        for(var r=0; r<brickRowCount; r++) {
```

```
            var b = bricks[c][r];
            if(b.status == 1) {
                if(x > b.x && x < b.x+brickWidth && y > b.y && y < b.y+brickHeight) {
                    dy = -dy;
                    b.status = 0;
                    score++;
                    if(score == brickRowCount*brickColumnCount) {
                        alert("YOU WIN, CONGRATS!");
                        document.location.reload();
                    }
                }
            }
        }
    }
}

function drawBall() {
    ctx.beginPath();
    ctx.arc(x, y, ballRadius, 0, Math.PI*2);
    ctx.fillStyle = "#0095DD";
    ctx.fill();
    ctx.closePath();
}
function drawPaddle() {
    ctx.beginPath();
    ctx.rect(paddleX, canvas.height-paddleHeight, paddleWidth, paddleHeight);
    ctx.fillStyle = "#0095DD";
    ctx.fill();
    ctx.closePath();
}
function drawBricks() {
    for(var c=0; c<brickColumnCount; c++) {
        for(var r=0; r<brickRowCount; r++) {
            if(bricks[c][r].status == 1) {
                var brickX = (r*(brickWidth+brickPadding))+brickOffsetLeft;
                var brickY = (c*(brickHeight+brickPadding))+brickOffsetTop;
                bricks[c][r].x = brickX;
                bricks[c][r].y = brickY;
                ctx.beginPath();
                ctx.rect(brickX, brickY, brickWidth, brickHeight);
                ctx.fillStyle = "#0095DD";
                ctx.fill();
                ctx.closePath();
            }
        }
    }
}
function drawScore() {
    ctx.font = "16px Arial";
    ctx.fillStyle = "#0095DD";
    ctx.fillText("Score: "+score, 8, 20);
}
```

```
    function drawLives() {
        ctx.font = "16px Arial";
        ctx.fillStyle = "#0095DD";
        ctx.fillText("Lives: "+lives, canvas.width-65, 20);
    }
    function draw() {
        ctx.clearRect(0, 0, canvas.width, canvas.height);
        drawBricks();
        drawBall();
        drawPaddle();
        drawScore();
        drawLives();
        collisionDetection();
        if(x + dx > canvas.width-ballRadius || x + dx < ballRadius) {
            dx = -dx;
        }
        if(y + dy < ballRadius) {
            dy = -dy;
        }
        else if(y + dy > canvas.height-ballRadius) {
            if(x > paddleX && x < paddleX + paddleWidth) {
                dy = -dy;
            }
            else {
                lives--;
                if(!lives) {
                    alert("GAME OVER");
                    document.location.reload();
                }
                else {
                    x = canvas.width/2;
                    y = canvas.height-30;
                    dx = 2;
                    dy = -2;
                    paddleX = (canvas.width-paddleWidth)/2;
                }
            }
        }
        if(rightPressed && paddleX < canvas.width-paddleWidth) {
            paddleX += 7;
        }
        else if(leftPressed && paddleX > 0) {
            paddleX -= 7;
        }
        x += dx;
        y += dy;
        requestAnimationFrame(draw);
    }
    draw();
</script>
</body>
</html>
```

Приложение 2.3. Примерные вопросы тестов.

Раздел 1. Алгоритмизация и программирование

1) Алгоритмом можно считать:
- описание решения квадратного уравнения
- расписание уроков в школе
- технический паспорт автомобиля
- список класса в журнале

2) Как называется свойство алгоритма, означающее, что данный алгоритм применим к решению целого класса задач?
- понятность
- определённость
- результативность
- массовость

3) Как называется свойство алгоритма, означающее, что он всегда приводит к результату через конечное, возможно, очень большое, число шагов?
- понятность
- определённость
- результативность
- массовость

4) Как называется свойство алгоритма, означающее, что он задан с помощью таких предписаний, которые исполнитель может воспринимать и по которым может выполнять требуемые действия?
- понятность
- определённость
- результативность
- массовость

5) Как называется свойство алгоритма, означающее, что путь решения задачи разделён на отдельные шаги?
- дискретность
- определённость
- результативность
- массовость

6) Величины, значения которых меняются в процессе исполнения алгоритма, называются:
- константами
- переменными
- табличными
- постоянными

7) Величиной целого типа является
- рост человека
- количество мест в зрительном зале
- площадь государства

8) Дан фрагмент линейного алгоритма,

a=8
b=6+3*a
a=b/3*a

Чему равно значение переменной **a** после его исполнения?

9) Исполните следующий фрагмент линейного алгоритм для a = x и b = y.

a=a+b
b=b—a
a=a+b
b=—b
Какие значения присвоены переменным a и b ?

- y, x
- x+y, x-y
- x, y
- -y, x

10) Среди четырёх монет есть одна фальшивая. Неизвестно, легче она или тяжелее настоящей. Какое минимальное количество взвешиваний необходимо сделать на весах с двумя чашками без гирь, чтобы определить фальшивую монету?

- 2
- 3
- 4
- 5

11) Результатом этапа алгоритмизации может быть …

- программа
- блок-схема
- математическая модель

12) Процесс проверки работоспособности программы и исправления обнаруженных при этом ошибок происходит на этапе …

- отладки и тестирования
- программирования
- алгоритмизации
- выполнения расчетов

Раздел 2. Программирование на JavaScript.

№1 Укажите допустимый синтаксис комментария.
1. /* Комментарий */
2. # Комментарий
3. /* Комментарий
4. // Комментарий
№2 Какие типы данных не существуют в Java?
1. int
2. float
3. string
4. unknown
5. Double
№3 Можно ли создать программу (приложение) на Java, не используя среду разработки (IDE)?
1. Да
2. Нет, так как необходимо скомпилировать исходный код в байт-код
№4 Какое расширение имеют файлы с исходным кодом Java?
1. .javac
2. .java
3. .class
4. .classpath
№5 Может ли файл содержать более одного класса Java?
1. Да, но только если один внешний класс имеет модификатор доступа public
2. Да, если все внешние классы будут иметь модификатор доступа private
3. Нет
№6 Какое имя переменной является синтаксически недопустимым?
1. 53someVariable

2. _someVariable
3. some-variable
4. somevariable
5. someVariable
6. Somevariable53
7. _53someVariable

№7 От какого класса наследуются все создаваемые классы в Java?
1. Классы наследуется от типа указанного после ключевого слова extends, если тип не указан, значит класс не является наследником
2. Object
3. Class

№8 Импорт какого пакета в Java происходит автоматически?
1. Все пакеты нужно явно указывать
2. java.util
3. java.lang
4. java.text

№9 Укажите какой тип данных, из представленных, занимает наибольшее место в памяти.
1. int
2. byte
3. double
4. float
5. никакой, так как все примитивные типы занимают одинаковый объём в памяти

№10 Какой арифметический оператор в Java не существует?
1. --
2. %
3. **
4. ++

№11 Какого оператора сравнения в Java не существует
1. !=
2. ==
3. <>
4. ===
5. >=

№12 Какой тип преобразования не произойдёт автоматически?
1. byte в short
2. int в long
3. char в short
4. long в float

№13 Какой тип данных не поддерживает оператор switch?
1. int
2. String
3. char
4. long

№14 Какое выражение создания переменной и массива является синтаксически некорректным?
1. int array[] = new int[0];
2. int[] array = new int[1];
3. int array = new int[1];
4. int[] array = new int[] {1, 2};
5. int[] array = {1, 2};

№15 Выбирите верное утверждение. Добавление ключевого слова final к полю класса означает, что
1. поле класса может быть инициализировано только один раз
2. поле должно инициализироваться при объявлении, в конструкторе или в инициализаторе экземпляра
3. поле по умолчанию становится закрытым (аналогично использованию модификатора доступа private)

№16 Что не влияет на перегрузку метода?
1. Количество параметров
2. Тип возвращаемого значения
3. Модификаторы доступа

4. Типы параметров
№17 Значение ключевого слова void?
1. Метод не может переопределяться в классе наследнике
2. Метод ничего не возвращает
3. Метод не может перегружаться
№18 Назначение ключевого слова extends?
1. Используется, чтобы указать класс, интерфейс, от которого происходит наследование
2. Добавляется к методу класса, указывая, что метод может наследоваться
3. Используется в обобщениях (generics) для наложение ограничений типа
№19 Какое ключевое слово не относится к модификаторам доступа?
1. public
2. package
3. private
4. protected
№20 Как вызвать конструктор класса? class SomeClass { }
1. construct()
2. constructor()
3. SomeClass()
4. someClass()
№21 Все методы интерфейса по умолчанию являются
1. public и abstract
2. protected и abstract
3. public и super
№22 Какие типы не относятся к примитивным?
1. int
2. Byte
3. char
4. short
5. Object
6. String
№23 Выбирите только целочисленные типы.
1. int
2. String
3. double
4. char
5. short
6. long
№24 Можно ли создать свой примитивный тип данных?
1. Нет
2. Да
№25 Укажите отличия переменных примитивных и ссылочных типов.
1. Переменные примитивных типов не могут являться полями в классах, в отличии от ссылочных.
2. Переменные примитивных типов хранят значение, а переменные ссылочных типов ссылку на объект.
3. Значение переменных примитивных типов нельзя вернуть из метода, а ссылочных можно.
№26 Какое значение по умолчанию будет присвоено для поля counter? public class Some { private Short counter; }
1. 0
2. undefined
3. null
4. Произойдет ошибка компиляции, так как private поля нужно обязательно инициализировать.
№27 Что будет выведено в результате выполнения кода? public class Main { public static void main(String[] args) { Some some1 = new Some(); Some some2 = some1; System.out.println(some1 == some1); } }
1. false
2. true
№28 Какой результат будет выведен в резульатет выполнения кода? public class Main { public static void main(String[] args) { int number = 1; boolean isSome = (boolean) number; System.out.println(isSome); } }
1. 1

2. true
3. Код не скомпилируется, так как существует ошибка привидения типа int к boolean.
4. null
№29 Что будет выведно, в результате выполнения кода? public class Main { public static void main(String[] args) { int a = 10; int b = 2; if ((a++ > 0) | (b++ > 0)) { b += 4; } System.out.println(b); } }
1. 7
2. 6
3. 2
№30 Какой из типов относится к беззнаковым?
1. int
2. char
3. long
4. double
№31 Какая запись вызовет ошибку во время компиляции?
1. int a = 12;
2. int a = 014;
3. int a = 1_2;
4. int a = 0b1100;
5. Все записи корректны, ошибки не будет.
№32 Какое значение будет выведено, при выполнении кода? public class Main { public static void main(String[] args) { int a = 10; some(a++); } public static void some(int a) { System.out.println(a); } }
1. 11
2. 12
3. 10
№33 Какое значение будет выведено, при выполнении кода? public class Main { public static void main(String[] args) { byte a = 127; System.out.println(++a); } }
1. 128
2. -128
3. 127
4. Произойдёт ошибка, так как тип byte не может хранить число более 127.
№34 Какое значение будет выведено, при выполнении кода? public class Main { public static void main(String[] args) { int a = 1; int b = 2; System.out.println(a | b); } }
1. 2
2. false
3. 3
4. true
№35 Возможны ли арифметические операции с типом char?
1. Да.
2. Нет.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ АРХИТЕКТУРНО-СТРОИТЕЛЬНЫЙ УНИВЕРСИТЕТ»

Институт непрерывного образования

СОГЛАСОВАНО
Директор ИНО

_____ Н.Р.Шадейко
(подпись)

"____"_____20___ г.

УТВЕРЖДАЮ
Проректор по учебной работе

_____ Д.Н. Песцов
(подпись)

"____"_____20___ г.

ПРОГРАММА МОДУЛЯ

PHP и базы данных
(наименование дисциплины)

ДОПОЛНИТЕЛЬНАЯ ПРОФЕССИОНАЛЬНАЯ ПРОГРАММА

«Web-программирование»
(наименование программы)

Томск 2025

Составитель *профессор кафедры ПМ, Г.А. Онопенко*
(Должность, Ф.И.О., подпись)

Рецензент *профессор кафедры ПМ, Б.М. Шумилов*
(Должность, Ф.И.О., подпись)

Введена в действие с «___» _____ 20___ г.

СОДЕРЖАНИЕ

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ 4
1.1 Учебный план 4
1.2 Планируемые результаты обучения 4
2. СОДЕРЖАНИЕ ПРОГРАММЫ 5
2.1 Учебно-тематический план 5
3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ 6
3.1 Материально-технические условия реализации программы 6
3.2 Учебно-методическое и информационное обеспечение программы 6
3.2.1 Литература 6
3.2.2 Методические разработки 7
3.3 Процедура зачета 7
3.3 Кадровое обеспечение 7
ПРИЛОЖЕНИЕ 1 8
ПРИЛОЖЕНИЕ 2 9

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ

1.1 Учебный план

Наименование дисциплины	Общая трудоем-кость, час.	Всего, ауд. час.	Аудиторные занятия, час.			СРС, час.	Текущий контроль*, (шт.)			Промежуточная аттестация	
			лекции	лабораторные работы	прак. занятия, семинар		РК, РГР, реф.	КР	КП	Зачет	Экзамен
1	2	3	4	5	6	7	8	9	10	11	12
Модуль 3. PHP и базы данных											
Базы данных	28	20	6		14	8					
Динамические веб-технологии. Язык PHP	52	36	8		28	16					
Итого	**80**	**56**	**14**		**42**	**24**				**Зачёт**	

1.2 Планируемые результаты обучения

Слушатель программы переподготовки будет обладать следующими профессиональными компетенциями:

Матрица компетенций дисциплины

Профессиональные компетенции	ID-28	ID-35
	Применяет языки программирования для решения профессиональных задач	Разрабатывает различные веб-архитектуры

Слушатель программы переподготовки будет способен осуществлять следующие трудовые действия:

- Выявление первоначальных требований заказчика к ИР (ID-28);
- Определение возможности достижения соответствия ИР первоначальным требованиям заказчика (ID-28);
- Проектирование структуры разделов ИР (ID-28);
- Разработка интерфейса пользователя для ИР с использованием стандартов в области web-разработки (ID-35).
- Настройка контроля целостности файлов ИР и прикладного программного обеспечения (ID-35);
- Мониторинг атак на ИР и выполнение соответствующих действий (ID-35).

Слушатель программы переподготовки будет обладать следующими необходимыми умениями:

- умеет применять стандартные алгоритмы и использовать типовые решения задач программирования (ID-28, У2);
- умеет использовать стандартные средства программирования для моделирования объектов и процессов (ID-28, У3);
- умеет применять вспомогательные инструментальные программные средства проектирования программного обеспечения, структур данных, программных интерфейсов (ID-35, У3).

Слушатель программы переподготовки будет обладать следующими необходимыми знаниями:

- знает базовые технологии разработки алгоритмов и программ (ID-28, З2);
- знает основные синтаксические конструкции языков программирования высокого уровня, необходимых для разработки web-сайта (ID-28, З3);
- знает стандартные программные средства проектирования программного обеспечения, структур данных, программных интерфейсов (ID-35, З3).

2. СОДЕРЖАНИЕ ПРОГРАММЫ

2.1 Учебно-тематический план

№ п/п	Наименование разделов/тем	Общая трудоемкость, час.	Всего ауд. час.	По учебному плану с использованием дистанционных образовательных технологий, час.						СРС, час.
				Аудиторные занятия, час.			Дистанционные занятия, час.			
				лекции	лабораторные работы	практические занятия	лекции	лабораторные работы	практические и семинарские занятия	
1	2	3	4	5	6	7	8	9	10	11
1	**Раздел 7. Базы данных**									
2	Краткое введение в базы данных (БД). Системы управления базами данных (СУБД). Реляционная модель данных. Язык SQL.	*10*	*8*	*4*		*4*				*2*
3	Настройка базы данных	*3*	*2*			*2*				*1*
4	Работа с таблицами и запросами.	*7*	*5*	*1*		*4*				*2*
5	IDE для работы с БД. Расширение PDO для интерпретатора PHP для работы с БД.	*8*	*5*	*1*		*4*				*3*
6	**Итого**	**28**	**20**	**6**	**-**	**14**	**-**	**-**	**-**	**8**
7	**Раздел 8. Динамические веб-технологии. Язык PHP**									
8	Основы языка PHP. Типы данных, основные операторы и функции.	*7*	*5*	*1*		*4*				*2*
9	Установка и настройка веб-сервера. Подключение модуля PHP.	*11*	*7*	*1*		*6*				*4*
10	Включение PHP кода в HTML.	*10*	*8*	*2*		*6*				*2*
11	Переменные и типы данных. Массивы. Функции. Классы, объекты и объявление методов. Наследование.	*12*	*8*	*2*		*6*				*4*
12	Объектно-ориентированный интерфейс БД. Выполнение запросов к СУБД с помощью функций PHP. Базы данных и веб-формы.	*12*	*8*	*2*		*6*				*4*
13	**Итого**	**52**	**36**	**8**	**-**	**28**	**-**	**-**	**-**	**16**
14	**Всего**	**80**	**56**	**14**		**42**				**24**

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

3.1 Материально-технические условия реализации программы

Наименование специализированных аудиторий, кабинетов, лабораторий	Вид занятий	Наименование оборудования, программного обеспечения
Компьютерный класс 304/1	лекции, практические занятия	Компьютеры, проектор, экран, доска, Интернет, Интернет-браузеры.

3.2 Учебно-методическое и информационное обеспечение программы

3.2.1 Литература

Основная литература

1. Гуриков, С.Р. Интернет-технологии: Учебное пособие / С.Р. Гуриков. - М.: Форум: НИЦ ИНФРА-М, 2015. - 184 с. – Режим доступа: http://znanium.com/go.php?id=488074
2. Никулова Г.А. WEB-программирование. Серверные технологии: PHP. Часть 1: Учебно-методическое пособие / Никулова Г. А., Субботин В. Р. - Липецк: Липецкий ГПУ, 2017. – 59 с. URL: https://e.lanbook.com/book/111934.
3. Кожевникова П.В. PHP и MySQL: учебное пособие / Кожевникова П.В. - Ухта: УГТУ, 2020. - 51 с. URL: https://e.lanbook.com/book/209591
4. Янцев В.В. JavaScript и PHP. Content management system: учебное пособие для вузов / Янцев В.В. - Санкт-Петербург: Лань, 2022. - 192 с. URL: https://e.lanbook.com/book/266657.

Дополнительная литература

1. Ружицкая Е.А. Основы web-технологий: работа с данными формы в PHP: практическое пособие / Ружицкая Е.А., Кечко Е.П. - Гомель: ГГУ имени Ф. Скорины, 2022. - 47 с. URL: https://e.lanbook.com/book/320990

Ресурсы информационно-коммуникационной сети «Интернет»

1. Электронная научно-техническая библиотека ТГАСУ [Электронный ресурс]. – Режим доступа http://tsuab.ru/ru/struktura-tgasu/nt-library/, свободный.
2. Курсы дистанционного обучения ТГАСУ [Электронный ресурс] – Режим доступа: https://digital.tsuab.ru/, сервер ТГАСУ- требуется регистрация.
3. Электронные книги и учебники по веб-дизайну и веб-программированию для начинающих [электронный ресурс] / Режим доступа: https://html5beginner.github.io/books/index.html
5. Сайт по эффективному программированию на PHP [Электронный ресурс] / Режим доступа: http://phpfaq.ru.
6. Сайт по администрированию phpMyAdmin [Электронный ресурс]. - URL: http://php-myadmin.ru.
7. Статья по установке Apache, PHP, MySQL под Windows [Электронный ресурс]. - URL: http://www.cyberforum.ru/apache/thread83339.html.
8. Сайт по поддержке на PHP, SQL и MySQL [Электронный ресурс] / Режим доступа: https://php.zone/test-mysql
9. Базы данных Oracle [электронный ресурс] / Режим доступа: https://oracleplsql.ru/
10. Справочные и учебные материалы для создания сайтов [электронный ресурс] / Режим доступа: https://basicweb.ru/
11. Академия HTML. Сервис интерактивного онлайн-обучения [электронный ресурс] / Режим доступа: http://www.htmlacademy.ru/

3.2.2 Методические разработки

1. Технология создания web-страниц: методические указания к самостоятельной работе и задания к контрольной работе / сост. Э.А. Эшаров, Г.А. Онопенко, Е.Ю. Корниенко. – Томск: Изд-во Том. гос. архит.- строит. ун-та, 2015. – 44 с.

3.3 Процедура зачета

Формой промежуточной аттестации является зачет, который проводится в форме теста. К зачету допускаются студенты, прослушавшие лекции, выполнившие практические работы.

Зачтено	Выставляется студенту в случае, если уровень выполнения, предусмотренной зачетом работы, отвечает большинству требований, теоретическое содержание дисциплины освоено полностью или частично, но пробелы не носят существенного характера, необходимые навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено. По результатам итогового теста студент набрал более, чем 70% правильных ответов.
Не зачтено	Выставляется студенту в случае, если уровень выполнения, предусмотренной зачетом работы слабый, уровень выполнения не отвечает большинству требований, теоретическое содержание курса освоено частично, некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения задания не выполнены, либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному. По результатам итогового теста студент набрал менее 70% правильных ответов.

В случае неудовлетворительной оценки студент имеет право пересдать зачет в установленном порядке.

3.4 Кадровое обеспечение

Авторы и преподаватели:

№ п/п	ФИО, должность	Модули (темы, лекции)	Часов, всего
1.	Титов А.В., доцент кафедры прикладной математики	Модуль 3. **Раздел 7.** Базы данных	28
2.	Воронкова И.А., старший преподаватель кафедры прикладной математики	Модуль 3. **Раздел 8.** Динамические веб-технологии. Язык PHP	52
3.	Кульков Д.Н., руководитель отдела ИТ ООО «ТН Диджитал»	Модуль 3. **Раздел 8.** Динамические веб-технологии. Язык PHP	52

ПРИЛОЖЕНИЕ 1

АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ

PHP и базы данных

(название модуля)

1. Цели и задачи изучения модуля:

Формирование знаний о базах данных (БД) и системах управления базами данных (СУБД); об инструментарии языка запросов SQL и его применения на практике при разработке и организации работы web-сайта; применение на практике навыков теоретической и практической подготовки в области разработки веб-приложений с использованием современного языка программирования PHP, СУБД MySQL, а также современных сред разработки.

ID-28. Применяет языки программирования для решения профессиональных задач

ID-35. Разрабатывает различные веб-архитектуры

3. Планируемые результаты обучения по модулю.

- умеет применять стандартные алгоритмы и использовать типовые решения задач программирования (ID-28, У2);
- умеет использовать стандартные средства программирования для моделирования объектов и процессов (ID-28, У3);
- умеет применять вспомогательные инструментальные программные средства проектирования программного обеспечения, структур данных, программных интерфейсов (ID-35, У3);
- знает базовые технологии разработки алгоритмов и программ (ID-28, З2);
- знает основные синтаксические конструкции языков программирования высокого уровня, необходимых для разработки web-сайта (ID-28, З3);
- знает стандартные программные средства проектирования программного обеспечения, структур данных, программных интерфейсов (ID-35, З3).

4. Тематическое содержание модуля

Краткое введение в базы данных (БД). Системы управления базами данных (СУБД). Реляционная модель данных. Язык SQL. Настройка базы данных. Работа с таблицами и запросами. IDE для работы с БД. Расширение PDO для интерпретатора PHP для работы с БД.

Основы языка PHP. Типы данных, основные операторы и функции. Установка и настройка веб-сервера. Подключение модуля PHP. Включение PHP кода в HTML. Переменные и типы данных. Массивы. Функции. Классы, объекты и объявление методов. Наследование.

Объектно-ориентированный интерфейс БД. Выполнение запросов к СУБД с помощью функций PHP. Базы данных и веб-формы.

ПРИЛОЖЕНИЕ 2

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО МОДУЛЮ

PHP И БАЗЫ ДАННЫХ

Составитель профессор, Г.А. Онопенко
(должность, ФИО, подпись)

Рецензент профессор, Б.М. Шумилов
(должность, ФИО, подпись)

Приложение 2.1. Вопросы теста.

Раздел 1. Базы данных

1. Что такое SQL?
   - Язык программирования
   - Лучший способ приготовить кофе
   - **Язык запросов структурированных данных**
   - Протокол передачи данных
2. Какой запрос позволяет выбрать все записи из таблицы "students"?
   - SELECT * INTO students;
   - **SELECT * FROM students;**
   - INSERT INTO students SELECT *;
   - SELECT pizzas;
3. Какой оператор используется для фильтрации записей?
   - **WHERE**
   - SELECT
   - JOIN
   - FILTER COFFEE
4. Какая команда объединяет строки из двух таблиц по общему полю?
   - CONCAT
   - UNION
   - DANCE TOGETHER
   - **JOIN**
5. Что делает оператор 'DISTINCT' в запросе SELECT?
   - Сортирует строки
   - **Удаляет дубликаты**
   - Танцует вальс
   - Объединяет таблицы
6. Какой запрос удалит таблицу с названием "employees"?
   - DELETE TABLE employees;
   - **DROP TABLE employees;**
   - REMOVE TABLE employees;
   - THROW TABLE employees OUT OF WINDOW;
7. Какой тип JOIN возвращает только совпадающие строки из обеих таблиц?
   - **INNER JOIN**
   - FULL JOIN
   - LEFT JOIN THE PARTY
   - LEFT JOIN
8. Какой SQL-запрос обновит значение "age" до 30 в таблице "persons", где "name" равно 'Dima'?
   - MODIFY persons SET age=30 WHERE name='Dima';
   - UPDATE persons SET age=30 AND name='Dima';
   - CHANGE persons SET age=30 WHERE name='Dima';
   - **UPDATE persons SET age=30 WHERE name='Dima';**
9. Какая команда используется для создания индекса?
   - INDEX
   - **CREATE INDEX**
   - MAKE A SHOPPING LIST
   - ADD INDEX
10. Какой тип данных в SQL используется для хранения даты и времени?
    - DATE

- TIME
- DATETIME
- **TIMESTAMP**

11. Что делает следующий запрос: INSERT INTO customers (id, name) VALUES (1, 'Ann');?
    - **Добавляет новую строку с id 1 и name 'Ann' в таблицу customers**
    - Обновляет значения в таблице customers
    - Удаляет строку с id 1 и name 'Ann' из таблицы customers
    - Ничего из вышеперечисленного
12. Какой оператор используется для сортировки результатов запроса?
    - ORDER A PIZZA
    - ORDER
    - SORT BY
    - **ORDER BY**
13. Что такое первичный ключ в SQL?
    - Основной ключ, который может быть дублирован
    - Ключ для шифрования данных
    - Ключ для моего сердца
    - **Уникальный ключ для каждой строки в таблице**
14. Какую роль выполняет команда HAVING в SQL?
    - Определяет порядок сортировки
    - **Фильтрует строки после выполнения группировки**
    - Устанавливает соединение с базой данных
    - Создает индекс в таблице
15. Какой запрос удаляет все строки из таблицы "orders", но не удаляет саму таблицу?
    - DROP TABLE orders;
    - DELETE orders;
    - DELETE * FROM orders;
    - **DELETE FROM orders;**

Раздел 2. Динамические веб-технологии. Язык PHP

1. Как называется программа, которая выполняет код.
2. С какого символа начинаются многострочные комментарии.
3. Что можно оставлять внутри кода.
4. Напишите код для вывода на экран слова «goodbye».
5. Какой тег ставится в самом начале кода на PHP.
6. Какой базовый стандарт написания кода (Basic Coding Standard).
7. Какой расширенный стандарт стиля кодирования (Extended Coding Style).
8. С какого символа начинаются однострочные комментарии.
9. Какая функция возвращает длину строки, то есть количество символов в ней.
10. Какая функция подсчитывает количество слов в строке.
11. Какая функция возвращает строку, перевернутую задом наперед.
12. Какая функция заменяет некоторые символы на другие символы в строке.
13. Какая функция ищет подстроку в строке и в случае успеха возвращает позицию начала этой подстроки.
14. Какая функция удаляет из строки начальные и конечные пробелы, а также управляющие символы '\n', '\r', '\t'.
15. Преимущества операции интерполяции.
16. Какое значение будет выведено на экран?

```
<?php
$applesPerBox = 60;
```

```
$boxesPerContainer = 50;
$container = 4;
$applesCount = $applesPerBox * $boxesPerContainer * $container;
print_r($applesCount);
```

17. Сколько переменных можно подставлять в строку для интерполяции.
18. С какими кавычками работает интерполяция.
19. Напишите код, с помощью которого из заданной строки извлекаются первый и последний символы и выводятся на экран.
20. Выведите на экран приветствие с помощью функции show(), выводящей на экран тестовую переменную $text="Hello, Vasya!"
21. Определите функцию sum, возвращающую в качестве результата сумму двух переменных a и b. Закрывающий тег ?> в конце программы ставить не нужно.
22. В PHP есть функция ucfirst(). Изучите ее сигнатуру (https://php.net/manual/ru/function.ucfirst.php) и напишите программу, которая использует функцию ucfirst() с переменной $sent и выводит результат выполнения функции на экран. Значение переменной $sent="Hope is a good breakfast, but a bad supper".
23. Определите функцию result, которая находит максимальный элемент из множества шести чисел {-10, 20, 30, 44,-88, 99}, используя математическую функцию max. Запишите найденный элемент в переменную $resul, выведите результат на экран.
24. Необходимо реализовать функцию getLastChar($string), возвращающую последний символ в строке (строка - параметр функции). Напоминание - первый символ в строке имеет индекс 0, последний символ в строке имеет индекс -1. Для возврата значения функции используйте return. В качестве строки можно использовать поговорки:
    a) A bad workman blames his tools.
    b) A bargain is a bargain.
    c) A word spoken is past recalling.
25. Напишите код, определяющий, что значение переменной $a больше, меньше или равно нулю, используя if, else и elseif.
26. Напишите код, который выбирает цвет фона страницы в зависимости от времени суток. С 20 часов вечера до 6 утра - ночь (темный цвет), с 6 утра до 12 - утро (голубой цвет), остальное время - день (желтый цвет). Чтобы определить время суток используйте функцию date("H") (https://www.php.net/manual/ru/datetime.format.php). Возможно, в составном условии понадобится логическая связка ИЛИ (дизъюнкция, ||). Для вывода на эран цвета фона страницы используйте оператор echo "<body style='background-color: #000000;>".
27. Реализуйте функцию reverseString($str), осуществляющую переворот строки с помощью цикла for.
28. Напишите код, который выводит на экран сумму чисел от 1 до 250. Цикл while.
29. Создайте цикл, который выводит числа то 0 до 100 в HTML-элементах div; окраска HTML-элементов должна чередоваться («зебра»).
30. Задан массив с информацией об общем количестве аэропортов в разных странах (Информация) . Названия разных стран являются ключами, а количество аэропортов значениями. Отсортируйте названия стран по алфавиту с помощью функций ksort() и krsort(). Функция ksort() сортирует ключи по возрастанию, а krsort() сортирует ключи по убыванию. Выведите на экран результаты обеих сортировок.
31. Определите класс Person, который содержит два свойства ($name и $age) и один метод hello(), который вызывает приветствие: Hello! Jerry
32. Определите класс Person, который содержит два свойства ($name и $age) и один метод message(), который выводит: Name: Jerry; Age: 18
33. С помощью какого запроса можно получить список имен пользователей без повторений?

a) SELECT name FROM table WHERE name NOT IN (name);
b) SELECT name FROM table GROUP BY name;
c) SELECT u1.name FROM table u1 INNER JOIN table u2 ON u1.name <> u2.name;
d) SELECT name FROM table ORDER BY name;

34. С помощью какого запроса можно получить имена пользователей, встречающихся более одного раза?
a) SELECT name FROM table WHERE MIN(name)>1 GROUP BY name DESC;
b) SELECT name FROM table GROUP BY name HAVING COUNT(name)>1;
c) SELECT name FROM table WHERE COUNT(name)>=1 GROUP BY name DESC;

35. С помощью какого ключевого слова можно задать условия фильтрации после выполнения группировки?
a) SELECT
b) HAVING
c) WHERE

36. С помощью какого ключевого слова можно сделать автоматическую генерацию первичного ключа?
a) PRIMARY KEY
b) AUTO_INCREMENT
c) UNIQUE
d) INDEX

37. Какой является СУБД MySQL?
a) Графовой
b) Векторной
c) Реляционной
d) Документо-ориентированной

38. С помощью какого запроса можно получить список всех имен пользователей без повторов?
a) SELECT DISTINCT name FROM table;
b) SELECT name FROM table WHERE name NOT IN (name);
c) SELECT name FROM table ORDER BY name;

39. С помощью какого ключевого слова можно соединить строки из нескольких таблиц, основываясь на их связи?
a) CONCAT
b) IMPLODE
c) JOIN

40. С помощью какого ключевого слова задаются условия запроса?
a) WHEN
b) WHERE
c) ORDER
d) INSERT

41. С помощью какой конструкции можно изменять структуру таблиц?
a) EDIT TABLE
b) DROP TABLE
c) ALTER TABLE

42. Каким запросом можно выбрать базу my_db для работы?
a) update my_db
b) use my_db
c) select my_db

43. С помощью каких ключевых слов задаётся порядок сортировки?
a) ASC/DESC
b) UP/DOWN
c) FROM/TO

44. Создайте пользовательскую форму с сообщением «Введите значение длины и ширины прямоугольника». Создайте два текстовых поля, одно для ввода длины и одно для ввода ширины. Напишите функцию для обработки значений, введенных пользователем, верните пользователю результат: «Прямоугольник длиной $l и шириной $w имеет площадь $area», где $l и $w - аргументы, а $area - результат.
45. Создайте HTML-форму с элементом <select>, который позволит пользователю выбрать один из предложенных результатов. Элемент <select> должен иметь атрибут name, который будет использоваться для идентификации форм данных в PHP. В файле handle-it.php, указанном в форме атрибута action, обработайте отправленные данные. Используйте глобальный массив $_POST для доступа к отправленным данным.
46. Создайте HTML-форму (form.html), которая будет запрашивать у пользователя имя, фамилию, тип домашнего питомца. В PHP-скрипте (action_form.php) введенная пользователем информация будет передаваться в таблицу базы данных (survey_responses), а затем обрабатываться SQL-запросом, подсчитывающим одинаковые результаты ответов.

Приложение 2.3. Практические задания.

Раздел 1. Базы данных

Практическое занятие 1. Знакомство с консольными утилитами MySQL и просмотр объектов СУБД

Цель: ознакомить студентов с консольными утилитами, входящими в пакет MySQL и дать навыки, позволяющие узнавать какие объекты есть в базе данных и оперировать с ними.

Содержание работы: Подключение к удаленной СУБД. Просмотр списка доступных баз данных - SHOW DATABASES. Выбор текущей БД - USE [имя БД]. Просмотр списка таблиц из *текущей БД* - SHOW TABLES. Просмотр списка полей из указанных преподавателем таблиц - SHOW FIELDS FROM [имя таблицы], DESC [имя таблицы].

Практическое занятие 2. Изучение структур баз данных.

Цель: ознакомить студентов с основными типами данных, допустимыми в MySQL, дать навыки, позволяющие определять структуры таблиц в базе данных и связи между таблицами.

Содержание работы: Подключение к удаленной СУБД. Просмотр списка доступных баз данных - SHOW DATABASES. Выбор в качестве текущей БД TV_DB - USE TV_DB. Знакомство со структурой БД TV_DB**.** Просмотр списка таблиц из *текущей БД* - SHOW TABLES. Просмотр списка полей из указанных таблиц - SHOW FIELDS FROM [имя таблицы], DESC [имя таблицы]. Определение количества записей в каждой таблице БД. Построение схемы связи таблиц в БД.

Практическое занятие 3. MySQL. Выборка данных из базы данных. Знакомство с оператором SELECT.

Цель: научиться извлекать данные из таблицы посредством простых запросов.

Содержание работы: Подключение к базе данных TV_DB. Вывод данных из таблицы с использованием оператора SELECT. Вывод определенного количества строк из таблицы с использованием оператора SELECT. Определение списка полей при выводе при выводе результата.

Практическое занятие 4. MySQL. Выборка данных с помощью условий

Цель: научиться извлекать данные из таблицы посредством запросов, содержащих различные условия.

Содержание работы: Подключение к базе данных TV_DB. Вывод из таблиц данных с использованием оператора «SELECT FROM WHERE <логическое выражение>». Проверка на вхождение во множество значений (IN (value1 [, value2, ... valuen])), проверка на значение null (is null), проверка на соответствие строки маске (like <pattern>), проверка на вхождение в диапазон (between <bottom> and <top>).

Практическое занятие 5. MySQL. Сортировка, упорядочивание данных, поиск минимального и максимального значения.

Цель: научиться извлекать данные из таблиц, сортируя результат - в алфавитном порядке для текста или по возрастанию/убыванию - для цифровых значений.

Содержание работы: Подключение к базе данных TV_DB. Используя оператор ORDER BY, выполнить сортировку выводимых данных можно по текстовым полям (от А к Я или от Я к А), числовым полям (от наименьших к наибольшим или от наибольших к наименьшим), а также датам и времени (от старых к новым или от новых к старым) в одном или нескольких полях таблицы. При этом можно задать возрастающую (ASC) или убывающую (DESC) последовательность сортировки для каждого поля. Используя возможности сортировки выполнить поиск максимальных и минимальных значений.

Практическое занятие 6. Использование групповых функций в запросах MySQL. Группировка данных с помощью GROUP BY.

Цель: научиться разделить все данные на логические наборы и применять агрегирующие функции, выполняя статистические вычисления отдельно в каждой группе.

Содержание работы: Подключение к базе данных TV_DB. Выполнение заданий по выводу результатов по различным группам данных, используя предложения GROUP BY оператора SELECT и групповые функции COUNT(), SUM(), MIN(), MAX(), AVG(). Осуществить фильтрацию по сгруппированным данным используя оператор HAVING. Выполнить сортировку результатов, полученных после фильтрации (оператор ORDER BY).

Практическое занятие 7. MySQL. Создание подзапросов.

Цель: научиться использовать подзапросы в различных операторах MySQL.

Содержание работы: Подключение к базе данных TV_DB. Выполнение заданий к лабораторной работе, применяя подзапросы в операторе SELECT: через условие в выражении WHERE; в условии в выражении HAVING; в качестве таблицы для выборки в выражении FROM. Использование подзапросов вместе с оператором IN, для выбора набора значений. Использование операторов ALL, SOME или ANY для реализации подзапросов. Использование подзапросов при добавлении, обновлении и удалении данных.

Практическое занятие 8. MySQL. Объединения таблиц в MySQL. Оператор JOIN.

Цель: изучить возможные виды соединения таблиц и научиться выражать отношения между таблицами в запросах MySQL.

Содержание работы: Подключение к базе данных TV_DB. Выполнение заданий по выводу данных из базы данных с использованием внутреннего соединения (INNER JOIN), внешнего соединения (OUTER JOIN), соединений слева и справа (LEFT JOIN и RIGHT JOIN). Выполнение запросов с исключением пересечения таблиц (UNION ALL).

Практическое занятие 9. Создание таблицы с помощью MySQL. Описание типов данных. Изменение структуры таблиц базы данных.

Цель: ознакомиться с возможностями MySQL по созданию базы данных и таблиц БД. Научиться изменять схему данных и структуру таблиц БД.

Содержание работы: Изучить набор команд языка SQL, связанный с созданием базы данных, созданием, модификацией структуры таблиц и их удалением, вставкой, модификацией и удалением записей таблиц. Подключиться к серверу MySQL. Создать локальную. базу данных по предложенному преподавателем заданию и набор таблиц в ней, используя команды: создание базы данных - CREATE DATABASE имя БД; выбор существующей базы данных - USE DATABASE имя БД; создание таблицы базы данных - CREATE TABLE. После задания активной БД можно с помощью средств, предоставляемых программой изменять структуру БД (ALTER DATABASE) и структуру таблиц согласно заданию, добавляя и удаляя колонки из таблиц, изменяя названия и описания полей.

Практическое занятие 10. Ввод данных в таблицы БД средствами MySQL. Модификация данных таблиц БД средствами MySQL.

Цель: ознакомиться с возможностями MySQL вставкой, модификацией и удалением записей таблиц.

Содержание работы: После создания пустых таблиц в предыдущей лабораторной работе наполните их данными, используя два подхода:

Вручную, используя команду: insert into <имя таблицы>[(<имя колонки>, ...)] values (<значение колонки>, ...)[, ...]. Введите в таблицу не менее 10 записей;

Загрузить данные из текстового файла в одну из таблиц созданной БД. Синтаксис команды LOAD DATA INFILE. Создайте текстовый файл (имя файла на ваше усмотрение) для ввода данных в таблицу БД. При создании текстового файла по умолчанию значения данных внутри записи отделяются друг от друга символом табуляции, а записи друг от друга - переводом строки (ENTER). Никаких дополнительных сведений внутри файла данных быть не должно. Введите в таблицу не менее 10 записей.

Раздел 2. Динамические веб-технологии. Язык PHP

Практические занятия 1. Язык программирования PHP: основы синтаксиса, встраивание, типы данных, выражения и операторы. Основные конструкции. Работа с файлами.

Практические занятия 2. Язык программирования PHP: обработка данных формы. Анализ примеров с кодами - обработчиками форм. Встраивание простейших скриптов. Выполнение интерактивных заданий на сайте https://htmlacademy.ru/courses/337/run/1 .

Практические занятия 3. **Работа с базами данных с помощью PHP. СУБД MySQL**
Цель работы: формирование умений применения PHP для доступа к базам данных, и формирование умений работы с MySQL.
Содержание лабораторной работы:
1. Зайдите на сервер в программу phpMyAdmin. Создайте базу данных (название БД должно начинаться с вашего login'a, например - vasy-catalog).
Создайте таблицу в этой БД со следующими полями (количество символов выставьте на свое усмотрение):
- Имя (индексируемое)
- Фамилия (уникальное)
- Отчество (индексируемое)
- e-mail (индексируемое)
- телефон (индексируемое)
- город (индексируемое)
- страна (индексируемое)
- фотография

Попробуйте загрузить, удалить и отредактировать данные.
2. Создайте форму для добавления записей в эту базу данных (должны присутствовать все поля). Создайте скрипт обработки этого запроса.
Создайте форму поиска к этой базе данных (должно присутствовать только поле "Фамилия"). Создайте скрипт обработки этого запроса.
Создайте форму запроса поиска к этой базе данных, для поиска записей, которые необходимо редактировать (должны присутствовать все поля). Создайте скрипт обработки этого запроса.
Создайте форму запроса поиска к этой базе данных, для поиска записей, которые необходимо удалить (должно присутствовать только поле "Фамилия"). Создайте скрипт обработки этого запроса.
Проверьте работоспособность на сервере.
Сделайте ссылки на первой странице.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ АРХИТЕКТУРНО-СТРОИТЕЛЬНЫЙ УНИВЕРСИТЕТ»

Институт непрерывного образования

СОГЛАСОВАНО
Директор ИНО

_____ Н.Р.Шадейко
(подпись)

"____"_____20___ г.

УТВЕРЖДАЮ
Проректор по учебной работе

_____ Д.Н. Песцов
(подпись)

"____"_____20___ г.

ПРОГРАММА МОДУЛЯ

Стажировка
(наименование дисциплины)

ДОПОЛНИТЕЛЬНАЯ ПРОФЕССИОНАЛЬНАЯ ПРОГРАММА

«Web-программирование»
(наименование программы)

Томск 2025

Составитель *профессор кафедры ПМ, Г.А. Онопенко*
(Должность, Ф.И.О., подпись)

Рецензент *профессор кафедры ПМ, Б.М. Шумилов*
(Должность, Ф.И.О., подпись)

Введена в действие с «___» _____ 20___ г.

СОДЕРЖАНИЕ

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ 4
1.1 Учебный план 4
1.2 Планируемые результаты обучения 4
2. СОДЕРЖАНИЕ ПРОГРАММЫ 5
2.1 Учебно-тематический план 5
3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ 5
3.1 Материально-технические условия реализации программы 5
3.2 Перечень информационных технологий, включая перечень программного обеспечения и информационных справочных систем 6
3.2.1. Перечень информационных технологий 6
3.2.2. Перечень программного обеспечения 6
3.2.3. Перечень информационных справочных систем 6
3.3 Учебно-методическое и информационное обеспечение модуля 7
3.3.1 Литература 7
3.3.2 Методические разработки 8
3.4. Процедура зачета 8
3.5. Кадровое обеспечение 8
ПРИЛОЖЕНИЕ 1 9
ПРИЛОЖЕНИЕ 2 10

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ

1.1 Учебный план

Наименование дисциплины	Общая трудоем-кость, час.	Всего, ауд. час.	Аудиторные занятия, час.			СРС, час.	Текущий контроль*, (шт.)			Промежуточная аттестация	
			лекции	лабораторные работы	прак. занятия, семинар		РК, РГР, реф.	КР	КП	Зачет	Экзамен
Модуль 4. Стажировка											
1. Стажировка в профильной организации	36	36			36						
Итого	**36**	**36**			**36**					**Зачет**	

1.2 Планируемые результаты обучения

Слушатель программы переподготовки будет обладать следующими профессиональными компетенциями:

Матрица компетенций дисциплины

Профессиональные компетенции	ID-28	ID-35
	Применяет языки программирования для решения профессиональных задач	Разрабатывает различные веб-архитектуры

Слушатель программы переподготовки будет способен осуществлять следующие трудовые действия:

- Выявление первоначальных требований заказчика к ИР (ID-28);
- Определение возможности достижения соответствия ИР первоначальным требованиям заказчика (ID-28);
- Проектирование структуры разделов ИР (ID-28);
- Разработка интерфейса пользователя для ИР с использованием стандартов в области web-разработки (ID-35);
- Создание прототипа интерфейса пользователя программными средствами проектирования (ID-28);
- Настройка контроля целостности файлов ИР и прикладного программного обеспечения (ID-35);
- Мониторинг атак на ИР и выполнение соответствующих действий (ID-35).

Слушатель программы переподготовки будет обладать следующим необходимым опытом практической деятельности:

- владеет навыками разработки и поддержки web-сайта с использованием современных web-технологий (ID-28, ОПД1);
- имеет навыки алгоритмизации и программной реализации типовых задач программирования (ID-28, ОПД2);
- владеет навыками применения стандартных средств программирования для моделирования объектов и процессов (ID-28, ОПД3);
- имеет навыки применения имеющейся технической и/или программной архитектуры (ID-35, ОПД1);

- владеет навыками использования методов и приемов формализации и алгоритмизации поставленных задач (ID-35, ОПД2);
- владеет навыками применения вспомогательных инструментальных программных средств проектирования программного обеспечения, структур данных, программных интерфейсов (ID-35, ОПД3).

Цель стажировки (сформулированная в соответствии с ФГОС3++ и Профессиональным стандартом).

- подготовка к решению производственных задач предприятия;
- закрепление и углубление теоретических знаний, полученных при изучении дисциплин учебного плана;
- приобретение и развитие необходимых практических умений и навыков в соответствии с требованиями к уровню подготовки выпускника;
- изучение современного состояния и направлений развития компьютерной техники и информационных технологий;
- закрепление и углубление практических навыков в области web-программирования;
- повышение уровня освоения компетенций в профессиональной деятельности.

2. СОДЕРЖАНИЕ ПРОГРАММЫ

2.1 Учебно-тематический план

№ п/п	Наименование разделов/тем	Общая трудоемкость, ч	Всего ауд. ч	По учебному плану с использованием дистанционных образовательных технологий, ч						СРС, ч
				Аудиторные занятия, ч			Дистанционные занятия, ч.			
				лекции	лабораторные работы	практические и семинарские занятия	лекции	лабораторные работы	практические и семинарские занятия	
1	2	3	4	5	6	7	8	9	10	11
1	**Стажировка в профильной организации**									
2	Этап 1. Подготовительный этап Теоретическая и техническая подготовка	*2*	*2*			*2*				
3	Этап 2. Основной этап прохождения практики: теоретическая часть	*15*	*15*			*15*				
4	Этап 3. Основной этап прохождения практики: практическая часть	*15*	*15*			*15*				
5	Этап 4. Заключительный этап	*4*	*4*			*4*				
6	**Итого:**	**36**	**36**			**36**				

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

3.1 Материально-технические условия реализации программы

Реализация рабочей программы стажировки предполагает прохождение ее на предприятиях в соответствии с договорами. Студент использует материально техническую базу организации, в которой он проходит стажировку.

Примерное оборудование:

- оргтехника;
- комплексы документов;
- информационные ресурсы.

Средства обучения:

- мультимедиа проектор;
- персональные компьютеры с лицензионным программным обеспечением;
- справочные правовые системы;
- Доступ к «Интернет» при СРС. Web-браузер.
- *Электронно-образовательная система: IDO.TSUAB.RU*
- Электронные библиотеки: электронная библиотечная система znanium.com, многопрофильный образовательный ресурс «Консультант студента» www.studentlibrary.ru, электронное издание ЭБС Юрайт https://www.biblio-online.ru/, электронно-библиотечная система ЭБС «Лань» https://e.lanbook.com/
- LibreOffice.org.

Наименование специализированных аудиторий, кабинетов, лабораторий	Вид занятий	Наименование оборудования, программного обеспечения
Компьютерный класс 304/1	практические занятия, консультации	Компьютеры, доска, специализированные программы, Интернет.

3.2 Перечень информационных технологий, включая перечень программного обеспечения и информационных справочных систем

3.2.1. Перечень информационных технологий

Студент использует материально техническую базу организации, в которой он проходит стажировку.

При проведении стажировки используются следующие информационные технологии:

- справочные правовые системы;
- Доступ к «Интернет», Web-браузер.
- *Электронно-образовательная система IDO.TSUAB.RU: https://digital.tsuab.ru/*

3.2.2. Перечень программного обеспечения

Студент использует материально техническую базу организации, в которой он проходит стажировку.

При проведении стажировки используется программное обеспечение:

- лицензионное программное обеспечение, использующееся организацией;
- Доступ к «Интернет», Web-браузер.
- Специализированные программы проектирования и разработки сайта: Notepad++; Sublime Text; Visual Studio Code; Ms mySQL и др.

3.2.3. Перечень информационных справочных систем

1. Научная электронная библиотека e-LIBRARY (электронный ресурс; режим доступа http://www.elibrary.ru/);

2. Электронно-библиотечная система «ZNANIUM.COM» (электронный ресурс; режим доступа http:// http://znanium.com/);

3. Электронно-библиотечная система РУКОНТ (электронный ресурс; режим доступа https://lib.rucont.ru/);

4. Система электронного обучения ТГАСУ (https://ido.tsuab.ru/); (https://digital.tsuab.ru/);

5. Научно-техническая библиотека ТГАСУ (https://lib.tsuab.ru/);

6. Академия HTML. Сервис интерактивного онлайн-обучения [электронный ресурс] / Режим доступа: http://www.htmlacademy.ru/

7. Справочник CSS. Изучаем с нуля [электронный ресурс] / Режим доступа: http://www.eltisbook.ru/category/css

8. Электронные книги и учебники по веб-дизайну и веб-программированию для начинающих [электронный ресурс] / Режим доступа: https://html5beginner.github.io/books/index.html

9. Справочные и учебные материалы для создания сайтов [электронный ресурс] / Режим доступа: https://basicweb.ru/

10. Библиотека программиста [электронный ресурс] / Режим доступа: https://hhrd.ru/library/243

11. Сайт по эффективному программированию на PHP [Электронный ресурс] / Режим доступа: http://phpfaq.ru.

12. Базы данных Oracle [электронный ресурс] / Режим доступа: https://oracleplsql.ru/

13. Лаборатория линуксоида [электронный ресурс] / Режим доступа: https://younglinux.info/sqlite/sqlite

Нормативные документы:

1. Федеральный закон от 27.07.2006 № 149 ФЗ «Об информации, информационных технологиях и защите информации».

3.3 Учебно-методическое и информационное обеспечение модуля

3.3.1 Литература

1. Гуриков, С.Р. Интернет-технологии: Учебное пособие / С.Р. Гуриков. - М.: Форум: НИЦ ИНФРА-М, 2015. - 184 с. – Режим доступа: http://znanium.com/go.php?id=488074
2. Матюшок, В.М. Информатика для экономистов [Электронный ресурс] учебник / В.М. Матюшок. – М.: ООО "Научно-издательский центр ИНФРА-М", 2016. 460 с. – Режим доступа: http://znanium.com/go.php?id=541005
3. Дунаев В. HTML, скрипты и стили / Дунаев В. – 3-е изд., переб. и доп. – СПб.: БХВ-Петербург, 2011. – 810 с. – Режим доступа: http://znanium.com/catalog.php?bookinfo=350371
4. Дронов В. А. HTML 5, CSS 3 и Web 2.0. Разработка современных Web-сайтов. – СПб.: БХВ-Петербург, 2011. – 414 с. Режим доступа: http://znanium.com/catalog.php?bookinfo=351455
5. Пилгрим М. Погружение в HTML5/ Пилгрим М.- перев. с англ. – СПб.: БХВ-Петербург, 2011. – 294 с. – Режим доступа: http://znanium.com/catalog.php?bookinfo=355295.
6. Панфилов К. Создание веб-сайта от замысла до реализации. [Электронный ресурс]/ К. Панфилов. – М.: ДМК Пресс, 2009. – 440 с. – Режим доступа: http://znanium.com/catalog.php?item= booksearch&code = HTML&page=2
7. Загуменов, А.П. Как раскрутить и разрекламировать Web-сайт в сети Интернет [Электронный ресурс] / А. П. Загуменов. – М.: ДМК Пресс, 2007. – 384 с. – Режим доступа: http://znanium.com/catalog.php?item=booksearch&code=HTML
8. Попов Е.Б. Профессиональный иностранный язык: английский язык [Электронный ресурс] / Е.Б. Попов. – М.: НИЦ ИНФРА - М, 2016. – 150 с. Режим доступа: http://znanium.com/catalog.php?bookinfo=760140
9. Янцев В.В. JavaScript и PHP. Content management system: учебное пособие для вузов / Янцев В.В. - Санкт-Петербург: Лань, 2022. - 192 с. URL: https://e.lanbook.com/book/266657.
10. Никулова Г.А. WEB-программирование. Серверные технологии: PHP. Часть 1: Учебно-методическое пособие / Никулова Г. А., Субботин В. Р. - Липецк: Липецкий ГПУ, 2017. – 59 с. URL: https://e.lanbook.com/book/111934.
11. Заяц А. М. Основы WEB технологий. Разработка WEB-приложений современными инструментальными средствами: учебно-методическое пособие для бакалавров по

направлению подготовки 09.03.02 информационные системы и технологии / Заяц А.М. - Санкт-Петербург: СПбГЛТУ, 2021. - 116 с. URL: https://e.lanbook.com/book/191164.
12. Ружицкая Е.А. Основы web-технологий: работа с данными формы в PHP: практическое пособие / Ружицкая Е.А., Кечко Е.П. - Гомель: ГГУ имени Ф. Скорины, 2022. - 47 с. URL: https://e.lanbook.com/book/320990.
13. Филиппов Ф.В. HTTP + PHP в примерах и задачах: учебное пособие / Филиппов Ф.В., Губин А.Н. - Санкт-Петербург: СПбГУТ им. М.А. Бонч-Бруевича, 2015. - 67 с. URL: https://e.lanbook.com/book/180044.
14. Кожевникова П.В. PHP и MySQL: учебное пособие / Кожевникова П.В. - Ухта: УГТУ, 2020. - 51 с. URL: https://e.lanbook.com/book/209591

3.3.2 Методические разработки

1. Технология создания web-страниц: методические указания к самостоятельной работе и задания к контрольной работе / сост. Э.А. Эшаров, Г.А. Онопенко, Е.Ю. Корниенко. – Томск: Изд-во Том. гос. архит.- строит. ун-та, 2015. – 44 с. Режим доступа: http://portal.tsuab.ru/Metodi_/2015/met038_2015.pdf.
2. Основы каскадных таблиц стилей: методические указания к самостоятельной работе и задания к контрольной работе / сост. Э.А. Эшаров, Г.А. Онопенко, Б.М. Шумилов. – Томск: Изд-во Том. гос. архит.- строит. ун-та, 2016. – 46 с.

3.4 Процедура зачета

Формой промежуточной аттестации является зачет. Зачет состоит из проверки практических навыков и теоретической подготовки.

Защита стажировки состоит из ответов на вопросы по выполненному заданию (или теста).

Шкала оценивания

Зачтено	Выставляется слушателю в случае, если уровень выполнения, предусмотренной зачетом работы, отвечает большинству требований, теоретическое содержание стажировки освоено полностью или частично, но пробелы не носят существенного характера, необходимые навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения заданий практики выполнено.
Не зачтено	Выставляется слушателю в случае, если уровень выполнения, предусмотренной зачетом работы слабый, уровень выполнения не отвечает большинству требований, теоретическое содержание стажировки освоено частично, некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения задания не выполнены, либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному.

В случае неудовлетворительной оценки слушатель имеет право пересдать зачет в установленном порядке.

3.5 Кадровое обеспечение

Авторы и преподаватели:

№ п/п	ФИО, должность	Модули (темы, лекции)	Часов, всего
1.	Профильная ИТ-организация (ООО «ТН Диджитал» и другие)	Модуль 4. Стажировка.	36

ПРИЛОЖЕНИЕ 1

АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ МОДУЛЯ

Стажировка

(название модуля)

1. Цели и задачи изучения модуля.

Закрепление теоретической подготовки, полученной во время аудиторных занятий и самостоятельной работы, подготовка к решению производственных задач предприятия; приобретение и развитие необходимых практических умений и навыков в соответствии с требованиями к уровню подготовки выпускника; накопление студентами профессиональных знаний и практических навыков.

Задачи прохождения стажировки:

- обобщение, систематизация, конкретизация и закрепление теоретических знаний, полученных при освоении образовательной программы на основе изучения опыта работы конкретного предприятия (организации, учреждения);
- приобретение практического опыта работы в коллективе;
- анализ и изучение опыта разработки и применения конкретных профессиональных программ и систем информационного обеспечения для решения реальных задач в условиях конкретных заданий заказчика и их реализации в команде организации или веб-студии;
- приобретение навыков практического решения информационных задач на конкретном рабочем месте в качестве исполнителя или стажера;
- приобретение практического опыта работы в команде;
- сбор, анализ и обработка необходимых материалов для подготовки к итоговому проекту по программе.

2. Содержание компетенций, формируемых при прохождении модуля.

ID-28. Применяет языки программирования для решения профессиональных задач.
ID-35. Разрабатывает различные веб-архитектуры.

3. Планируемые результаты прохождения модуля.

- владеет навыками разработки и поддержки web-сайта с использованием современных web-технологий (ID-28, ОПД1);
- имеет навыки алгоритмизации и программной реализации типовых задач программирования (ID-28, ОПД2);
- владеет навыками применения стандартных средств программирования для моделирования объектов и процессов (ID-28, ОПД3);
- имеет навыки применения имеющейся технической и/или программной архитектуры (ID-35, ОПД1);
- владеет навыками использования методов и приемов формализации и алгоритмизации поставленных задач (ID-35, ОПД2);
- владеет навыками применения вспомогательных инструментальных программных средств проектирования программного обеспечения, структур данных, программных интерфейсов (ID-35, ОПД3).

4. Тематическое содержание модуля.

Этап 1. Общие сведения о практике. Теоретическая и техническая подготовка.
Этап 2. Выбор и формулирование темы работы на стажировке. Знакомство с местом прохождения практики, изучение организационной структуры веб-студии.
Этап 3. Изучение и совершенствование профессиональных навыков. Сбор данных и выполнение индивидуального задания.
Этап 4. Заключительный этап. Подготовка к зачёту по результатам стажировки.

ПРИЛОЖЕНИЕ 2

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО МОДУЛЮ

Составитель профессор, Г.А. Онопенко
(должность, ФИО, подпись)

Рецензент профессор, Б.М. Шумилов
(должность, ФИО, подпись)

Приложение 2.1. Форма, вид и порядок отчетности обучающихся о прохождении стажировки

По результатам аттестации выставляется отметка (зачтено/ не зачтено). Результаты защиты стажировки отражаются в ведомости. Студенты, не сдавшие в установленные сроки материалы по стажировке, считаются имеющими академическую задолженность.

Задание на стажировку формируется руководителем предприятия.

Стажировка проводится непрерывно, на неё отведено 36 часов практических занятий. Форма промежуточной аттестации – **зачет**.

Процесс прохождения стажировки состоит из нескольких этапов:

- подготовительный этап (инструктаж по технике безопасности, знакомство с коллективом, рабочим местом, подписанием договора о неразглашении (если необходимо) пакетами прикладных программ, составлением индивидуального плана работы;
- проектный этап;
- этап анализа полученных результатов, формирование выводов.
- подготовка к зачету по результатам стажировки.

Проектный этап предполагает закрепление полученных знаний в процессе обучения посредством разработки индивидуального/группового проекта по техническому заданию профильной организации.

Требования к содержанию проекта: разработка web-сайта (не менее трёх страниц, связанных между собой), на основе технологий HTML, CSS и JavaScript (без использования конструкторов), реализации блочной или фреймовой модели сайта с использованием элементов HTML, каскадных таблиц стилей CSS (в том числе классов), управляющих элементов языка JavaScript.

За период стажировки слушатель должен изучить и выполнить поставленную перед ним задачу, применяя специализированное программное обеспечение. Аттестация по стажировке осуществляется в форме зачета.

Результаты стажировки являются основой для выполнения итоговой аттестации, что предполагает модификацию и доработку индивидуального/группового проекта.

Приложение 2.2. Методические указания для обучающихся по прохождению стажировки

Стажировка проводится в соответствии с учебным планом. Стажировка представляет собой вид учебных, практических и самостоятельных занятий, непосредственно ориентированных на профессионально-практическую подготовку слушателя.

Каждый слушатель согласовывает с руководителем цель, тему предстоящей работы, определяет объем и порядок сбора материалов, достаточных для решения поставленных задач.

Первый этап: заключение договоров с предприятиями и организациями, составление приказа на стажировку, проведение организационного собрания, выдача заданий на стажировку.

Второй этап: организация прохождения студентами инструктажа по безопасности и охране труда в организации, знакомство с правилами внутреннего распорядка, составление календарного графика работы студентов с указанием порядка и сроков выполнения программы стажировки.

Третий и четвёртый этапы предполагают закрепление полученных знаний в процессе обучения посредством разработки индивидуального/группового проекта по техническому заданию профильной организации/университета.

Руководитель стажировки от предприятия организует и контролирует её прохождение в соответствии с программой и утвержденным графиком, помогает студентам в сборе материалов для проекта, намечает пути решения, поставленных перед студентами задач.