

**Бюджетное общеобразовательное учреждение
«Тарская средняя общеобразовательная школа №3»
Тарского муниципального района Омской области
Центр цифрового образования детей «IT-куб»**

Рассмотрено на
заседании
педагогического
совета
Протокол № 2 от
30.08.2024

Согласовано:
Руководитель Центра
цифрового образования «IT-
куб» БОУ «Тарская СОШ №3»
Л.А. Клименко
30.08. 2024 г.

«Утверждаю»
И.о. директора БОУ
«Тарская СОШ №3»
Е.А. Ларионова
Приказ № 86 от 30.08.2024



**Дополнительная общеобразовательная общеразвивающая программа
«Web-разработка»**

Направленность: техническая

Целевая группа: 11 - 18 лет

Срок реализации: 2 года

Общая трудоемкость: 180 часов

Форма реализации: очная

Уровень сложности содержания: базовый, углубленный

Автор - составитель:
Моисеенко Никита Сергеевич
педагог дополнительного образования
Центра цифрового образования «IT-Куб»
БОУ «Тарская средняя общеобразовательная школа №3»
Тарского МР Омской области

Тара, 2024

Оглавление

Пояснительная записка	3
Учебно-тематическое планирование	9
Содержание программы	13
Контрольно-оценочные средства	30
Условия реализации программы	32
Список литературы	34
Приложение 1	36
Приложение 2	40

Пояснительная записка

Дополнительная общеобразовательная (общеразвивающая) многоуровневая программа разработана на основе педагогического опыта автора-составителя программы, рабочей программы по направлению «Web-разработка».

Направленность программы: Дополнительная общеобразовательная (общеразвивающая) программа «Web-разработка» (далее - Программа), технической направленности и предназначена педагогам для использования в системе дополнительного образования детей.

Новизна данной дополнительной образовательной программы заключается в том, что по форме организации образовательного процесса она является модульной. Таким образом, новизна учебной программы может заключаться в интеграции современных технологий и подходов, которые помогут обучающимся не только овладеть базовыми навыками, но и подготовиться к работе с последними тенденциями и требованиями в веб-разработке.

Реализация программы направлена на формирование научного мировоззрения, развитие мышления посредством изучения вопросов программирования, на формирование у обучающихся информационной культуры и грамотности, приобщение к информационным технологиям.

Глобальная информатизация и компьютеризация общества предъявляют высокие требования к подрастающему поколению, которому необходимо обладать высоким уровнем компьютерной грамотности, уметь быстро находить необходимую информацию, оперативно ее обрабатывать, передавать, хранить и грамотно представлять.

Возраст детей, участвующих в реализации данной дополнительной образовательной программы, колеблется от 11 до 18 лет.

Сроки реализации и трудоёмкость программы: 2 года. (180 часов)

Режим занятий:

1 год обучения (11-15 лет) - 2 раза в неделю по 1 академическому часу для Базового уровня

2 год обучения (12-18 лет) - 3 раза в неделю по 1 академическому часу для Углубленного уровня.

Условия набора и добора обучающихся: Набор детей в группы происходит по желанию на любой год обучения (уровень сложности). Добор по программе не предусмотрен.

Наполняемость в группе: 12 обучающихся.

Форма обучения: очная.

Уровень освоения программы: базовый и углубленный

Особенности организации образовательного процесса: так как данная программа является модульной, схема построения содержания модульной программы является комбинированной.

Цель программы: развитие интереса обучающихся к web-разработке; реализация их творческих идей в области программирования в виде проектов различного уровня сложности.

Задачи программы:

- обучить основам вёрстки web-сайтов;
- научить использованию каскадных таблиц стилей;
- способствовать формированию творческого отношения к выполняемой работе;
- развивать творческую инициативу и самостоятельность;
- развивать психофизиологические качества обучающихся: память, внимание, способность логически мыслить, анализировать, концентрировать внимание на главном;
- развивать умения излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений.

Планируемые результаты

Базовый уровень (1 год обучения):

Личностные результаты

- Способность разрабатывать визуально привлекательные веб-страницы, применяя основы веб-дизайна и цветовой гармонии.
- Умение создавать уникальные и оригинальные дизайны, отражающие личные предпочтения и креативный подход.
- Способность самостоятельно изучать и применять новые технологии и методы веб-разработки, искать решения проблем и устранять ошибки.
- Умение создавать качественный, валидный и совместимый код, а также проверять и тестировать результаты своей работы.
- Стремление продолжать изучение веб-технологий и развитие навыков в этой области.
- Умение ставить перед собой цели в освоении веб-разработки и последовательно их достигать.

Предметные результаты

- Знание структуры HTML-документа, умение создавать и использовать базовые элементы HTML (теги, атрибуты, заголовки, абзацы, списки, таблицы).
- Умение применять семантические элементы для создания понятной и доступной структуры контента (header, footer, article, section и т.д.).
- Способность создавать формы для ввода данных и использовать элементы управления (текстовые поля, кнопки, чекбоксы, радиокнопки).
- Владение основами CSS для стилизации HTML-элементов, включая цвет, шрифты, отступы, границы и фоны.
- Знание методов позиционирования элементов, включая использование Flexbox и Grid для создания адаптивных макетов.
- Способность создавать адаптивные веб-страницы, которые корректно отображаются на различных устройствах и экранах (мобильные телефоны, планшеты, десктопы).

- Умение работать с инструментами разработчика в браузере для отладки и тестирования HTML и CSS.

- Способность проверять валидность кода и тестировать его в различных браузерах для обеспечения кроссбраузерной совместимости.

Метапредметные результаты

- Умение анализировать и решать задачи, связанные с веб-дизайном и версткой, находить и устранять ошибки в коде.

- Способность оптимизировать код для повышения производительности веб-страниц и улучшения пользовательского опыта.

- Навыки планирования, разработки и реализации веб-проектов, включая создание макетов, верстку и стилизацию страниц.

- Способность работать над проектами в команде, сотрудничать с другими разработчиками и дизайнерами, обмениваться идеями и получать обратную связь.

- Умение находить и использовать ресурсы для обучения и решения проблем, связанные с HTML и CSS (документация, форумы, онлайн-курсы).

- Способность документировать процесс разработки и представлять свои работы другим, включая написание технических описаний и создание презентаций.

Углубленный уровень (2 год обучения):

Личностные результаты

- Способность создавать уникальные и креативные веб-дизайны.

- Умение внедрять собственные идеи и решения в веб-проекты, демонстрируя личное творчество и инновации.

- Способность самостоятельно искать информацию и ресурсы для решения проблем, связанных с веб-разработкой.

- Умение создавать и поддерживать качественный, читаемый и оптимизированный код, а также исправлять ошибки и улучшать функциональность.

- Стремление изучать новые технологии и инструменты, связанные с веб-разработкой, и применять их в проектах.

- Умение ставить и достигать конкретные цели в обучении веб-разработке, такие как завершение проектов и изучение новых технологий.

Предметные результаты

- Знание структуры HTML-документа и умение использовать базовые HTML-элементы для разметки веб-страниц.

- Способность использовать семантические теги для создания логичной и доступной структуры контента.

- Умение создавать и стилизовать формы для ввода данных, включая различные элементы управления (текстовые поля, кнопки, флажки).

- Владение основами CSS для стилизации элементов, создания макетов и управления внешним видом веб-страниц.

- Способность создавать адаптивные веб-страницы, которые корректно отображаются на различных устройствах с использованием медиа-запросов и гибких макетов.

- Основные навыки работы с CSS-препроцессорами (например, Sass или LESS) для упрощения написания и управления стилями.

- Знание синтаксиса JavaScript, включая переменные, операторы, функции, объекты и массивы.

- Умение манипулировать элементами веб-страницы с помощью JavaScript, включая изменение содержимого, стилей и атрибутов элементов.

- Способность обрабатывать пользовательские события (щелчки, ввод данных и т.д.) и реагировать на них, используя обработчики событий.

Метапредметные результаты

- Способность анализировать и отлаживать HTML, CSS и JavaScript код, выявлять и исправлять ошибки.

- Умение оптимизировать код для повышения производительности и улучшения пользовательского опыта.

- Навыки планирования, разработки и реализации веб-проектов, включая создание макетов, верстку страниц и реализацию интерактивности.
- Способность работать над проектами в команде, эффективно обмениваться идеями.
- Умение находить и использовать документацию, обучающие ресурсы и форумы для решения задач и освоения новых технологий.
- Способность документировать процесс разработки, создавать техническую документацию и представлять свои решения и проекты другим.

2. Учебно-тематическое планирование

Базовый уровень

№ раздела	Название раздела, тема занятия	Количество часов
1	Введение в веб-разработку	6
	Тема 1.1. Вводное занятие. Что такое веб-разработка?	1
	Тема 1.2. Языки веб-разработки: HTML, CSS.	1
	Тема 1.3. Структура веб-страницы: клиент и сервер.	1
	Тема 1.4. Установка и настройка редактора кода (Notepad++).	1
	Тема 1.5. Знакомство с браузерами и инструментами разработчика.	2
2	Основы HTML	20
	Тема 2.1. Введение в HTML: история и предназначение.	2
	Тема 2.2. Структура HTML-документа: doctype, html, head, body.	2
	Тема 2.3. Основные HTML-теги: заголовки, абзацы, списки.	2
	Тема 2.4. Работа с изображениями: тег .	2
	Тема 2.5. Создание гиперссылок: тег <a>.	2
	Тема 2.6. Формы: ввод данных с помощью тегов <form>, <input>, <textarea>.	2
	Тема 2.7. Таблицы: создание и стилизация с помощью тегов <table>.	2
	Тема 2.8. Семантические теги: <header>, <footer>, <article>, <section>.	2
	Тема 2.9. Встраивание медиафайлов: аудио и видео.	4
3	Основы CSS	22
	Тема 3.1. Введение в CSS: назначение и использование.	2
	Тема 3.2. Подключение CSS: встроенные, внутренние и внешние стили.	4
	Тема 3.3. Селекторы: типы селекторов (по тегу, классу, ID).	2
	Тема 3.4. Цвета и шрифты: использование свойств color и font-family.	2
	Тема 3.5. Блочная модель: margin, padding, border.	2
	Тема 3.6. Работа с фоном: свойства background-color и background-image.	2
	Тема 3.7. Стилизация текста: размеры, жирность, интерлиньяж.	2
	Тема 3.8. Flexbox: основные концепции и свойства.	2

	Тема 3.9. Grid Layout: создание сеток на странице.	2
	Тема 3.10. Анимации и переходы: использование свойств transition и animation.	2
4	Введение в адаптивный дизайн	10
	Тема 4.1. Что такое адаптивный дизайн? Зачем он нужен?	2
	Тема 4.2. Медиа-запросы: использование и примеры.	4
	Тема 4.3. Основы мобильного дизайна.	4
5	Проектная деятельность	14
	Тема 5.1. Обсуждение идей для итогового проекта.	2
	Тема 5.2. Планирование и экранные концепты.	4
	Тема 5.3. Реализация проекта с использованием всех изученных технологий.	4
	Тема 5.4. Презентация итогового проекта.	2
	Всего:	72

Углубленный уровень

№ раздела	Название раздела, тема занятия	Количество часов
1	Введение в веб-разработку	6
	Тема 1.1. Вводное занятие. Что такое веб-разработка?	1
	Тема 1.2. Языки веб-разработки: HTML, CSS, JavaScript.	1
	Тема 1.3. Структура веб-страницы: клиент и сервер.	1
	Тема 1.4. Установка и настройка редактора кода (Notepad++).	1
	Тема 1.5. Знакомство с браузерами и инструментами разработчика.	1
2	Основы HTML	6
	Тема 2.1. Введение в HTML: структура HTML-документа (doctype, html, head, body).	1
	Тема 2.2. Основные HTML-теги: заголовки, абзацы, списки, ссылки и изображения.	1
	Тема 2.3. Формы: создание простых форм с использованием различных элементов.	1
	Тема 2.4. Семантические теги: использование <header>, <footer>,	1

	<article>, <section>.	
	Тема 2.5. Встраивание медиафайлов: работа с аудио и видео.	2
3	Основы CSS	8
	Тема 3.1. Введение в CSS: подключение стилей к HTML-документу.	1
	Тема 3.2. Селекторы и свойства: базовые селекторы и их применение.	2
	Тема 3.3. Блочная модель: margin, padding, border.	1
	Тема 3.4. Стилизация текста и использование шрифтов.	2
	Тема 3.5. Flexbox: основные принципы и применение.	2
4	Основы JavaScript	20
	Тема 4.1. Введение в JavaScript: назначение и история.	2
	Тема 4.2. Подключение JavaScript к HTML: встроенный и внешний код.	2
	Тема 4.3. Типы данных в JavaScript: строки, числа, булевы значения.	2
	Тема 4.4. Переменные: var, let, const.	2
	Тема 4.5. Условия: операторы if, else, switch.	2
	Тема 4.6. Циклы: for, while, do while.	2
	Тема 4.7. Функции: объявление и вызов.	2
	Тема 4.8. Объекты и массивы: создание и использование.	2
	Тема 4.9. Методы и свойства массивов.	2
	Тема 4.10. Работа с событиями: обработка кликов, наведения и других событий.	2
5	DOM и взаимодействие с веб-страницей	16
	Тема 5.1. Что такое DOM? Введение.	2
	Тема 5.2. Динамическое изменение HTML с помощью JavaScript.	2
	Тема 5.3. Поиск элементов в DOM: методы getElementById, querySelector.	2
	Тема 5.4. Изменение CSS-свойств с помощью JavaScript.	2
	Тема 5.5. Создание и удаление элементов на странице.	2
	Тема 5.6. Обработка событий: addEventListener.	2
	Тема 5.7. Формы и JavaScript: обработка данных формы.	2
	Тема 5.8. Валидация форм с помощью JavaScript.	2
6	Введение в адаптивный дизайн	10

	Тема 6.1. Что такое адаптивный дизайн? Зачем он нужен?	2
	Тема 6.2. Медиа-запросы: использование и примеры.	2
	Тема 6.3. Основы мобильного дизайна.	2
	Тема 6.4. Сетки и фреймы для адаптивного дизайна.	2
	Тема 6.5. Использование фреймворков: Bootstrap и его возможности.	2
7	Основы работы с API	8
	Тема 7.1. Что такое API? Введение и примеры.	2
	Тема 7.2. Получение данных с помощью Fetch API.	2
	Тема 7.3. Обработка JSON-данных.	2
	Тема 7.4. Отправка данных на сервер: метод POST.	2
8	Создание простого проекта	10
	Тема 8.1. Проект: создание личной страницы.	2
	Тема 8.2. Дизайн и макет страницы.	2
	Тема 8.3. Реализация адаптивности.	2
	Тема 8.4. Интерактивные элементы: форма обратной связи.	2
	Тема 8.5. Обработка данных и их валидация.	2
9	Расширенные темы	10
	Тема 9.1. Основы TypeScript: введение.	2
	Тема 9.2. Работа с библиотеками и фреймворками (например, jQuery).	2
	Тема 9.3. Введение в системы контроля версий: Git.	2
	Тема 9.4. Развертывание сайта: хостинг и домены.	2
	Тема 9.5. Оптимизация производительности веб-сайтов.	2
10	Проектная деятельность	14
	Тема 10.1. Обсуждение идей для итогового проекта.	2
	Тема 10.2. Планирование и экранные концепты.	4
	Тема 10.3. Реализация проекта с использованием всех изученных технологий.	4
	Тема 10.4. Презентация итогового проекта.	2
	Всего:	108

3. Содержание программы

Базовый уровень

Раздел 1. Введение в веб-разработку

Тема 1.1. Вводное занятие. Что такое веб-разработка?

Форма проведения занятия: Беседа.

Виды учебной деятельности: Обсуждение понятия "веб-разработка", ознакомление с основными направлениями и задачами веб-разработки.

Терминология: Веб-разработка, фронтенд, бэкенд, клиент, сервер, веб-страница, веб-сайт.

Тема 1.2. Языки веб-разработки: HTML, CSS.

Форма проведения занятия: лекция.

Виды учебной деятельности: просмотр примеров кода, обсуждение функций языков, вопросы и ответы.

Терминология: HTML (HyperText Markup Language), CSS (Cascading Style Sheets), теги, селекторы, свойства, правила.

Тема 1.3. Структура веб-страницы: клиент и сервер

Форма проведения занятия: лекция.

Виды учебной деятельности: изучение схем и диаграмм, участие в обсуждении, работа с примерами.

Терминология: клиентская часть, серверная часть, HTTP/HTTPS, запросы и ответы, веб-страница, веб-приложение.

Тема 1.4. Установка и настройка редактора кода (Notepad++).

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: установка редактора, настройка среды, выполнение практических заданий.

Терминология: редактор кода, интерфейс, настройки, плагины.

Тема 1.5. Знакомство с браузерами и инструментами разработчика.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: исследование инструментов разработчика в браузерах, выполнение упражнений, анализ кода.

Терминология: браузер, инструменты разработчика, консоль, инспектор элементов, дебаггинг.

Раздел 2. Основы HTML

Тема 2.1. Введение в HTML: история и предназначение.

Форма проведения занятия: лекция.

Виды учебной деятельности: прослушивание лекции, обсуждение, работа с историческими примерами.

Терминология: HTML, история HTML, разметка, гипертекст, элементы.

Тема 2.2. Структура HTML-документа: doctype, html, head, body.

Форма проведения занятия: лекция.

Виды учебной деятельности: анализ структуры HTML-документа, создание примеров кода.

Терминология: doctype, html, head, body, элементы, атрибуты.

Тема 2.3. Основные HTML-теги: заголовки, абзацы, списки.

Форма проведения занятия: лекция с практическим заданием.

Виды учебной деятельности: изучение тегов, написание примеров кода, выполнение упражнений.

Терминология: теги, заголовки (h1-h6), абзацы (p), списки (ul, ol, li).

Тема 2.4. Работа с изображениями: тег .

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: вставка изображений на страницу, настройка атрибутов тега.

Терминология: тег , атрибуты src, alt, width, height.

Тема 2.5. Создание гиперссылок: тег <a>.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: создание гиперссылок, работа с атрибутами тега.

Терминология: тег <a>, атрибут href, ссылки, якоря.

Тема 2.6. Формы: ввод данных с помощью тегов <form>, <input>, <textarea>.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: создание форм, работа с различными типами элементов формы.

Терминология: тег <form>, элементы <input>, <textarea>, атрибуты name, type, placeholder.

Тема 2.7. Таблицы: создание и стилизация с помощью тегов <table>.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: создание таблиц, добавление строк и ячеек, стилизация таблиц.

Терминология: тег <table>, <tr>, <td>, <th>, атрибуты border, colspan, rowspan.

Тема 2.8. Семантические теги: <header>, <footer>, <article>, <section>.

Форма проведения занятия: лекция с практическим заданием.

Виды учебной деятельности: использование семантических тегов, создание примеров страниц.

Терминология: семантические теги, <header>, <footer>, <article>, <section>.

Тема 2.9. Встраивание медиафайлов: аудио и видео.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: вставка медиафайлов на страницу, работа с тегами и атрибутами.

Терминология: теги <audio>, <video>, атрибуты src, controls, autoplay, loop.

Раздел 3. Основы CSS

Тема 3.1. Введение в CSS: назначение и использование.

Форма проведения занятия: лекция.

Виды учебной деятельности: изучение роли CSS, обсуждение примеров.

Терминология: CSS, стилизация, правила стилей, селекторы.

Тема 3.2. Подключение CSS: встроенные, внутренние и внешние стили.

Форма проведения занятия: лекция.

Виды учебной деятельности: подключение стилей к HTML-документу, работа с различными типами стилей.

Терминология: встроенные стили, внутренние стили, внешние стили, теги <style>, <link>.

Тема 3.3. Селекторы: типы селекторов (по тегу, классу, ID).

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: написание правил стилей с использованием разных селекторов.

Терминология: селекторы, теги, классы, ID, псевдоклассы.

Тема 3.4. Цвета и шрифты: использование свойств color и font-family.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: изменение цветов и шрифтов на странице, работа со свойствами.

Терминология: color, font-family, цветовые модели, шрифты.

Тема 3.5. Блочная модель: margin, padding, border.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение блочной модели, работа с отступами и границами.

Терминология: блочная модель, margin, padding, border, box-sizing.

Тема 3.6. Работа с фоном: свойства background-color и background-image.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: стилизация фона элементов, работа с цветом и изображениями.

Терминология: background-color, background-image, background-repeat, background-size.

Тема 3.7. Стилизация текста: размеры, жирность, интерлиньяж.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: изменение стиля текста, работа с различными свойствами.

Терминология: font-size, font-weight, line-height, текстовые свойства.

Тема 3.8. Flexbox: основные концепции и свойства.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение свойств Flexbox, создание макетов с помощью Flexbox.

Терминология: Flexbox, flex-direction, justify-content, align-items.

Тема 3.9. Grid Layout: создание сеток на странице.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: создание сеток, работа с Grid-свойствами.

Терминология: Grid Layout, grid-template-columns, grid-template-rows, grid-area.

Тема 3.10. Анимации и переходы: использование свойств transition и animation.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: создание анимаций и переходов, работа с CSS-анимацией.

Терминология: transition, animation, keyframes, duration, easing.

Раздел 4. Введение в адаптивный дизайн

Тема 4.1. Что такое адаптивный дизайн? Зачем он нужен?

Форма проведения занятия: лекция.

Виды учебной деятельности: обсуждение целей адаптивного дизайна, анализ примеров.

Терминология: адаптивный дизайн, респонсивный дизайн, медиа-запросы, пользовательский опыт.

Тема 4.2. Медиа-запросы: использование и примеры.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: создание медиа-запросов, работа с примерами адаптивных стилей.

Терминология: медиа-запросы, @media, breakpoints, адаптивные стили.

Тема 4.3. Основы мобильного дизайна.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение принципов мобильного дизайна, создание макетов для мобильных устройств.

Терминология: мобильный дизайн, мобильная версия, UX/UI для мобильных устройств.

Раздел 5. Проектная деятельность

Тема 5.1. Обсуждение идей для итогового проекта.

Форма проведения занятия: беседа.

Виды учебной деятельности: обсуждение и презентация идей, оценка возможностей реализации.

Терминология: концепция проекта, мозговой штурм, цели проекта.

Тема 5.2. Планирование и экранные концепты.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: создание экранных концептов, планирование структуры проекта.

Терминология: экранные концепты, wireframes, прототипирование.

Тема 5.3. Реализация проекта с использованием всех изученных технологий.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: работа над проектом, применение изученных технологий, тестирование.

Терминология: интеграция технологий, тестирование, отладка, реализация.

Тема 5.4. Презентация итогового проекта.

Форма проведения занятия: защита проектов.

Виды учебной деятельности: представление проекта, ответы на вопросы, получение обратной связи.

Терминология: презентация, защита проекта, обратная связь, демонстрация.

Углубленный уровень

Раздел 1. Введение в веб-разработку

Тема 1.1. Вводное занятие. Что такое веб-разработка?

Форма проведения занятия: Беседа.

Виды учебной деятельности: Обсуждение понятия "веб-разработка", ознакомление с основными направлениями и задачами веб-разработки.

Терминология: Веб-разработка, фронтенд, бэкенд, клиент, сервер, веб-страница, веб-сайт.

Тема 1.2. Языки веб-разработки: HTML, CSS, JavaScript.

Форма проведения занятия: Лекция.

Виды учебной деятельности: Изучение основных языков веб-разработки, их назначение и взаимосвязь.

Терминология: HTML, CSS, JavaScript, разметка, стилизация, интерактивность.

Тема 1.3. Структура веб-страницы: клиент и сервер.

Форма проведения занятия: Беседа.

Виды учебной деятельности: Обсуждение клиент-серверной архитектуры, роли браузера и сервера в загрузке и отображении веб-страницы.

Терминология: Клиент, сервер, HTTP, URL, веб-сервер, веб-браузер.

Тема 1.4. Установка и настройка редактора кода (Notepad++).

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Установка и настройка редактора кода, знакомство с его основными возможностями.

Терминология: Редактор кода, IDE, синтаксическое выделение, автодополнение.

Тема 1.5. Знакомство с браузерами и инструментами разработчика.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Изучение различных веб-браузеров, использование инструментов разработчика для анализа и отладки веб-страниц.

Терминология: Веб-браузер, инструменты разработчика, инспектор элементов, консоль, сеть, источники.

Раздел 2. Основы HTML

Тема 2.1. Введение в HTML: структура HTML-документа (doctype, html, head, body).

Форма проведения занятия: Лекция.

Виды учебной деятельности: Изучение основной структуры HTML-документа, назначение основных тегов.

Терминология: HTML, DOCTYPE, тег, элемент, атрибут, head, body.

Тема 2.2. Основные HTML-теги: заголовки, абзацы, списки, ссылки и изображения.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Создание простых HTML-страниц с использованием основных тегов.

Терминология: Заголовки, абзацы, списки, ссылки, изображения, теги h1-h6, p, ul/ol/li, a, img.

Тема 2.3. Формы: создание простых форм с использованием различных элементов.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Создание HTML-форм с использованием различных элементов ввода.

Терминология: Формы, input, textarea, checkbox, radio, select, button, label.

Тема 2.4. Семантические теги: использование <header>, <footer>, <article>, <section>.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Применение семантических тегов для структурирования веб-страницы.

Терминология: Семантические теги, header, footer, article, section, nav, aside.

Тема 2.5. Встраивание медиафайлов: работа с аудио и видео.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Добавление аудио и видео на веб-страницу с использованием тегов <audio> и <video>.

Терминология: Аудио, видео, медиафайлы, кодеки, атрибуты src, controls, autoplay.

Раздел 3. Основы CSS

Тема 3.1. Введение в CSS: подключение стилей к HTML-документу.

Форма проведения занятия: Лекция.

Виды учебной деятельности: Изучение способов подключения CSS к HTML-документу, внутренние и внешние стили.

Терминология: CSS, каскадные таблицы стилей, селекторы, свойства, значения.

Тема 3.2. Селекторы и свойства: базовые селекторы и их применение.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Использование различных типов селекторов для стилизации HTML-элементов.

Терминология: Селекторы, свойства, значения, тип, класс, id, универсальный.

Тема 3.3. Блочная модель: margin, padding, border.

Форма проведения занятия: Практическое занятие

Виды учебной деятельности: Изучение блочной модели CSS, работа с margin, padding и border.

Терминология: Блочная модель, margin, padding, border, ширина, высота.

Тема 3.4. Стилизация текста и использование шрифтов.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Применение различных CSS-свойств для стилизации текста, подключение внешних шрифтов.

Терминология: Шрифт, семейство шрифтов, размер, цвет, выравнивание, декорирование.

Тема 3.5. Flexbox: основные принципы и применение.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Изучение основ Flexbox, создание макетов с использованием Flexbox.

Терминология: Flexbox, flex-container, flex-item, flex-direction, justify-content, align-items.

Раздел 4. Основы JavaScript

Тема 4.1. Введение в JavaScript: назначение и история.

Форма проведения занятия: Беседа.

Виды учебной деятельности: Обсуждение роли JavaScript в веб-разработке, краткая история языка.

Терминология: JavaScript, клиентский язык, интерпретируемый язык, ECMAScript.

Тема 4.2. Подключение JavaScript к HTML: встроенный и внешний код.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Изучение способов подключения JavaScript к HTML-документу.

Терминология: Встроенный JavaScript, внешний JavaScript, тег script, атрибут src.

Тема 4.3. Типы данных в JavaScript: строки, числа, булевы значения.

Форма проведения занятия: Лекция.

Виды учебной деятельности: Знакомство с основными типами данных в JavaScript, их особенности.

Терминология: Типы данных, строки, числа, булевы значения, undefined, null.

Тема 4.4. Переменные: var, let, const.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Объявление и использование переменных в JavaScript.

Терминология: Переменные, var, let, const, область видимости.

Тема 4.5. Условия: операторы if, else, switch.

Форма проведения занятия: Практическое занятие

Виды учебной деятельности: Изучение условных операторов if, else, switch для управления логикой программы.

Терминология: Условные операторы, if, else, switch, сравнение, логические операторы.

Тема 4.6. Циклы: for, while, do while.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Знакомство с различными типами циклов в JavaScript.

Терминология: Циклы, for, while, do while, итерация, счетчик.

Тема 4.7. Функции: объявление и вызов.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Изучение объявления и использования функций в JavaScript.

Терминология: Функции, объявление функций, вызов функций, аргументы, возвращаемое значение.

Тема 4.8. Объекты и массивы: создание и использование.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Создание и работа с объектами и массивами в JavaScript.

Терминология: Объекты, свойства, методы, массивы, индексы, элементы.

Тема 4.9. Методы и свойства массивов.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Изучение встроенных методов и свойств массивов в JavaScript.

Терминология: Методы массивов, push, pop, slice, splice, forEach, map, filter, reduce.

Тема 4.10. Работа с событиями: обработка кликов, наведения и других событий.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Создание обработчиков событий для взаимодействия с пользователем.

Терминология: События, обработчики событий, click, mouseover, keydown, addEventListener.

Раздел 5. DOM и взаимодействие с веб-страницей

Тема 5.1. Что такое DOM? Введение.

Форма проведения занятия: Лекция.

Виды учебной деятельности: Изучение понятия DOM, его структуры и взаимодействия с JavaScript.

Терминология: DOM, Document Object Model, дерево DOM, узлы, элементы.

Тема 5.2. Динамическое изменение HTML с помощью JavaScript.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Использование JavaScript для создания, изменения и удаления HTML-элементов.

Терминология: innerHTML, textContent, createElement, appendChild, removeChild.

Тема 5.3. Поиск элементов в DOM: методы getElementById, querySelector.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Поиск и выборка элементов на странице с помощью различных методов.

Терминология: getElementById, querySelector, querySelectorAll, поиск по id, классу, тегу.

Тема 5.4. Изменение CSS-свойств с помощью JavaScript.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Динамическое изменение стилей элементов с помощью JavaScript.

Терминология: style, CSS-свойства, значения, classList, add, remove, toggle.

Тема 5.5. Создание и удаление элементов на странице.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Использование методов для добавления и удаления элементов на веб-странице.

Терминология: createElement, appendChild, removeChild, insertBefore.

Тема 5.6. Обработка событий: addEventListener.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Изучение использования addEventListener для обработки событий на странице.

Терминология: События, обработчики событий, addEventListener, callback-функции.

Тема 5.7. Формы и JavaScript: обработка данных формы.

Форма проведения занятия: Практическое занятие

Виды учебной деятельности: Получение и обработка данных, введенных пользователем в форму.

Терминология: Формы, input, value, submit, preventDefault.

Тема 5.8. Валидация форм с помощью JavaScript.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Реализация валидации форм на клиентской стороне с использованием JavaScript.

Терминология: Валидация, required, pattern, minlength, maxlength, type.

Раздел 6. Введение в адаптивный дизайн

Тема 6.1. Что такое адаптивный дизайн? Зачем он нужен?

Форма проведения занятия: Беседа.

Виды учебной деятельности: Обсуждение понятия адаптивного дизайна, его важности и преимуществ.

Терминология: Адаптивный дизайн, мобильные устройства, viewport, разрешение экрана.

Тема 6.2. Медиа-запросы: использование и примеры.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Изучение и применение медиа-запросов для создания адаптивного дизайна.

Терминология: Медиа-запросы, min-width, max-width, orientation, device-pixel-ratio.

Тема 6.3. Основы мобильного дизайна.

Форма проведения занятия: Лекция.

Виды учебной деятельности: Рассмотрение особенностей мобильного дизайна, оптимизация контента и навигации.

Терминология: Мобильный дизайн, touch-события, жесты, навигация, компактность.

Тема 6.4. Сетки и фреймы для адаптивного дизайна.

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Создание адаптивных макетов с использованием сеточных систем.

Терминология: Сетки, фреймы, колонки, ряды, гибкость, responsive.

Тема 6.5. Использование фреймворков: Bootstrap и его возможности

Форма проведения занятия: Лекция, практическое занятие.

Виды учебной деятельности: Изучение основ фреймворка Bootstrap, создание макета с использованием Bootstrap, работа с компонентами и сеткой Bootstrap

Терминология: Bootstrap, фреймворк, компоненты, сетка, классы

Раздел 7. Основы работы с API

Тема 7.1. Что такое API? Введение и примеры

Форма проведения занятия: Лекция.

Виды учебной деятельности: Прослушивание лекции, изучение примеров API, обсуждение случаев использования API в реальных проектах.

Терминология: API (Application Programming Interface), REST, JSON, эндпоинт, запрос.

Тема 7.2. Получение данных с помощью Fetch API

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Написание кода для получения данных с использованием Fetch API, обработка ответов.

Терминология: Fetch API, промисы (Promises), асинхронные операции, запросы GET

Тема 7.3. Обработка JSON-данных

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Работа с JSON-данными, преобразование данных, создание и парсинг JSON.

Терминология: JSON (JavaScript Object Notation), сериализация, десериализация, объект.

Тема 7.4. Отправка данных на сервер: метод POST

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Написание кода для отправки данных на сервер с использованием метода POST, анализ серверных ответов.

Терминология: POST-запрос, тело запроса, серверный ответ, форма данных.

Раздел 8. Создание простого проекта

Тема 8.1. Проект: создание личной страницы

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Разработка личной страницы, планирование структуры, создание макета.

Терминология: Личная страница, HTML, CSS, структура документа.

Тема 8.2. Дизайн и макет страницы

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Создание и стилизация макета страницы, работа с визуальными элементами и типографикой.

Терминология: Макет, типографика, визуальный дизайн, прототипирование.

Тема 8.3. Реализация адаптивности

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Применение медиа-запросов для создания адаптивного дизайна, тестирование на различных устройствах.

Терминология: Адаптивность, медиа-запросы, точки перелома.

Тема 8.4. Интерактивные элементы: форма обратной связи

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Создание и стилизация формы обратной связи, работа с валидацией и обработкой данных.

Терминология: Форма обратной связи, валидация, интерактивные элементы, события.

Тема 8.5. Обработка данных и их валидация

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Реализация валидации формы, обработка пользовательского ввода, написание кода для проверки данных.

Терминология: Валидация данных, обработка формы, регулярные выражения.

Раздел 9. Расширенные темы

Тема 9.1. Основы TypeScript: введение Форма проведения занятия: Лекция и практическое занятие
Виды учебной деятельности: Изучение основ TypeScript, написание простых программ, работа с типами данных
Терминология: TypeScript, типизация, компиляция, интерфейсы

Тема 9.2. Работа с библиотеками и фреймворками

Форма проведения занятия: Лекция и практическое занятие.

Виды учебной деятельности: Изучение основ работы с библиотеками и фреймворками, применение jQuery в проектах.

Терминология: Библиотеки, фреймворки, jQuery, DOM-манипуляция, AJAX.

Тема 9.3. Введение в системы контроля версий: Git

Форма проведения занятия: Лекция, практическое занятие.

Виды учебной деятельности: Изучение основ Git, выполнение операций по контролю версий, создание репозитория.

Терминология: Git, версия, репозиторий, коммит, ветвление.

Тема 9.4. Развертывание сайта: хостинг и домены

Форма проведения занятия: Лекция и практическое занятие.

Виды учебной деятельности: Изучение процессов развертывания сайта, работа с хостингом и доменами, настройка DNS.

Терминология: Хостинг, домен, DNS, FTP, развертывание.

Тема 9.5. Оптимизация производительности веб-сайтов

Форма проведения занятия: Лекция, практическое занятие.

Виды учебной деятельности: Анализ и применение методов оптимизации веб-сайтов, работа с инструментами анализа производительности.

Терминология: Оптимизация, производительность, кэширование, минимизация ресурсов.

Раздел 10. Проектная деятельность

Тема 10.1. Обсуждение идей для итогового проекта

Форма проведения занятия: Беседа, мозговой штурм.

Виды учебной деятельности: Обсуждение и выбор идей для итогового проекта, формирование групп и распределение задач.

Терминология: Идея проекта, планирование, мозговой штурм, концепция.

Тема 10.2. Планирование и экранные концепты

Форма проведения занятия: Практическое занятие.

Виды учебной деятельности: Создание и обсуждение экранных концептов, планирование функциональности и дизайна проекта.

Терминология: Экранные концепты, прототипирование, пользовательский интерфейс (UI), пользовательский опыт (UX).

Тема 10.3. Реализация проекта с использованием всех изученных технологий

Форма проведения занятия: Практическое занятие

Виды учебной деятельности: Разработка итогового проекта с применением всех изученных технологий, работа в группах

Терминология: Реализация проекта, интеграция технологий, тестирование

Тема 10.4. Презентация итогового проекта

Форма проведения занятия: Защита проектов.

Виды учебной деятельности: Презентация выполненных проектов, обсуждение и защита решений, получение отзывов.

Терминология: Презентация, защита проекта, обратная связь.

4. Контрольно-оценочные средства

Базовый уровень

Входной контроль – начальная диагностика в форме опроса «Основы алгоритмики» с целью определения уровня по изучаемой программе (Приложение 1). Анализ результатов входного контроля проводится с помощью следующей шкалы:

<i>Оценка параметров</i>	<i>Уровень по сумме баллов</i>
Начальный уровень	7-14 баллов
Средний уровень	15-21 баллов
Высокий уровень	22-31 баллов

Текущий контроль (оценка усвоения изучаемого материала) осуществляется педагогом на протяжении всей программы в форме педагогического наблюдения, анализа опросов, выполненных практических заданий.

Итоговый контроль осуществляется в виде оценки продукта обучающегося, разработанного в рамках выполнения итогового проекта сайта, написанного на HTML и CSS с применением ранее полученных знаний и навыков.

Углубленный уровень

Входной контроль – начальная диагностика в форме опроса «Основы веб-программирования» с целью определения уровня по изучаемой программе (Приложение 2). Анализ результатов входного контроля проводится с помощью следующей шкалы:

<i>Оценка параметров</i>	<i>Уровень по сумме баллов</i>
Начальный уровень	10-14 баллов
Средний уровень	15-20 баллов

Высокий уровень	21-26 баллов
-----------------	--------------

Текущий контроль (оценка усвоения изучаемого материала) осуществляется педагогом на протяжении всей программы в форме педагогического наблюдения, анализа опросов, выполненных практических заданий.

Итоговый контроль осуществляется в виде оценки продукта обучающегося, разработанного в рамках выполнения итогового проекта функционирующего сайта с обработкой данных на сервере с применением ранее полученных знаний и навыков.

5. Условия реализации программы

Раздел программы	Учебно-методическое обеспечение программы	Материально-техническое обеспечение программы	Информационные образовательные ресурсы
1. Введение в веб-разработку	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Кто создает сайты и как?»	Ноутбуки с необходимым установленным программным обеспечением (Браузер, Notepad++) и компьютерные мыши для каждого обучающегося и педагога, интерактивная панель.	Начало работы с Web: https://developer.mozilla.org/ru/docs/Learn
2. Основы HTML	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Слова языка HTML: тэги» «Создаем первую страничку», «Какими были первые сайты?».		Понятия и применения тегов HTML: https://html5book.ru/osnovy-html/
3. Основы CSS	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Дизайн и стиль сайта», «Обработка элементов: цвет, форма», «Быстрый коддинг на CSS», «При чём тут коробки?»		Базовый CSS: https://webref.ru/course/css-basics
4. Основы JavaScript	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Язык прямо в браузере», «Переменные и типы данных JavaScript», «Интеграция JavaScript в сайт», «Пишем скрипты»		Современный учебник JS: https://learn.javascript.ru/first-steps
5. DOM и взаимодействие с веб-страницей	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы:		DOM-модель документа: https://ipc.susu.ru/41088.html

	«Делаем живой сайт»		
6. Введение в адаптивный дизайн	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Макеты сайтов на разных устройствах»		Адаптивный дизайн: https://metanit.com/web/html5/11.1.php
7. Основы работы с API	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Как подключить API к своему проекту»		Введение в web APIs
8. Создание простого проекта	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Делаем сайт под заказ»		Памятка про сайты на заказ: https://clck.ru/3DJxR
9. Расширенные темы	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Продвинутый TypeScript»		Руководство по TypeScript: https://metanit.com/web/typescript/
10. Проектная деятельность	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Подготовка проекта», «Основы успешной презентации»		Разработка сайта до самого конца: https://clck.ru/3DJz3d

Кадровое обеспечение – педагог дополнительного образования.

Список литературы

Нормативные правовые документы:

1. Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации (с изменениями и дополнениями, вступившими в силу с 01.01.2021)
2. Концепция развития дополнительного образования детей до 2030 года (утверждена распоряжением Правительства РФ от 31 марта 2022 г. № 678-р)
3. Приказ Министерства просвещения Российской Федерации от 27.07.2022 № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».
4. Постановление Главного государственного санитарного врача РФ от 28.09.2020 №28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи.

Учебно-методическая литература для педагога:

5. Бандура, Г.А. Основы проектной и исследовательской деятельности. Практикум / Г.А. Бандура, Ж.В. Морозова, Н.В. Пушкина. – Санкт-Петербург: «Лань», 2021. – 152 с.
6. Флэнаган Д. JavaScript. Полное руководство. 7-е издание. – М.: Издательство «Диалектика-Вильямс». – Питер, 2021.
7. Хавербеке М. Выразительный JavaScript. Современное веб-программирование. 3-е издание. – Питер, 2023.
- 8.

Литература для обучающихся и родителей:

9. Смирнов, А. А. Введение в HTML и CSS. — М.: Эксмо, 2019.
10. Тихомиров, С. И. JavaScript: от начинающего до профессионала. — М.: Урал, 2023.
11. Карасев, А. И. Современные технологии веб-разработки. — М.: Курс, 2023.

Электронные образовательные ресурсы:

12. Современный учебник JavaScript / [Электронный ресурс] // JSR: [сайт]. Режим доступа: <https://learn.javascript.ru/> (дата обращения: 10.08.2024).

13. Руководство по HTML и CSS / [Электронный ресурс] // MDN Web Docs: [сайт]. Режим доступа: https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/HTML_basics (дата обращения: 04.08.2024).

14. Справочник CSS/ [Электронный ресурс] // WebReference: [сайт]. Режим доступа: <https://webref.ru/css> (дата обращения: 10.08.2024).

Входной тест по теме: «Основы алгоритмики».

1. Алгоритм - это
 - а) правила выполнения определенных действий;
 - б) предписание исполнителю совершить последовательность действий, направленных на достижение поставленных целей;
 - в) набор команд для компьютера.
2. Какой из документов является алгоритмом?
 - а) Правила техники безопасности.
 - б) Инструкция по получению денег в банкомате.
 - в) Расписание уроков.
3. Какой из объектов может являться исполнителем?
 - а) Луна. б) Карта. в) Принтер. г) Книга
4. Дискретность – свойство алгоритма, означающее...
 - а) однозначность правил выполнения алгоритма
 - б) правильность результатов выполнения алгоритма
 - в) деление алгоритма на отдельные шаги
5. Свойством алгоритма является:
 - а) конечность;
 - б) цикличность;
 - в) возможность изменения последовательности команд;
 - г) возможность выполнения алгоритма в обратном порядке.
6. Алгоритм называется линейным, если:
 - а) он составлен так, что его выполнение предполагает многократное повторение одних и тех же действий;
 - б) ход его выполнения зависит от истинности тех или иных условий;
 - в) его команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий.
7. Алгоритм структуры «ветвление» предусматривает
 - а) выбор условий, б) выбор алгоритмов, в) выбор команд (действий)
8. Алгоритм называется циклическим, если:
 - а) он составлен так, что его выполнение предполагает многократное повторение одних и тех же действий;
 - б) ход его выполнения зависит от истинности тех или иных условий;

в) его команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий.

9. Алгоритм называется вспомогательным, если

а) он предполагает выбор действий

б) повторяет действия до выполнения какого – либо условия;

в) решает часть задачи и вызывается из основной программы.

10. Цикл со счётчиком

а) зависит от некоторого условия; б) зависит от известного числа повторений.

11. Какой тип алгоритмической структуры необходимо применить, если последовательность команд выполняется или не выполняется в зависимости от условия

а) цикл б) ветвление в) линейный.

12. Ромб — графический объект, используемый в блок-схеме для записи:

а) ввода, вывода данных; б) вычислительных действий;

в) конца выполнения задачи; г) условия выполнения действий

13. Когда необходимо составлять блок-схему программы?

а) До начала составления самой программы

б) После составления программы

в) В процессе составления программы

14. Наиболее наглядной формой описания алгоритма является структурно-стилизированный метод

а) представление алгоритма в виде схемы

б) язык программирования высокого уровня

в) словесное описание алгоритма

15. Перевод программ с языка высокого уровня на язык более низкого уровня обеспечивает программа -

а) ассемблер

б) паскаль

в) компилятор

16. Разработка алгоритма решения задачи – это

а) сведение задачи к математической модели, для которой известен метод решения

- б) точное описание данных, условий задачи и ее целого решения
- в) определение последовательности действий, ведущих к получению результатов

17. Разработке алгоритма предшествует

- а) постановка задачи, разработка математической модели, проектирование программ
- б) постановка задачи, разработка математической модели
- в) постановка задачи, выбор метода решения, проектирование программ

18. Раздел переменных определяется служебным словом:

а) LABEL

б) TYPE

в) VAR

19. Для вычисления квадратного корня применяется функция:

а) SQR(X)

б) SQRT(X)

в) ORD(X)

20. Укажите неправильно записанный оператор:

а) if a<b then a:=a*a else b:=b*b;

б) if x and y then s:=s+1; else s:=s-1;

в) if k<>m then k:=m;

21. В Паскале различие в высоте букв (прописные или строчные)

а) имеет значение при написании текстовых констант

б) имеет значение для написания служебных слов

в) имеет значение для написания имен переменных

22. Какое выражение неверно?

а) Структура подпрограммы аналогична структуре основной программы

б) Подпрограмма предназначена для выполнения какой-то последовательности

действий

в) В Паскале имеется две разновидности подпрограмм – процедуры и функции

23. Каким символом разделяются перечень разнотипных формальных параметров в процедуре?

а) запятой

б) точкой с запятой

в) двоеточием

24. Из предложенных заголовков программы правильным является:

а) Program сумма;

б) programm сумма;

в) programm summa

25. Конечное значение переменной У в результате выполнения следующих действий:

$Y:=5; X:=2; Y:=(Y-2*X)/2+X$

будет равно:

а) 0,5;

б) 2;

в) 2,5; г) 5.

26. Примером разветвленного алгоритма является:

а) переход улицы по сигналу светофора

б) жизнь растения

в) заваривание чая

27. К свойствам алгоритма относится?

а)результативность, дискретность, массовость

б)дискретность, массовость, абсолютность

в)массовость, дискретность, результативность, детерминированность

28. Что не является способом записи алгоритма?

а)Словесное описание

б) Программа

в)Блок-схема

29. Алгоритмическая структура какого типа записана ниже?

IF Условие THEN

Серия 1

ELSE

Серия 2

END IF

а)Циклическая

б)Ветвление

в)Линейная

30. Вставьте пропущенное слово, выбрав его из списка. Блок-схема - это форма записи алгоритма, при которой для обозначения различных шагов алгоритма используются....

а)формулы

б)списки

в)геометрические фигуры

Входной тест по направлению «Web-разработка. Углубленный уровень»

1. Web-страница (документ HTML) представляет собой:
 - a. Текстовый файл с расширением txt или doc
 - b. Текстовый файл с расширением htm или html
 - c. Двоичный файл с расширением com или exe
 - d. Графический файл с расширением gif или jpg
2. Для вставки изображения в документ HTML используется команда:
 - a. ``
 - b. `<body background="ris.jpg">`
 - c. ``
 - d. `<input="ris.jpg">`
3. Гипертекст - это:
 - a. Текст очень большого размера
 - b. Текст, в котором используется шрифт большого размера
 - c. Структурированный текст, где возможны переходы по выделенным меткам
 - d. Текст, в который вставлены объекты с большим объемом информации
4. Чему равно по умолчанию значение параметра SIZE тега ?
 - a. 6
 - b. 3
 - c. 4
 - d. 5
5. Каким тегом объявляется web-страница?
 - a. `<html> </html>`
 - b. `<head> </head>`
 - c. `<title> </title>`
 - d. `<body> </body>`.
6. Какой код для пустой web-страницы правильный?
 - a. `<html> <head> <title> </head> <body> </body> </html>`
 - b. `<html> <head> <title> </title> </head> <body> </body> </html>`
 - c. `<html> <head> <title> <body> </body> </html>`
 - d. `<html> <head> <title> </title> </head> <body> </body>`
7. Каким тегом задается вставка гиперссылки на web-страницу?
 - a. ` `
 - b. ``
 - c. ` `
 - d. ``
8. Инструкция браузеру, указывающая способ отображения текста:
 - a. Программный код
 - b. Тэг
 - c. Файл
 - d. Гиперссылка
9. Программа для просмотра гипертекстовых страниц называется:

- a. Сервер
- b. Протокол
- c. HTML
- d. Браузер

10. Какие тэги задают размер заголовка?

- a. <p></p>
- b.
- c. <body></body>
- d. <h1></h1>

11. Назовите атрибут обязательный для тега :

- a. src
- b. with
- c. title
- d. href

12. Тег <I> позволяет задать:

- a. курсив
- b. подчеркнутый шрифт
- c. моноширинный шрифт
- d. полужирный шрифт

13. Заполните фразу: PHP (от англ. _____) это _____ язык программирования, основанный на технологии _____

- a. "PHP: Hypertext Preprocessor", "скриптовый язык", "Zend"
- b. "PHP: Hypertext Preprocessor", "скриптовый язык", "PHP"
- c. "Personal Hypertext Processor", "скриптовый язык", "Zend"
- d. "Personal Hypertext Processor", "скриптовый язык", "PHP"

14. Почему в PHP файлах не рекомендуется ставить ">" закрывающийся тег?

- a. Для того чтобы предотвратить случайное включение пробелов или переносов на новую строку
- b. Подобная рекомендация улучшает читабельность
- c. Это важный аспект безопасности
- d. Не рекомендуется? Не, это не правда

15. Что будет выведено в результате выполнения следующего кода?

```
<?php
echo 'Hello';
if (false)
    echo "PHP "; echo " Certification";

echo "!!!";
```

- a. Hello Certification!!!
- b. Возникнет ошибка
- c. Hello PHP Certification!!!
- d. HelloPHP Certification!!!
- e. Hello Certification!!!

16. Дан программный код. Что будет выведено после запуска скрипта?

```
<?php
for ($i = 0; $i < 5; $i++) {
    if ($i % 2 == 0) continue;
    echo $i;
}
?>
```

- a. 024
- b. 24
- c. 13
- d. 013

17. Есть массив \$arr. Как вывести все элементы массива \$arr?

- a. print_r(\$arr);
- b. print \$arr;
- c. echo \$arr;
- d. это возможно сделать только через цикл.

18. Следующий фрагмент кода создает список:

```
<ul>
<li>элемент 1</li>
<li>элемент 2</li>
<li>элемент 3</li>
</ul>
```

Что верно о созданном списке ?

- a. будет создан неупорядоченный список
- b. будет создан упорядоченный список
- c. каждый элемент списка будет начинаться с номера по порядку
- d. каждый элемент списка будет начинаться с маркера

19. Что выведет следующий код?

```
$arr = array(1,3,5);
$count = count($arr);
if ($count = 0) {
    echo "Массив пуст.";
} else {
    echo "Массив состоит из $count элементов.";
}
```

- a. Массив состоит из 3 элементов.
- b. Массив пуст.
- c. Массив состоит из 0 элементов.

20. Что такое сложный запрос MySQL?

- a. это запрос, в котором участвует одна таблица базы данных
- b. это один запрос к нескольким таблицам
- c. это база данных сайта
- d. это обращение к базе данных

21. Что делает данный фрагмент кода?

```
if ($a < $b) {echo 'OK';}
```

- a. если \$a меньше чем \$b выводит "OK"
- b. если \$a меньше или равно \$b выводит "OK"

- c. если \$a\$ больше чем \$b\$ выводит "OK"
 - d. если \$a\$ больше или равно \$b\$ выводит "OK"
22. Горизонтальная линия шириной 50%, толщиной 2 пикселя, нерельефная, синего цвета обозначается
- a. `<hr size=2 width=50% color=blue>`
 - b. `<hr size=50% width=2 color=blue noshade>`
 - c. `<hr size=2 width=50% color=blue noshade>`
 - d. `<hr size=50% width=2 color=blue >` строкой
23. Выберите четверку правильно написанных параметров выравнивания:
- a. justify,centre,right,left
 - b. center,left,riht,justify
 - c. right,centre,justify,left
 - d. left,center,justify,right
24. Какого вида URL адреса применяемых в гиперссылки не существует
- a. Относительного
 - b. Абсолютного
 - c. Полного
25. Тег который рисует линии
- a. `<TH>...</TH>`
 - b. `<Href>...</Href>`
 - c. `<HR>...</HR>`
26. Каким тегом определяется нумерованный список
- a. `...`
 - b. `...`
 - c. `...`