Управление образования администрации

Татарского муниципального округа

муниципальное бюджетное общеобразовательное

учреждение – лицей г. Татарска

структурное подразделения - Центр цифрового образования детей «IT-куб»

РАССМОТРЕНО
на педагогическом совете
МБОУ – ЛИЦЕЯ
протокол № 1 от27.08.2026 г.

УТВЕРЖДАЮ
Директор
МБОУ – ЛИЦЕЯ
_____ Т.А. Чернова
приказ №62/1/ОД от 27.08.2026 г.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

технической направленности

«Введение в веб-разработку»

Возраст обучающихся: 11-14 лет

Срок реализации программы: 1 год

Автор-составитель программы:
Моисеенко Никита Сергеевич,
педагог дополнительного образования

Татарск, 2026

Раздел 1. Комплекс основных характеристик программы

1.1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Введение в веб-разработку» имеет техническую направленность.

Актуальность программы

Актуальность программы «Введение в веб-разработку» заключается в том, что веб-технологии являются основой современного интернета, и благодаря этому освоение базовых языков разметки и стилей (HTML, CSS) открывает перед обучающимися широкие возможности для создания собственных проектов — от визиток до блогов и мини-игр. Актуальность программы обусловлена тем, что в настоящее время работодатели и вузы всё чаще обращают внимание на наличие цифрового портфолио у абитуриентов, а школьные уроки информатики не дают практических навыков вёрстки и веб-дизайна. К числу наиболее актуальных проблем относится дефицит доступных образовательных программ, которые сочетали бы основы веб-разработки с проектным подходом и позволяли бы подросткам создавать реально работающие веб-страницы уже в процессе обучения.

Отличительные особенности программы

Характерные свойства, отличающие программу от других аналогичных, связаны с принципом «от идеи к готовому продукту»: обучающиеся не просто изучают теги и свойства, а сразу применяют их для создания реальных проектов — личного блога, портфолио, сайта для школьного мероприятия. Отличительные черты программы: включение элементов веб-дизайна (цвет, шрифты, композиция) в технические занятия, что развивает творческое мышление и чувство стиля; наличие «чек-листов качества» для самооценки работ перед сдачей проекта. Основные идеи, которые придают программе своеобразие: каждый раздел завершается созданием законченного веб-ресурса, который можно показать друзьям и родителям, что значительно повышает мотивацию и даёт ощущение значимости своей работы.

Целевая группа

Программа предназначена для обучающихся 11–14 лет, проявляющих интерес к информационным технологиям, интернету и созданию сайтов. Половозрастной состав группы смешанный. Программа доступна для обучающихся с ОВЗ с уточнением нозологии заболевания.

Потребности и интересы: стремление к творческому самовыражению через цифровые инструменты, желание научиться создавать собственные веб-страницы и блоги, интерес к современным технологиям.

Ожидаемые эффекты: освоение базовых навыков вёрстки (HTML, CSS), развитие творческого и логического мышления, формирование цифровой грамотности.

Объём и срок освоения программы

Объём программы — 72 часа в год.

Программа рассчитана на 1 год обучения.

Форма обучения: очная.

Уровень программы – базовый.

Особенности организации образовательного процесса

Общая оценка соответствия планируемого содержания программы имеющимся в ОО условиям ее реализации является положительной.

Программа обеспечена необходимыми кадровыми ресурсами: педагог имеет профильное образование и практические навыки фронтенд и бекенд веб-разработки.

Материально-техническая база включает учебный класс на 10–12 мест с доступом к сети Интернет, оснащенный необходимым ПО (Visual Studio Code, Notepad++) и мультимедийным оборудованием. Организационно занятия проводятся во второй половине дня по расписанию, в очной форме, с возможностью применения дистанционных технологий. Набор в группы осуществляется на добровольной основе.

Имеющиеся условия полностью соответствуют целям и задачам программы и позволяют достичь планируемых результатов в полном объеме.

Режим занятий, периодичность и продолжительность занятий

Занятия проводятся два раза в неделю продолжительностью 1 академический час(1 академический час равен 45 минутам). Перерыв между учебными занятиями 10-15 минут.

Количество обучающихся в одной группе: 10-12 человек.

1.2. Цель и задачи программы

Цель: формирование базовых навыков веб-разработки и творческого мышления у обучающихся через создание собственных веб-страниц и сайтов.

Задачи программы:

Обучающие: сформировать знания основ языка разметки HTML (структура документа, основные теги, атрибуты); обучить основам каскадных таблиц стилей CSS (оформление текста, цвета, размеры, позиционирование); познакомить с базовыми принципами работы с графикой и цветом в веб-дизайне; сформировать навыки создания статических веб-страниц с использованием HTML и CSS; познакомить с основами адаптивной вёрстки (Flexbox/Grid, медиазапросы) и работой с браузерными инструментами разработчика; обучить базовым элементам интерактивности с использованием JavaScript (обработка событий, простые анимации); сформировать навыки работы в текстовом редакторе кода и публикации сайтов в интернете.

Развивающие: развивать алгоритмическое и логическое мышление через решение практических и олимпиадных задач; формировать навыки самостоятельной работы с технической документацией и поиска информации в сети Интернет; формировать навыки командной работы, распределения ролей и коллективной разработки проектов.

Воспитательные: формировать ответственное отношение к результатам своего труда и качеству выполняемой работы; воспитывать культуру оформления

программного кода и соблюдение стандартов разработки; способствовать развитию коммуникативных навыков через участие в обсуждении решений и публичной защите проектов; воспитывать навыки здорового образа жизни, соблюдение санитарно-гигиенических норм при работе за компьютером; ориентировать обучающихся на осознанный выбор будущей профессии в сфере информационных технологий.

1.3. Содержание программы

Учебно-тематический план

№ п/п	Название раздела	Количество часов			Формы промежуточной аттестации/ контроля
		Всего	Из них		
			Теория	Практика	
1	Введение в веб-разработку. Инструменты.	4	2	2	Практическая работа
2	Основы HTML.	14	6	8	Тестирование, самостоятельная работа
3	Основы CSS.	16	6	10	Практическая работа, самостоятельная работа
4	Позиционирование и адаптивная вёрстка (Flexbox/Grid).	10	4	6	Практическая работа
5	Основы веб-дизайна и работа с графикой.	6	3	3	Самостоятельная работа
6	Введение в JavaScript. Интерактивность.	10	4	6	Практическая работа, тестирование
7	Итоговый проект.	12	2	10	Защита проекта
	Итого:	**72**	**27**	**45**	

Содержание программы

1. Введение в веб-разработку. Инструменты.

Теория: Техника безопасности и правила поведения в компьютерном классе. Как устроен интернет: клиент, сервер, браузер. Что такое веб-разработка. Знакомство с текстовыми редакторами кода (Visual Studio Code). Структура HTML-документа. Основные понятия: тег, атрибут, элемент. Браузерные инструменты разработчика (DevTools): открытие, просмотр кода страницы.

Практика: Установка и настройка Visual Studio Code (тема, шрифт, расширения). Создание первой HTML-страницы с заголовком и абзацем. Открытие страницы в браузере. Использование DevTools для просмотра

структуры страницы.

2. Основы HTML.

Теория: Структура HTML-документа: <!DOCTYPE html>, <html>, <head>, <body>. Заголовки (<h1>–<h6>). Абзацы (<p>). Списки: маркированные (<ul>) и нумерованные (<ol>). Ссылки (<a>): атрибуты href, target. Изображения (<img>): атрибуты src, alt, width, height. Таблицы (<table>, <tr>, <td>, <th>). Формы (<form>, <input>, <textarea>, <select>, <button>). Семантические теги HTML5 (<header>, <nav>, <main>, <section>, <article>, <footer>). Комментарии в HTML.

Практика: Создание страницы-визитки с заголовками, абзацами и списками. Добавление ссылок и изображений. Создание таблицы с данными. Разработка формы обратной связи (поля ввода, кнопка отправки). Создание структуры сайта с использованием семантических тегов (шапка, навигация, основное содержание, подвал). Сборка простого сайта из нескольких HTML-файлов с перелинковкой.

3. Основы CSS.

Теория: Способы подключения стилей: встроенные (style), внутренние (<style>), внешние (.css). Базовые селекторы: по тегу, по классу (.), по идентификатору (#). Свойства: цвет (color), фон (background), шрифты (font-family, font-size, font-weight, text-align), отступы (margin, padding), рамки (border). Блочная модель: width, height, box-sizing. Каскадность и наследование. Псевдоклассы (:hover, :active, :visited). Псевдоэлементы (::before, ::after). Группировка селекторов. Приоритеты в CSS.

Практика: Подключение внешнего CSS-файла к HTML-странице. Оформление текста: выбор шрифтов, размеров, цветов. Оформление ссылок (изменение цвета при наведении). Стилизация списков и таблиц. Оформление формы (поля ввода, кнопка). Создание меню навигации с использованием псевдоклассов. Стилизация всех страниц сайта в едином стиле. Создание цветовой схемы сайта.

4. Позиционирование и адаптивная вёрстка.

Теория: Позиционирование элементов: static, relative, absolute, fixed, sticky. Введение в Flexbox: основные свойства контейнера (display: flex, flex-direction, justify-content, align-items, flex-wrap) и элементов (flex-grow, flex-shrink, flex-basis). Введение в CSS Grid: основные свойства (grid-template-columns, grid-template-rows, gap). Медиазапросы (@media). Принципы адаптивной вёрстки (mobile-first, десктоп-версия). Браузерные инструменты для проверки адаптивности (Device Toolbar).

Практика: Создание гибкого меню с использованием Flexbox. Верстка карточек товаров с выравниванием. Построение макета сайта с использованием Grid. Адаптация страницы для просмотра на мобильных устройствах (изменение расположения блоков). Создание "резинового" макета. Тестирование адаптивности в DevTools.

5. Основы веб-дизайна и работа с графикой.

Теория: Основы цветоведения: цветовые схемы, сочетаемость цветов, контраст. Типографика в вебе: выбор шрифтов, размеры, межстрочный

интервал, иерархия текста. Композиция: сетки, выравнивание, визуальный баланс. Форматы изображений для веба: PNG, JPEG, GIF, SVG, WebP. Оптимизация изображений (размер, качество). Понятие favicon. Работа с графическими редакторами для создания макетов (обзор возможностей).

Практика: Подбор цветовой схемы для своего сайта. Выбор шрифтов с использованием бесплатных библиотек (подключение через ссылки). Разработка макета страницы в графическом редакторе (или на бумаге). Добавление и оптимизация изображений на сайт. Создание иконки сайта (favicon). Оформление страницы с учётом принципов дизайна.

6. Введение в JavaScript. Интерактивность.

Теория: Роль JavaScript в веб-разработке. Способы подключения JS (встроенный, внешний). Переменные и типы данных (числа, строки, булевы значения). Базовые операторы. Условные конструкции (if-else). Функции (объявление, вызов). События (onclick, onmouseover, onchange). Введение в DOM: получение элементов (getElementById, querySelector), изменение содержимого (innerHTML, textContent), изменение стилей (style). Таймеры (setTimeout, setInterval).

Практика: Создание кнопки, меняющей текст на странице при нажатии. Разработка простого калькулятора (сложение, вычитание). Создание игры "Угадай число" с подсказками. Изменение стилей элементов по событиям (смена цвета, скрытие/показ). Создание простого слайдера изображений. Обработка данных из формы.

7. Итоговый проектный модуль.

Теория: Этапы разработки сайта: идея, структура, дизайн, вёрстка, тестирование, публикация. Оформление проекта. Основы работы с системами контроля версий (базовые команды). Способы публикации сайта в интернете (бесплатные хостинговые сервисы). Подготовка презентации проекта.

Практика: Выбор темы проекта (сайт-портфолио, блог, сайт для школьного проекта, страница о хобби). Разработка структуры сайта (карта сайта). Создание дизайна (цвета, шрифты, макеты). Вёрстка всех страниц с использованием HTML, CSS, Flexbox/Grid. Добавление интерактивных элементов (меню, галерея, формы). Тестирование сайта в разных браузерах и на разных устройствах. Публикация сайта в интернете. Подготовка презентации. Защита проекта.

1.4. Планируемые результаты

Обучающие 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.

Развивающие результаты:

- Развитие критического и творческого мышления, умения анализировать, сравнивать, обобщать.
- Овладение универсальными учебными действиями: целеполагание, планирование, самоконтроль, рефлексия.
- Развитие коммуникативных навыков: умение вести диалог, распределять роли в совместной деятельности, эффективно взаимодействовать со сверстниками.
- Развитие регулятивных навыков: умение ставить цели, преодолевать трудности, оценивать свои действия.

Воспитательные результаты:

- Сформированность мотивации к познанию и творчеству, ценностное отношение к знаниям.
- Формирование культуры общения и уважительного отношения к мнению других.
- Развитие таких качеств, как трудолюбие, ответственность, самостоятельность, умение работать в команде.
-

Раздел 2. Комплекс организационно-педагогических условий

2.1. Календарный учебный график

№ п/п занятия	Тема занятия	Форма занятия	Количество часов	Форма контроля
Раздел 1. Введение в веб-разработку. Инструменты – 4 ч				
1	Введение в веб-разработку. Техника безопасности. Устройство интернета. Знакомство с редактором кода. Создание первой HTML-страницы.	КЗ	2	Практическая работа
2	Структура HTML-документа. Основные теги (заголовки, абзацы). Работа с DevTools.	КЗ	2	Практическая работа
Раздел 2. Основы HTML – 14 ч				

1	Списки и ссылки. Создание списков и ссылок.	КЗ	2	Практическая работа
2	Изображения и таблицы. Вставка изображений, создание таблиц.	КЗ	2	Практическая работа
3	Формы: основные элементы (input, textarea, select, button). Создание формы.	КЗ	2	Практическая работа
4	Семантические теги HTML5. Структура страницы (header, nav, main, footer). Создание структуры сайта.	КЗ	2	Самостоятельная работа
5	Сборка многостраничного сайта (перелинковка). Создание нескольких страниц с навигацией.	Практикум	2	Практическая работа
6	Повторение и закрепление HTML. Решение задач.	Практикум	2	Самостоятельная работа
7	Тестирование по разделу «Основы HTML».	Тестирование	2	Тестирование
	Раздел 3. Основы CSS – 16 ч			
	Введение в CSS. Способы подключения стилей. Базовые селекторы. Подключение CSS, изменение цвета и шрифта.	КЗ	2	Практическая работа
	Свойства текста и фона. Оформление текста, фона.	КЗ	2	Практическая работа
	Блочная модель: отступы, рамки, размеры. Работа с margin, padding, border.	КЗ	2	Практическая работа
	Каскадность и наследование.	КЗ	2	Практическая работа

Псевдоклассы (:hover). Создание меню с эффектами наведения.			
Псевдоэлементы и группировка селекторов. Оформление списков и ссылок.	КЗ	2	Самостоятельная работа
Стилизация форм и таблиц. Оформление формы, таблицы.	КЗ	2	Практическая работа
Создание единого стиля для сайта. Стилизация всех страниц.	Практикум	2	Практическая работа
Обобщающее занятие по CSS. Решение задач.	Практикум	2	Самостоятельная работа
Раздел 4. Позиционирование и адаптивная вёрстка – 10 ч			
Позиционирование элементов (relative, absolute, fixed). Позиционирование блоков.	КЗ	2	Практическая работа
Введение во Flexbox. Основные свойства. Создание гибкого меню.	КЗ	2	Практическая работа
Flexbox: выравнивание, распределение. Верстка карточек.	КЗ	2	Практическая работа
Введение в CSS Grid. Построение макета с Grid.	КЗ	2	Практическая работа
Адаптивная вёрстка: медиазапросы. Адаптация сайта под мобильные устройства.	КЗ	2	Практическая работа
Раздел 5. Основы веб-дизайна и работа с графикой – 6 ч			
Основы веб-дизайна: цвет, композиция, типографика. Подбор цветовой схемы, выбор шрифтов.	КЗ	2	Практическая работа
Работа с графикой: форматы изображений, оптимизация. Создание	КЗ	2	Самостоятельная работа

	favicon. Добавление и оптимизация изображений.			
	Разработка макета страницы (на бумаге или в графическом редакторе). Оформление страницы по принципам дизайна.	Практикум	2	Практическая работа
Раздел 6. Введение в JavaScript. Интерактивность – 10 ч				
	Введение в JavaScript. Подключение скриптов. Переменные, типы данных. Создание простого скрипта.	КЗ	2	Практическая работа
	Условные конструкции и функции. Создание калькулятора.	КЗ	2	Практическая работа
	События и обработчики (onclick, onmouseover). Кнопка меняет текст.	КЗ	2	Практическая работа
	Работа с DOM: получение элементов, изменение содержимого и стилей. Игра "Угадай число".	КЗ	2	Практическая работа
	Таймеры и анимации. Слайдер изображений.	КЗ	2	Самостоятельная работа
Раздел 7. Итоговый проектный модуль – 12 ч				
1	Этапы разработки проекта. Выбор темы. Создание структуры сайта.	ПР	2	Проектная деятельность
2	Разработка дизайна и макета. Верстка основных страниц.	ПР	2	Проектная деятельность
3	Добавление интерактивности (меню, формы, галерея).	ПР	2	Проектная деятельность
4	Тестирование, отладка, адаптация под разные устройства.	ПР	2	Проектная деятельность

5	Подготовка презентации и публикация проекта.	ПР	2	Проектная деятельность
6	Защита проектов	Защита проекта	2	Защита проекта

Календарный учебный график заполнен с помощью условных обозначений:

КЗ – комбинированные занятия, сочетающие элементы теории и практики;

П – проверочное занятие;

ПР – проектная работа (работа над кейсами).

2.2. Условия реализации программы

Материально-техническое обеспечение

− персональные компьютеры (на каждого обучающегося) с программным обеспечением, с минимальными системными требованиями (процессор Intel Core i3 или аналог AMD, 8 Гб оперативной памяти, 50 Гб свободного дискового пространства);

− экран, проектор или интерактивная доска для демонстрации учебного материала;

− доступ к сети Интернет (скорость не менее 10 Мбит/с) для работы с репозиториями и соревновательными платформами;

− установленный на каждый компьютер текстовый редактор кода (Visual Studio Code, Sublime Text или аналоги) для HTML/CSS/JS;

− современные браузеры (Google Chrome, Mozilla Firefox или Яндекс Браузер) с встроенными инструментами разработчика;

Информационное обеспечение

Специальные компьютерные программы:

− текстовый редактор кода (Visual Studio Code, Sublime Text или аналог);

− браузер для работы с онлайн-ресурсами с инструментами разработчика;

− офисный пакет для работы с документацией (LibreOffice).

Информационные технологии:

− использование облачных сервисов для хранения и обмена файлами (Яндекс Диск);

− использование сервисов для совместной работы над кодом (GitHub/GitLab).

Кадровое обеспечение

1. Требования к кадровому обеспечению деятельности Центра «IT-куб» определяются образовательной организацией самостоятельно с учетом действующего трудового законодательства.

2. Образовательную деятельность по дополнительным общеобразовательным программам на базе Центра «IT-куб» осуществляют педагоги дополнительного образования. В соответствии с пунктом 4 статьи 46 Федерального закона от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации» к занятию педагогической деятельностью по дополнительным общеобразовательным программам допускаются лица, обучающиеся по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующим направленности дополнительных общеобразовательных программ, и успешно прошедшие промежуточную аттестацию не менее чем за два года обучения. Соответствие образовательной программы высшего образования направленности дополнительной общеобразовательной программы определяется образовательной организацией.

2.3. Формы аттестации

Формы оценки уровня достижений обучающегося

Для контроля и самоконтроля за эффективностью обучения применяются методы:

− предварительные (наблюдение, входной тест);

− текущие (наблюдение, решение задач на занятии, проверка практических работ, самостоятельные работы);

− промежуточные (тестирование, практическая работа, зачётное занятие);

− итоговые (защита проекта).

Формы фиксации образовательных результатов

Для фиксации образовательных результатов в рамках курса используются:

− портфолио работ обучающихся;

− отзывы обучающихся по итогам занятий и итогам обучения.

Формы предъявления и демонстрации образовательных результатов:

− презентация проекта.

Формы подведения итогов реализации программы:

− педагогическое наблюдение;

− педагогический анализ выполнения обучающимися учебных заданий;

− активность обучающихся на занятиях.

2.4. Оценочные материалы

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:

Оценка параметров	*Уровень по сумме баллов*
Начальный уровень	7-14 баллов
Средний уровень	14-21 баллов
Высокий уровень	21-31 баллов

Текущий контроль (оценка усвоения изучаемого материала) осуществляется педагогом на протяжение 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 интерактивности JavaScript, на основе определенных критериев (Приложение 2).

3. Список литературы

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.
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. Смирнов, А. А. Введение в HTML и CSS. — М.: Эксмо, 2019.
9. Тихомиров, С. И. JavaScript: от начинающего до профессионала. — М.: Урал, 2023.
10. Карасев, А. И. Современные технологии веб-разработки. — М.: Курс, 2023.
11. Современный учебник JavaScript / [Электронный ресурс] // JSR: [сайт]. Режим доступа: https://learn.javascript.ru// (дата обращения: 10.08.2026).
12. 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: 04.08.2026).
13. Справочник CSS/ [Электронный ресурс] // WebReference: [сайт]. Режим доступа: https://webref.ru/css (дата обращения: 10.08.2026).

Приложение 1

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-стилизованный 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: У:=5; Х:=2; У:=(У-2*Х)/2+Х

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

Приложение 2

Критерии оценивания итогового проекта сайта:

№ п/п	Критерий	Показатели	Балл/ Макс.балл
1	Работоспособность	Сайт открывается, навигация работает, формы обрабатывают ввод, интерактив выполняется, ошибок в консоли нет.	___/25
2	Структура и семантика HTML	Правильное использование тегов, семантические элементы, иерархия заголовков, валидность кода.	___/15
3	Оформление и дизайн (CSS)	Единый стиль, оформление текста и фонов, применение Flexbox/Grid, адаптивность (медиазапросы), эстетичность.	___/20
4	Интерактивность (JavaScript)	Минимум два работающих интерактивных элемента (события, изменение стилей, анимация, таймеры), логичный код без ошибок.	___/20
5	Качество кода и оформление	Аккуратное форматирование, осмысленные имена, комментарии, структура папок, наличие README-файла.	___/10
6	Защита проекта	Четкая презентация цели и технологий, ответы на вопросы, демонстрация работы сайта.	___/10
Итого:	Оценка: _____		___/100
	Проверил: _____		

Шкала перевода баллов:

Уровень	**Баллы**	**Оценка**
Высокий	85–100	«отлично»
Выше среднего	70–84	«хорошо»

Средний	50–69	«удовлетворительно»
Низкий	менее 50	«неудовлетворительно»
Уровень	Баллы	Оценка