**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук  
Департамент программной инженерии

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Доцент департамента больших данных и информационного поиска на факультете компьютерных наук, к.ф.-м.н.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ И.Ю. Самоненко «\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2020 г. | УТВЕРЖДАЮ  Академический руководитель образовательной программы «Программная инженерия» профессор департамента программной инженерии, канд. техн. наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В. В. Шилов «\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2020 г. |

|  |  |
| --- | --- |
| ***Подп. и дата*** |  |
| ***Инв. № дубл.*** |  |
| ***Взам. инв. №*** |  |
| ***Подп. и дата*** |  |
| ***Инв. № подл*** | RU.17701729.503390-01 ТЗ 01-1-ЛУ |

**WEB ПРИЛОЖЕНИЕ ДЛЯ ВИЗУАЛИЗАЦИИ АЛГОРИТМА АХО-КОРАСИК**

**Техническое задание**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.** **04.13-01 01-1-ЛУ**

Исполнитель  
студент группы БПИ183  
\_\_\_\_\_\_\_\_\_\_ /Капур Даниэл Сингх  
«\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2020 г.

**Москва 2020**

УТВЕРЖДЕНRU.17701729.503390-01 ТЗ 01-1-ЛУ

|  |  |
| --- | --- |
| Подп. и дата |  |
| Инв. № дубл. |  |
| Взам. инв. № |  |
| Подп. и дата |  |
| Инв. № подл | RU.17701729.503390-01 ТЗ 01-1 |

**WEB ПРИЛОЖЕНИЕ ДЛЯ ВИЗУАЛИЗАЦИИ АЛГОРИТМА АХО-КОРАСИК**

**Техническое задание**

**RU.17701729.** **04.13-01 01-1-ЛУ**

**Листов 16**

**Москва 2020**

# **Аннотация**

Техническое задание – это основной документ, оговаривающий набор требований и порядок создания программного продукта, в соответствии с которым производится разработка программы, ее тестирование и приемка.

Настоящее Техническое задание на разработку «Web Приложение для визуализации алгоритма Ахо-Корасик» содержит следующие разделы: «Введение», «Основание для разработки», «Назначение разработки», «Требования к программе», «Требования к программным документам», «Технико-экономические показатели», «Стадии и этапы разработки», «Порядок контроля и приемки» и приложения [7].

В разделе «Введение» указано наименование и краткая характеристика области применения «Web Приложение для визуализации алгоритма Ахо-Корасик»

В разделе «Основания для разработки» указан документ на основании, которого ведется разработка и наименование темы разработки.

В разделе «Назначение разработки» указано функциональное и эксплуатационное назначение программного продукта.

Раздел «Требования к программе» содержит основные требования к функциональным характеристикам, к надежности, к условиям эксплуатации, к составу и параметрам технических средств, к информационной и программной совместимости, к маркировке и упаковке, к транспортировке и хранению, а также специальные требования.

Раздел «Требования к программным документам» содержит предварительный состав программной документации и специальные требования к ней.

Раздел «Технико-экономические показатели» содержит ориентировочную экономическую эффективность, предполагаемую годовую потребность, экономические преимущества разработки «Программного инструмента для построения и изучения многочленов, соответствующих геометрическим графам».

Раздел «Стадии и этапы разработки» содержит стадии разработки, этапы и содержание работ.

В разделе «Порядок контроля и приемки» указаны общие требования к приемке работы.

Настоящий документ разработан в соответствии с требованиями:

1. ГОСТ 19.101-77 Виды программ и программных документов [1];
2. ГОСТ 19.102-77 Стадии разработки [2];
3. ГОСТ 19.103-77 Обозначения программ и программных документов [3];
4. ГОСТ 19.104-78 Основные надписи [4];
5. ГОСТ 19.105-78 Общие требования к программным документам [5];
6. ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом [6];
7. ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению [7].

Изменения к данному Техническому заданию оформляются согласно ГОСТ 19.603-78 [8], ГОСТ 19.604-78 [9].

**Содержание**

[1. Введение 4](#_Toc498250216)

[1.1.Наименование программы 4](#_Toc498250217)

[1.2.Краткая характеристика области применения 4](#_Toc498250218)

[2. Основания для разработки 5](#_Toc498250219)

[2.1Документы, на основании которых ведется разработка 5](#_Toc498250220)

[2.2 Наименование темы разработки 5](#_Toc498250221)

[3. Назначение разработки 6](#_Toc498250222)

[3.1. Функциональное назначение 6](#_Toc498250223)

[3.2. Эксплуатационное назначение 6](#_Toc498250224)

[4. Требования к программе 7](#_Toc498250225)

[4.1. Требования к функциональным характеристикам 7](#_Toc498250226)

[4.1.1. Требования к составу выполняемых функций 7](#_Toc498250227)

[4.1.2. Требования к организации входных данных 7](#_Toc498250228)

[4.1.3. Требования к организации выходных данных 7](#_Toc498250229)

[4.2. Требования к надежности 7](#_Toc498250230)

[4.3. Требования к интерфейсу 8](#_Toc498250231)

[4.4. Условия эксплуатации 8](#_Toc498250232)

[4.5. Требования к составу и параметру технических средств 8](#_Toc498250233)

[4.6. Требования к информационной и программной совместимости 8](#_Toc498250234)

[4.7. Требования к маркировке и упаковке 8](#_Toc498250235)

[4.8. Требования к транспортированию и хранению 8](#_Toc498250236)

[5. Требования к программной документации 9](#_Toc498250237)

[6. Технико-экономические показатели 10](#_Toc498250238)

[7. Стадии и этапы разработки 11](#_Toc498250242)

[7.1. Необходимые стадии разработки, этапы и содержание работ 11](#_Toc498250243)

[7.2. Сроки разработки и исполнители 11](#_Toc498250244)

[8. Порядок контроля и приемки 13](#_Toc498250245)

[8.1. Виды испытаний 13](#_Toc498250246)

[8.2. Общие требования к приемке работы 13](#_Toc498250247)

9. Список использованной литературы……………………………………………………………………………………………...12

# **1. Введение**

* 1. **Наименование программы**

Наименование программы – «Web Приложение для визуализации алгоритма Ахо-Корасик» (Web Application for visualisation of Aho-Corasick algorithm).

* 1. **Краткая характеристика области применения**

Для успеш­ного усвоения механизма ра­боты алгоритмов необходимым условием является визуальное описание алгоритма для каж­дого этапа его выполнения.

Визуализация – это процесс представления информации, данных, знания в виде изобра­жения, имеющий целью макси­мальное удобство их восприя­тия, понимания и анализа.

При изучении механизма работы алгоритмов основу должны составлять наглядные сред­ства обучения. Для понима­ния структуры алгоритма и действий, которые происходят внутри алгоритма, необходимо наглядно увидеть каждый шаг выполнения алгоритма, что позволит студентам построить эффективную ментальную мо­дель, имитируя поведение исполняемого алгоритма.

Визуальное представление алгоритма позволит студенту лучше понять и увидеть какие структурные элементы исполь­зованы при проектировании алгоритма, каким образом описана логика взаимодей­ствия структурных элементов алгоритма, каким реальным объектам соответствуют опре­делённые элементы в реша­емой задачи. Следовательно, обучение алгоритмизации и программированию необходи­мо осуществлять с опорой на проектирование алгоритмов и дальнейшее написание про­грамм на основе разработан­ных алгоритмов с помощью конструкций языка програм­мирования.

# 

# **2. Основания для разработки**

* 1. **Документы, на основании которых ведется разработка**

Приказ Национального исследовательского университета "Высшая школа экономики" № 2.3-02/1012-02 от 10.12.19.

**2.2 Наименование темы разработки**

Наименование программы – «Web Приложение для визуализации алгоритма Ахо-Корасик» (Web Application for visualisation of Aho-Corasick algorithm).

Программа выполняется в рамках темы курсовой работы в соответствии с учебным планом подготовки бакалавров по направлению 09.03.04 «Программная инженерия» Национального исследовательского университета «Высшая школа экономики», факультет компьютерных наук, департамент программной инженерии.

# **3. Назначение разработки**

## **3.1. Функциональное назначение**

## Функциональным назначением приложения является обеспечение интерактивности и наглядности при управлением процессом визуализации со стороны пользователя. В то же время интерфейс для визуализатора должен быть прост в использовании.

Визуализатор должен понятно отображать все изменения значений переменных, используемых в алгоритме. Для большего понимания должны присутсвовать пояснения в ходе работы визуализатора.

## **3.2. Эксплуатационное назначение**

Преподаватель определяет условия по выполнению алго­ритма в зависимости от целей обучения (подробное или обоб­щённое изучение алгоритма). Например, определить набор входных данных, которые бу­дут использованы для решения некоторой задачи; определить шаг выполнения алгоритма и др. От студентов не требуется владеть определёнными зна­ниями и навыками для работы с визуализатором, они осмыс­ливают ход выполнения алго­ритма и пытаются предсказать следующий шаг работы алго­ритма. Кроме того, визуали­заторы алгоритмов могут быть использованы для развития ал­горитмического мышления не только у студентов, изучающих информатику.

Одной из сильных сторон визуализатора алгоритмов яв­ляется возможность задейство­вания слухового и зрительного канала для улучшения воспри­ятия учебной информации в процессе обучения.

Логику алгоритма нельзя полностью раскрыть лишь показывая, как меняются значения переменных, участву­ющих в алгоритме. Студентам необходимо предоставить пра­вильные графические представ­ления, которые лучше соответ­ствуют их ментальным моделям по выполнению конкретного алгоритма. Визуализатор алго­ритмов позволяет помочь сту­дентам построить правильные ментальные модели, связать структурные компоненты в ие­рархии построения алгоритма и обобщить модель алгоритма при решении подобных задач, визуализируя абстрактные кон­цепции и разворачивая основ­ную логику исследуемого алго­ритма.

# **4. Требования к программе**

## **4.1. Требования к функциональным характеристикам**

### **4.1.1. Требования к составу выполняемых функций**

Продукт должен реализовывать спроектированную систему взаимодействия с пользователем для успешного исполнения его потребностей. Продукт визуализирует алгоритм Ахо-Корасик и позволяет интерактивно работать с ним.

Программа должна давать пользователю возможность выполнять следующие действия:

* Предоставление возможности ввода своей строки для работы алгоритма.
* Демонстрация работы на определнных заранее строках, чтобы продемонстрировать работу алгоритма в худшем / лучшем / среднем случае.
* Шаг вперед (переход к следующему состоянию).
* Шаг назад (переход к предыдущему состоянию).
* Комментарии: текстовое описание того действия, которое сейчас происходит.
* Псевдокод: псевдокод и подсветка тех строк, которые сейчас выполняются.

### **4.1.2. Требования к организации входных данных**

Пользователю должна быть предоставлена возможность двух вариантов выбора входных данных:

* С клавиатуры. Пользователь сам определяет и вводит строку, на основе которой будет работать алгоритм.
* Выбор предустановленных вариантов. Варианты должны демонстрировать особенности алгоритма (состояний структуры данных): показывать работу в худшем / лучшем / среднем случае.

### **4.1.3. Требования к организации выходных данных**

Программа должна предоставлять пользователю возможность увидеть результирующее нагруженное дерево, найденные позиции вхождения подстрок, если есть, псевдокод для алгоритма, а также описание происходящего для текущего состояния.

## **4.2. Требования к надежности**

Для корректной работы программы необходима устойчивая работа компьютера, на котором она запущена.

Программа должна обрабатывать все исключительные ситуациии и обеспечивать проверку корректности входных данных.

## **4.3. Требования к интерфейсу**

Программа должна иметь интерфейс приложения, написанного на языках HTML, CSS, JavaScript с возможностью ввода входных данных и вывода работы алгоритма с пояснениями.

Должны присутсвовать поля для ввода строки и массива подстрок. Место для вывода резльтирующего нагруженного дерева и комментариев. Также кнопки для переключения шагов.

## **4.4. Условия эксплуатации**

Пользователь должен иметь начальное представление о работе со строками, в частности понимать термин «подстрока», чтобы сформировать массив подстрок.

## **4.5. Требования к составу и параметру технических средств**

Для нормального функционирования программы требуется компьютер, оснащенный следующими техническими компонентами:

1. процессор не ниже Intel Pentium/Celeron, AMDK6/Athlon/Duron или совместимый с ними с тактовой частотой не ниже 1 ГГц;
2. 1024 Мб ОЗУ или более;
3. жесткий диск с объемом свободной памяти не менее 1 Гб;
4. VGA-совместимые видеоадаптер и монитор с разрешением не ниже 1280х800;
5. Наличие браузера и стабильного подключения к интернету для отображения сайта.
6. клавиатура и мышь.

## **4.6. Требования к информационной и программной совместимости**

Для нормального функционирования программы требуется компьютер, оснащенный следующими программными компонентами:

1. операционная система MicrosoftWindows 7 / 8 / 8.1 / 10;
2. Любой современый браузер. К примеру Chrome, Firefox.

## **4.7. Требования к маркировке и упаковке**

Приложение распространяется посредством сети Интернет.

Программа сразу готова к запуску, ее установка не требуется.

## **4.8. Требования к транспортированию и хранению**

Требования к транспортированию и хранению программы соответствуют стандартным требованиям к транспортированию и хранению соответствующих электронных и бумажных носителей информации.

# **5. Требования к программной документации**

Состав программной документации должен включать в себя следующие компоненты:

1. Техническое задание (ГОСТ 19.201-78)
2. Программа и методика испытаний (ГОСТ 19.301-78)
3. Пояснительная записка (ГОСТ 19.404-79)
4. Руководство оператора (ГОСТ 19.505-79)
5. Текст программы (ГОСТ 19.401-78)

Вся документация должна быть составлена согласно ЕСПД (ГОСТ 19.101-77, 19.104-78, 19.105-78, 19.106-78 и ГОСТ к соответствующим документам (см. выше)) [3]. Вся документация сдаётся в печатном виде, с подписанными листами утверждения и в электронном виде в составе курсовой работы в систему LMS НИУ ВШЭ.

# **6. Технико-экономические показатели**

В настоящее время спрос на квалифицированных ИТ-специалистов на рынке труда оста­ётся высоким за счёт инте­грации информационных тех­нологий в различные области народного хозяйства и науки, информационного обслужива­ния, государственного управ­ления и образования, поэтому качественная подготовка буду­щих выпускников вузов в об­ласти информатики и инфор­мационных технологий (ИТ), способных применять совре­менные информационные тех­нологии для решения приклад­ных задач в профессиональной деятельности, является акту­альной проблемой.

Для успешной профессио­нальной деятельности будущих выпускников опреде­ляющее значение имеет изу­чение основ алгоритмизации и методов обработки структур данных, освоение которых позволит создать фундамент (базовые знания, умения и навыки) для формирования профессиональных компетен­ций выпускников. Обучение программированию может сы­грать решающую роль в улуч­шении когнитивных навыков студентов путём преподавания базовых алгоритмов и методов программирования, так как способствует развитию умений студентов составлять план для решения конкретной задачи, описывать в формализован­ном виде последовательность действий для достижения же­лаемого результата, прогнози­ровать результат выполнения спроектированных действий и вносить исправления в за­висимости от полученных ре­зультирующих данных. При составлении алгоритмов у сту­дентов вырабатываются навы­ки анализировать, сравнивать, обобщать, делать логические выводы, описывать план дея­тельности и излагать собствен­ные мысли в строгой логиче­ской последовательности.

Алгоритмизация являет­ся важным этапом в решении любой задачи с помощью ком­пьютера. Развитие алгоритми­ческого мышления является одной из основных задач при изучении программирования, так как даже простые алго­ритмы трудны для понима­ния первокурсникам, и поэ­тому, сложны для изучения. Практика обучения студентов показала, что изуче­ние языка программирования является «менее сложным» процессом, чем проектирова­ние алгоритмов.

Успешная деятельность специалиста в области ин­формационных технологий зависит не только от знания используемого им языка про­граммирования при разработ­ке приложений для решения прикладных задач, но и, в большей степени, зависит от «богатства» знаний в области построения хорошо структури­рованных алгоритмов обработ­ки информации, представляе­мой различными структурами данных, и накопленного опыта при решении приклад­ных задач.

При обучении программи­рованию студенты довольно ча­сто испытывают значительные трудности при поиске ответа на вопрос, почему данный ал­горитм подходит для решения задачи и позволяет получить необходимые результирующие данные. Для обоснования кор­ректности алгоритма без осоз­нания студентами свойств объ­екта, выделения логических связей между отдельными объ­ектами и отвлечения от дру­гих несущественных свойств и связей довольно сложно. На­пример, обладание знаниями о структуре цикла со счётчиком, не означает наличие умения на практике построить алгоритм для вычисления среднемесяч­ной температуры.

# **7. Стадии и этапы разработки**

## **7.1. Необходимые стадии разработки, этапы и содержание работ**

1. Техническое задание
   1. Обоснование необходимости разработки программы
      1. Постановка задачи
      2. Выбор и обоснование критериев эффективности и качества разрабатываемой программы
   2. Научно-исследовательские работы
      1. Определение структуры входных и выходных данных
      2. Предварительный выбор методов решения задачи
   3. Разработка и утверждение технического задания
      1. Определение требований к программе
      2. Определение стадий, этапов и сроков разработки программы и документации на неё
      3. Согласование и утверждение технического задания
2. Технический проект

2.1 Разработка технического проекта

2.1.1 Уточнение структуры входных и выходных данных

2.1.2 Разработка алгоритма решения задачи

2.1.3 Определение формы представления входных и выходных данных

2.1.4 Разработка структуры программы

2.2 Утверждение технического проекта

2.2.1Разработка пояснительной записки

* + 1. Согласование и утверждение технического проекта

1. Рабочий проект
   1. Разработка программы
      1. Программирование и отладка программы
   2. Разработка программных документов
      1. Разработка программных документов в соответствии с требованиями ГОСТ 19.101-77
   3. Испытания программы
      1. Разработка, согласование и утверждение программы и методики испытаний
      2. Корректировка программы и программной документации по результатам испытаний
2. Внедрение

4.1. Подготовка и передача программы

## **7.2. Сроки и исполнители**

Программа и документация к ней разрабатываются к утвержденным срокам защиты курсовой работы (20 – 30 мая 2020 года).

Исполнителем является студент НИУ ВШЭ группы БПИ183 Капур Даниэл Сингх

# **8. Порядок контроля и приемки**

## **8.1. Виды испытаний**

Виды испытаний описаны в документе «Программа и методика испытаний»(ГОСТ 19.301-78).

## **8.2. Общие требования к приемке работы**

Общие требования к приемке работы описаны в документе «Программа и методика испытаний»(ГОСТ19.301-78).

**9. Список использованной литературы**

1. ГОСТ 19.101-77 Виды программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
2. ГОСТ 19.102-77 Стадии разработки. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
3. ГОСТ 19.103-77 Обозначения программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
4. ГОСТ 19.104-78 Основные надписи. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
5. ГОСТ 19.105-78 Общие требования к программным документам. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
6. ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом.

//Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

1. ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
2. ГОСТ 19.603-78 Общие правила внесения изменений. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
3. ГОСТ 19.604-78 Правила внесения изменений в программные документы, выполненные печатным способом. //Единая система программной документации. –М.: ИПК Издательство стандартов, 2001.

части воздействия климатических факторов внешней среды. – М.: Изд-во стандартов, 1997.

1. ГОСТ 19.301-79 Программа и методика испытаний. Требования к содержанию и оформлению.

//Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

11)Управление рисками при внедрении ИТ проектов // [Электронный ресурс]

Режим доступа: http://www.econf.rae.ru/pdf/2007/10/Pesotskaya.pdf

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Изм. | Номера листов | | | | Всего листов (страниц) в документе | № документа | Входя- щий № сопро- водит. докум. и дата | Под- пись | Дата |
| изменен- ных | заменен- ных | новых | аннулиро- ванных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |