

Муниципальное бюджетное общеобразовательное учреждение г. о. Королёв  
Московской области «Гимназия № 3 имени Л.П.Данилиной»

*Индивидуальный творческий проект*

*по теме*

*«Различные алгоритмы программирования для учащихся средней школы»*

Учащийся 10-Б класса: Шумилин Иван

Руководитель: Герасимова Т. И., учитель информатики

Допуск к защите \_\_\_\_\_.

2021 год

## Оглавление

1. Введение.....	Стр.3
2. Теоретическая часть.....	Стр.5
2.1. Причина выбора сайта для хранения информации .....	Стр.5
2.2. Выбор инструментов для создания сайта.....	Стр.6
2.3. Подготовка конспектов с принципами работы алгоритмов.....	Стр.7
3. Практическая часть.....	Стр.7
3.1. Создание статей в формате .html и иллюстраций к ним.....	Стр.7
3.2. Разработка и реализация системы компиляции.....	Стр.8
3.3. Создание приятного интерфейса сайта.....	Стр.9
3.4. Добавление статей на сайт.....	Стр.10
4. Заключение.....	Стр.10
5. Список использованных источников.....	Стр.11
6. Приложение 1.....	Стр.12
7. Приложение 2.....	Стр.14

## 1. Введение

### *Цель проекта*

Прежде всего, целью проекта является создание веб-ресурса “Algorithm” – сайта в сети Интернет, который будет содержать некоторое количество статей об одной из практических разделов предмета “Информатика и ИКТ”, именуемого программированием. Предполагается, что данный веб-ресурс облегчит работу учителей предмета “Информатика и ИКТ” и улучшит понимание основ программирования учащимися. Поэтому, конечной целью проекта можно назвать помощь тем, кто обучает информатике или учит её.

### *Актуальность*

На взгляд проектанта, данный проект является весьма актуальным, а при успешной разработке будет весьма полезен для любого учащегося в 7-9 классах, владеющего русским языком. Причин для вышеуказанного заявления несколько, ниже приведены основные из них:

**Личный опыт:** в течение трёх лет обучения проектант постоянно сталкивался с многочисленными просьбами о помощи, исходящими от других учащихся, у которых возникали проблемы, с несложными, на взгляд разрабатывающей проект личности, алгоритмами. Также ему довелось напечатать небольшой неофициальный цикл статей, стиль объяснения внутри которых в профессиональной среде программистов называется “на пальцах”. К сожалению, некоторая часть статей была утеряна на безвозвратной основе, некоторая – не может быть опубликована, так как была нацелена на *очень* узкую аудиторию. Однако именно в тот период времени у проектанта зародилась идея о более серьёзном сборнике, который мог бы помочь учителям информатики в их работе.

**Проведённый опрос:** к сожалению, опрос проводился довольно давно (в 2019 году), носил устную форму и не был задокументирован, так как проектант не предполагал, что это пригодится ему позже. Но весьма многие личности тогда согласились, что были бы рады увидеть более полный цикл статей, описанный проектантом. Это стало первой и основной причиной выбора поименованной выше темы проекта.

### ***Описание предполагаемого продукта***

Продуктом, согласно замыслу автора проекта, является веб-ресурс “Algorithm”, разработанный на основе созданной проектантом системы *компиляции* (сборки) определённого количества *наполнения* (статей) в *.html* (формат хранения Интернет-сайтов) путём совмещения их с заранее созданными *шаблонами* (общими для любой страницы сайта частями). Описываемый веб-ресурс будет содержать некоторое количество статей об одной из практических частей предмета “Информатика и ИКТ”, именуемой программированием. Возможно, он будет размещён в сети Интернет (при наличии в свободном доступе бесплатного *хостинга* – ресурса, который позволяет разместить сайт в Сети), либо – до накопления достаточного количества денежных средств проектантом для использования коммерческого хостинга – он будет находиться в свободном доступе в *репозитории* – удалённом (*remote*) хранилище – **github.com**.

Важно заметить, что продукт проекта не будет являться полноценной заменой урокам “Информатики и ИКТ”, так как: **а)** будет содержать объяснения лишь небольшой части предмета, которая, согласно личному опыту проектанта и проведённым им опросам, вызывает наибольшие затруднения в освоении у учащихся; **б)** разработчик проекта также на данный момент является учащимся и не имеет педагогического образования.

### ***Этапы выполнения проекта***

- **Собрать определённое количество информации.** Этап можно разбить на следующие задачи:
  - Выбрать необходимые для создания сайтов инструменты;
  - Подготовить небольшие конспекты о принципах работы алгоритмов с наглядными примерами;
- **Создать цикл доступных статей об алгоритмах.** Этап можно разбить на следующие задачи:
  - Основываясь на созданных ранее конспектах, подготовить необходимое количество статей в формате *.html*;
  - Подготовить наглядные иллюстрации (т. е. “картинки”) в формате *.png* (весьма удобный для редактирования и поддерживаемый большинством браузеров формат хранения изображений);
- **Разработать web-ресурс для хранения алгоритмов, описанных выше.** Этап можно разбить на следующие задачи:

- Разработать и реализовать систему, позволяющую быстро компилировать сайт из готовых шаблонов и статей;
- Создать приятный внешне и удобный в использовании интерфейс сайта;
- Поочерёдно добавлять статьи на готовый сайт;

## Анализ работы над проектом

### 2. Теоретическая часть

#### 2.1. Причина выбора сайта для хранения цикла статей

Так как название проекта содержит слово “алгоритмов” во множественном числе, то логично сделать следующий вывод: продуктом будет являться не одна статья, разъясняющая конкретный алгоритм, а целый цикл или сборник.

Способов хранения массивов статей существует немалое количество: справочник или учебник в максимально привычном формате, компакт-диск с лекциями в формате видеоуроков, веб-ресурс и другие. Конечно, люди различаются по типу информации, которую им максимально удобно воспринимать, но всё же самым распространённым в современном обществе определёнno является текстовый способ хранения информации. Он имеет ряд преимуществ, благодаря которым и был мной выбран: объем, необходимый для хранения *(для сравнения, видео в формате HD720p (30fps) с форматом аудио AAC 192kbps длительностью в 43 (сорок три) минуты занимает 2Gb свободного места, а субтитры в виде текста к этому же видео – 50Kb, что почти в 42 000 (сорок две тысячи) раз меньше)*, относительная простота создания статей, доступность буквально каждому.

Местом же хранения был выбран доступный каждому современному человеку, не обладающему никакими специфическими навыками, ресурс – сеть Интернет. Как уже было сказано, его главным достоинством является общедоступность, что и сыграло ключевую роль в выборе места размещения продукта проекта.

Существует несколько способов хранения статей в Интернете, однако был выбран самый надёжный: воспользоваться хостинговым ресурсом и разместить сайт со статьями по специальному *URL-адресу* (адресу, через который можно получить доступ к находящемуся в сети Интернет сайту).

Этот способ имеет следующие преимущества и недостатки:

+ **Легкость восприятия:** информацию гораздо легче воспринимать при наличии красивого оформления и удобной навигации;

- **Сложность в создании:** “красивое оформление и удобную навигацию” придётся создавать вручную;

+ **Свобода выбора:** отсутствуют ограничения на что-либо; при желании можно размещать на своём сайте любое количество рекламы, вводить оплату за определённые действия, оформлять в собственном стиле;

- **Стоимость:** главный, с точки зрения проектанта минус; однако существуют хостинги, выдающие URL-адрес на ограниченный срок, чем можно воспользоваться для демонстрации работы проекта;

Таким образом, единственным существенным недостатком размещения сайта по URL-адресу является необходимость в наличии, собственно, грамотно созданного и приятно оформленного сайта.

## ***2.2. Выбор инструментов для создания сайта***

Так как сайт должен содержать лишь определённое количество статей и не будет выполнять продвинутого взаимодействия с пользователем, то набор инструментов для его создания не будет слишком большим:

- **Язык вёрстки: HTML** – стандартизированный язык разметки документов во Всемирной паутине (**ru.wikipedia.org**); именно он будет отвечать за элементы, находящиеся на сайте, то есть – за содержимое ресурса;
- **Язык оформления: CSS** – язык для формирования внешнего вида документа, созданного с помощью языка разметки (**ru.wikipedia.org**); при правильном использовании он гарантирует сайту благоприятный внешний вид;
- **Скриптовый язык: JavaScript** – прототипно-ориентированный сценарный язык программирования (**ru.wikipedia.org**); с его помощью будет реализована работа некоторых интерактивных элементов; к примеру – кнопки “скрыть меню навигации” (Приложение 2 на двух страницах, **рис. 1** – меню открыто, **рис. 2** – меню скрыто);
- **Язык для разработки системы компиляции: Python3** – высокоуровневый язык программирования общего назначения с динамической строгой типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ (**ru.wikipedia.org**); с его помощью планируется разработка уникальной авторской системы сборки сайта;

Утверждается, что при грамотном использовании этих инструментов можно создать жизнеспособный сайт с приятным оформлением. При применении же более продвинутого ПО необходимо качественное оборудование: рабочая ЭВМ (Электронно-Вычислительная Машина) проектанта не способна справиться с более современными средствами разработки.

### ***2.3. Подготовка конспектов с принципами работы алгоритмов***

Для достаточной грамотности в объяснении материала проектантом было изучено немало статей на подобных ресурсах (самый известный из них – **habr.com**, остальные были весьма разрознены и использовались для того, чтобы проектант мог увидеть, как другие люди разбирают алгоритмы) и собственных конспектов, которые были записаны во время занятий на курсах программирования. Все конспекты хранились в письменном виде и на данный момент, к сожалению, находятся вне зоны доступа.

Все примеры к использованию алгоритмов были придуманы проектантом самостоятельно и описаны на трёх языках программирования:

- Чаще всего в российских школах преподают **Pascal**, поэтому его нельзя было не включить;
- Язык **Python3** становится очень популярным; в некоторых столичных школах переходят на обучение ему; многие, кто учился программированию самостоятельно, обучался на нём;
- Язык **C++** уже много лет считается негласным стандартом в программировании и не сдаёт позиции; в большинстве случаев алгоритмы описывают именно на нём;

## **3. Практическая часть**

### ***3.1. Создание статей в формате .html и иллюстраций к ним***

Так как в следующей цели одна из задач предполагает создание уникальной (авторской) системы компиляции сайта, то файлы со статьями могут быть освобождены от лишней нагрузки (оформление и функционал будут вынесены в отдельные шаблоны), следовательно, они должны содержать только непосредственно текст статей.

Все статьи, как и поясняющие рисунки, во избежание каких-либо проблем с авторскими правами, были написаны проектантом собственноручно. Стоит отдельно сказать, что стиль текста был выбран научно-популярный (как наиболее понятный человеку с разумом учащегося средней школы), с некоторым упрощением объясняемой информации (к примеру, замена части терминов более простыми), но без перехода в шутки и без отходов от темы статьи.

Форматом иллюстраций был выбран **.png**, который охарактеризован учебником информатики за десятый класс К. Ю. Полякова и Е. А. Еремина следующим образом: **PNG** (англ. *Portable Network Graphics – переносимые сетевые изображения; файлы с расширением png*) – формат, поддерживающий как режим истинного цвета, так и кодирование с палитрой; части изображения могут быть прозрачными и даже полупрозрачными (32-битное кодирование RGBA, где четвёртый байт задаёт прозрачность); изображение сжимается без искажения; анимация не поддерживается. Причиной такого выбора стало:

- Небольшой относительно других часто используемых в WEB-разработке форматов размер;
- Отсутствие возможности реализовать анимацию, что не было необходимым;
- Возможность удобной работы с данным форматом во всех растровых редакторах, которые для своего запуска не требуют эмуляции DOS;

### ***3.2. Разработка и реализация системы компиляции***

Для максимально комфортной работы над разработкой WEB-сайта (под “максимально комфортной” автор подразумевает максимально автоматизированную, когда от человека требуется только создавать статьи и без излишних усилий добавлять их на сайт) проектанту требовалась уникальная система компиляции сайта. Такие, безусловно, существуют, и далеко не в едином экземпляре. Но есть один нюанс. Он заключается в том, что эти системы работают со слишком мощными для ЭВМ автора проекта инструментами. Следовательно, требуется своя система, которая будет использовать простейшие, поименованные выше средства разработки. Таковая была реализована проектантом в сжатые сроки (так как не являлась центральной темой проекта). Были написаны два *скрипта* (сценария, в котором прописаны отдельные последовательности действий, созданные для автоматического выполнения задачи). Один из них подставляет в файлы **.css**, что отвечают за оформление сайта, на места заранее определённых констант правильные цвета. Это очень удобно, так как в трёх файлах (для трёх размеров сайта) содержатся одни и те же цвета; вышеупомянутый скрипт был написан, чтобы не менять их значения вручную. Второй же собирает все страницы сайта по одной и той же схеме, подставляя содержимое внутрь и помогая избежать монотонной работы (к примеру, обновления меню навигации на всех страницах вручную) и сконцентрировать внимание разработчика на качестве статьи. Подробнее вы можете узнать о принципе работы системы компиляции в Приложении 1 на двух страницах (место в записке ограничено, да и информация третьестепенна).



### ***3.3. Создание приятного интерфейса сайта***

Оформление сайта есть работа скорее для людей творческих профессий. Создание всех элементов интерфейса не создало в своём процессе никаких сложностей, так как у автора проекта уже был подобный опыт. Но с выбором цветовой гаммы WEB-страниц возникли некоторые эксцессы. После нескольких малоудачных попыток оформить сайт проектантом самостоятельно, он обратился к специализированным источникам. Была выбрана подобранная образованными в нужной сфере людьми палитра, размещённая на сайте [https://www.canva.com/ru\\_ru/obuchenie/100-cvetovyx-sochetnij/](https://www.canva.com/ru_ru/obuchenie/100-cvetovyx-sochetnij/). Цвета из неё путём нехитрых перестановок были распределены по сайту максимально комфортным для глаз образом. После чего было совершено обращение за помощью к нескольким (двоим) специалистам по графическому дизайну, которые пожелали остаться неизвестными; таким образом была получена не раздражающая глаз и не отвлекающая от просмотра контента палитра. Цвета были применены моментально, что было обеспечено собственной системой компиляции сайта.

Меню навигации, которое сперва казалось очень удобным, чуть позже (когда началось размещение содержимого на сайте) начала мешать просмотру контента. Без этого меню не обходится ни один приличный сайт, посему было принято решение сделать её скрываемой. Для этого был применён язык JS, позволивший реализовать кнопку “Скрыть/показать меню” (Приложение 2 на двух страницах, **рис. 1** – меню открыто, **рис. 2** – меню скрыто). Таким образом, вышеописанная проблема была устранена в зародыше.

Ещё стоит добавить, что путём хитрых манипуляций удалось сделать сайт адаптивным, то есть подстраивающимся под размер экрана. Это заняло весьма немалое количество времени, однако результат был того достоин: теперь открыть ресурс можно с любого устройства – и везде он будет выглядеть отлично (Приложение 2 на двух страницах, **рис. 3** – сайт просматривается с большого экрана, **рис. 4** – с малого).

### *3.4. Добавление статей на сайт*

Эта задача была самой простой по той причине, что всё необходимое было подготовлено заранее. Статьи – написаны, страницы – оформлены, процесс добавления контента – автоматизирован. Хотя и несложная, эта задача была заключающей в списке. После её выполнения (и добавления некоторой информации на страницу “О нас”) проект можно считать завершённым.

## **4. Заключение**

В ходе выполнения данного проекта ведущий его учащийся заметно развил свои навыки сайтостроения (которые раньше находились не совсем на нулевом уровне, но были заметно ниже), разработал собственную систему сборки сайта, которая на данный момент уже используется им в другом проекте; он провёл большое количество времени за очень интересным занятием – вёрсткой WEB-страниц, их оформлением и описанием алгоритмов так, чтобы их мог понять любой. Все этапы выполнения были подробно задокументированы в данной записке.

На взгляд проектанта, получившийся продукт не только полностью удовлетворяет предварительному описанию, но и превосходит его, так как в процессе разработки была создана уникальная система компиляции, которую очень удобно применять для разработки небольших сайтов без динамического наполнения и которая уже используется в другом проекте этого же автора.

На данный момент весь проект размещён в открытом доступе на репозитории [https://github.com/Mirovengil/school\\_project/tree/](https://github.com/Mirovengil/school_project/tree/). Если цель обращения к исходным файлам проекта – их просмотр, то обращение следует совершать к ветке **demo**. В случае же, когда посетитель репозитория хочет получить все исходники для собственноручного редактирования, ему следует обратиться к ветке **Production**, где кроме самого сайта размещена и система компиляции.

## ЛИТЕРАТУРА

1. Информатика. Углублённый уровень : учебник для 10 класса : в 2 ч. Ч. 1 / К. Ю. Поляков, Е. А. Еремин. – 4-е изд. – М. : БИНОМ. Лаборатория знаний, 2015. – 344 с. : ил.
2. Усенков Д. Уроки Web-мастера. – М.: Лаборатория Базовых Знаний, 2001. – 432 с.: ил.

## ИНТЕРНЕТ-РЕСУРСЫ

1. [ru.wikipedia.org](http://ru.wikipedia.org) – общедоступная многоязычная универсальная интернет-энциклопедия со свободным контентом, реализованная на принципах вики.
2. [htmlbook.ru](http://htmlbook.ru) – набор взаимосвязанных уроков по изучению языка разметки документов каскадных таблиц стилей.
3. [welcome4u.ru](http://welcome4u.ru) – образовательный портал для начинающих программистов.

## Инструкция к использования системы

- Чтобы **изменить настройки** css – запустите ./css/color\_changer.py.
- Чтобы **собрать сайт** – запустите ./compile.py.

## Оформление

- Все значки находятся в ./img/icons. Если что-то не подходит, это можно перерисовать или добавить новые значки.
- В папке css находятся шаблоны .txt. В них можно менять оформление чего-либо. Цифры (1000; inf), (600; 1000), (0; 600) обозначают разрешение экрана, при котором применяется его содержимое.
- В нём же можно поменять цветовое оформление: все цвета заданы, как переменные. Комментарии в коде указывают, к чему относится цвет. Достаточно заменить цвета другим значением HEX-RGB и выполнить скрипт.
- В папке template/std содержатся файлы .txt – шаблоны элементов страницы сайта. Если вам не нравится верхняя плашка сайта, где отображаются название сайта, его логотип (он же – ссылка на главную страницу) и кнопка “скрыть/показать меню навигации”), или её содержимое, то можете изменить файл header.txt. Всё остальное (JS.txt, init.txt, footer.txt) изменять не рекомендуется, т.к. они не несут в себе графических элементов:
- init.txt подключает .css файлы к странице и открывает некоторые ведущие теги;
- footer.txt содержит стрелочку “Наверх”, которая отображается внизу страницы;
- JS.txt отвечает за работоспособность кнопки “скрыть/показать меню навигации”;

## Наполнение

- Добавить содержание статьи в файл `./template/<name>.txt`, где `<name>` – имя файла.
  - Файл может содержать определённые теги, которые при компиляции дадут различные эффекты:
    - тег `<~name~>` будет заменён на название страницы;
    - тег `<~tab~>` будет заменён на четыре пробела (что в стандартном HTML5 сделать проблематично);
    - содержимое тега `<~py~></>` будет отформатировано и помечено, как код на языке Python3;
    - содержимое тега `<~pa~></>` будет отформатировано и помечено, как код на языке Pascal;
    - содержимое тега `<~c~></>` будет отформатировано и помечено, как код на языке C/C++;
- Добавить в файл `./makefile.txt` строчку вида `<name><тег><Название страницы>`, где:
  - `<name>` – имя файла с содержанием страницы в папке `./template`;
  - `<Название страницы>` – название страницы, которое будет отображаться на вкладке и на ссылке в меню навигации (можно на русском);
  - `<тег>`:
    - `<~key~>`, если страницу надо добавить в меню навигации;
    - `<~page~>`, если страницу не надо добавлять в меню навигации (тогда на неё должна быть ссылка с другой страницы, созданная вручную);
- Выполнить скрипт `compile.py`.



рис. 1



рис. 2



рис. 3

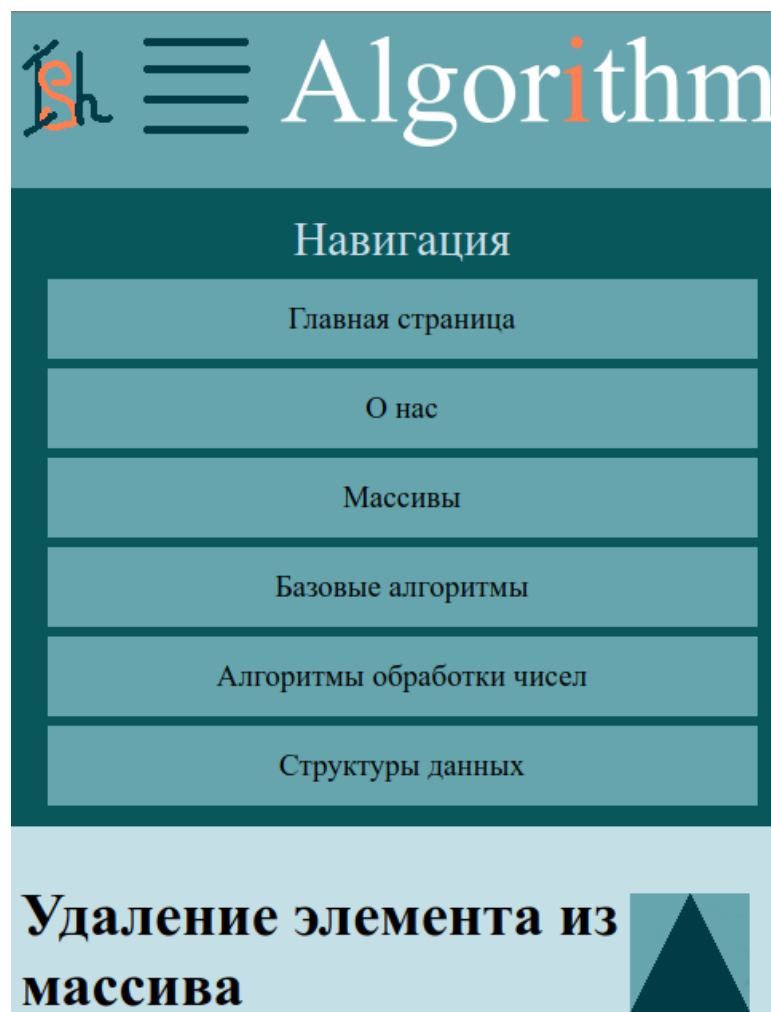


рис. 4