­Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Основы алгоритмизации и программирования (ОАиП)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту на тему

Программное средство «Калькулятор»

БГУИР КП I-40 01 01 406 ПЗ

Выполнил

студент: гр. 151004 Иванов И.И.

Проверил: Фадеева Е.П.

Минск 2022

содержание

[Введение 5](#_Toc101597395)

[1 Аналитический обзор литературы 6](#_Toc101597396)

[1.1 Обзор существующих аналогов 6](#_Toc101597397)

[1.1.1 Стандартный калькулятор Windows 6](#_Toc101597398)

[1.1.2 Онлайн-калькулятор Google 7](#_Toc101597399)

[1.1.3 Онлайн-калькулятор «okcalc» 8](#_Toc101597400)

[1.1.4 Онлайн-калькулятор «calculator888» 9](#_Toc101597401)

[1.1.5 Онлайн-калькулятор «kontrolnaya-rabota» 10](#_Toc101597402)

[1.1.6 Онлайн-калькулятор «kalkulyatoronlajn» 11](#_Toc101597403)

[1.2 Анализ методов и способов разработки 12](#_Toc101597404)

[1.2.1 Связный список 12](#_Toc101597405)

[1.2.2 Работа с текстовым файлом 14](#_Toc101597406)

[2 Постановка задачи 17](#_Toc101597407)

[2.1 Назначение разработки 17](#_Toc101597408)

[2.2 Перечень функциональных требований 17](#_Toc101597409)

[2.3 Входные и выходные параметры 17](#_Toc101597410)

[2.4 Состав и параметры технических и программных средств 18](#_Toc101597411)

[3 Разработка программного средства 19](#_Toc101597412)

[3.1 Описание алгоритмов 19](#_Toc101597413)

[3.2 Структура данных 27](#_Toc101597414)

[3.2.1 Структура типов программы 27](#_Toc101597415)

[3.2.2 Структура данных программы 28](#_Toc101597416)

[3.2.3 Структура данных алгоритма SaveData 29](#_Toc101597417)

[3.2.4 Структура данных алгоритма btn0Click 29](#_Toc101597418)

[3.2.5 Структура данных алгоритма ConvertSF 30](#_Toc101597419)

[3.2.6 Структура данных алгоритма btnEqClick 30](#_Toc101597420)

[3.2.7 Структура данных алгоритма ZeroException 30](#_Toc101597421)

[3.2.8 Структура данных алгоритма OneException 30](#_Toc101597422)

[3.2.9 Структура данных алгоритма NoException 31](#_Toc101597423)

[3.2.10 Структура данных алгоритма FactException 31](#_Toc101597424)

[3.2.11 Структура данных алгоритма LimException 31](#_Toc101597425)

[3.2.12 Структура данных алгоритма NegException 31](#_Toc101597426)

[3.2.13 Структура данных алгоритма TwoNumbers 32](#_Toc101597427)

[3.2.14 Структура данных алгоритма PlaceSymbol 32](#_Toc101597428)

[3.2.15 Структура данных алгоритма btnClearClick 32](#_Toc101597429)

[3.2.16 Структура данных алгоритма btnHistClick 32](#_Toc101597430)

[3.2.17 Структура данных алгоритма btnBulkClick 33](#_Toc101597431)

[3.2.18 Структура данных алгоритма btn1Click 33](#_Toc101597432)

[3.2.19 Структура данных алгоритма btn2Click 33](#_Toc101597433)

[3.2.20 Структура данных алгоритма btn3Click 34](#_Toc101597434)

[3.2.21 Структура данных алгоритма btn4Click 34](#_Toc101597435)

[3.2.22 Структура данных алгоритма btn5Click 34](#_Toc101597436)

[3.2.23 Структура данных алгоритма btn6Click 34](#_Toc101597437)

[3.2.24 Структура данных алгоритма btnVersinClick 34](#_Toc101597438)

[3.2.25 Структура данных алгоритма btn7Click 34](#_Toc101597439)

[3.2.26 Структура данных алгоритма btn8Click 35](#_Toc101597440)

[3.2.27 Структура данных алгоритма btn9Click 35](#_Toc101597441)

[3.2.28 Структура данных алгоритма btnCommaClick 35](#_Toc101597442)

[3.2.29 Структура данных алгоритма btnSinClick 35](#_Toc101597443)

[3.2.30 Структура данных алгоритма btnCosClick 35](#_Toc101597444)

[3.2.31 Структура данных алгоритма btnTgClick 35](#_Toc101597445)

[3.2.32 Структура данных алгоритма btnCtgClick 36](#_Toc101597446)

[3.2.33 Структура данных алгоритма btnAsinClick 36](#_Toc101597447)

[3.2.34 Структура данных алгоритма btnAcosClick 36](#_Toc101597448)

[3.2.35 Структура данных алгоритма btnAtgClick 36](#_Toc101597449)

[3.2.36 Структура данных алгоритма btnActgClick 36](#_Toc101597450)

[3.2.37 Структура данных алгоритма btn10PowerClick 36](#_Toc101597451)

[3.2.38 Структура данных алгоритма btn2PowerClick 37](#_Toc101597452)

[3.2.39 Структура данных алгоритма btnAcscClick 37](#_Toc101597453)

[3.2.40 Структура данных алгоритма btnAscClick 37](#_Toc101597454)

[3.2.41 Структура данных алгоритма btnBackClick 37](#_Toc101597455)

[3.2.42 Структура данных алгоритма btnChClick 37](#_Toc101597456)

[3.2.43 Структура данных алгоритма btnClearAllClick 37](#_Toc101597457)

[3.2.44 Структура данных алгоритма btnCscClick 38](#_Toc101597458)

[3.2.45 Структура данных алгоритма btnCschClick 38](#_Toc101597459)

[3.2.46 Структура данных алгоритма btnCthClick 38](#_Toc101597460)

[3.2.47 Структура данных алгоритма btnDfactorialClick 38](#_Toc101597461)

[3.2.48 Структура данных алгоритма btnDivideClick 38](#_Toc101597462)

[3.2.49 Структура данных алгоритма btnEilerClick 38](#_Toc101597463)

[3.2.50 Структура данных алгоритма btnExscClick 39](#_Toc101597464)

[3.2.51 Структура данных алгоритма btnExpClick 39](#_Toc101597465)

[3.2.52 Структура данных алгоритма btnExcscClick 39](#_Toc101597466)

[3.2.53 Структура данных алгоритма btnFactorialClick 39](#_Toc101597467)

[3.2.54 Структура данных алгоритма btnFloatClick 39](#_Toc101597468)

[3.2.55 Структура данных алгоритма btnHavercosClick 39](#_Toc101597469)

[3.2.56 Структура данных алгоритма btnHaversinClick 40](#_Toc101597470)

[3.2.57 Структура данных алгоритма btnLgClick 40](#_Toc101597471)

[3.2.58 Структура данных алгоритма btnLnClick 40](#_Toc101597472)

[3.2.59 С Структура данных алгоритма btnVercosClick 40](#_Toc101597473)

[3.2.60 труктура данных алгоритма btnMinusClick 40](#_Toc101597474)

[3.2.61 Структура данных алгоритма btnMultipleClick 40](#_Toc101597475)

[3.2.62 Структура данных алгоритма btnPeeClick 41](#_Toc101597476)

[3.2.63 Структура данных алгоритма btnPercentClick 41](#_Toc101597477)

[3.2.64 Структура данных алгоритма btnPlusClick 41](#_Toc101597478)

[3.2.65 Структура данных алгоритма btnPosNegClick 41](#_Toc101597479)

[3.2.66 Структура данных алгоритма btnPower2Click 41](#_Toc101597480)

[3.2.67 Структура данных алгоритма btnPower3Click 41](#_Toc101597481)

[3.2.68 Структура данных алгоритма btnPowerYClick 42](#_Toc101597482)

[3.2.69 Структура данных алгоритма btnScClick 42](#_Toc101597483)

[3.2.70 Структура данных алгоритма btnSchClick 42](#_Toc101597484)

[3.2.71 Структура данных алгоритма btnShClick 42](#_Toc101597485)

[3.2.72 Структура данных алгоритма btnSqrtClick 42](#_Toc101597486)

[3.2.73 Структура данных алгоритма btnThClick 42](#_Toc101597487)

[3.2.74 Структура данных алгоритма btnTrigClick 43](#_Toc101597488)

[3.2.75 Структура данных алгоритма btnClearFileClick 43](#_Toc101597489)

[3.2.76 Структура данных алгоритма Calculate 43](#_Toc101597490)

[3.2.77 Структура данных алгоритма Display 43](#_Toc101597491)

[3.2.78 Структура данных алгоритма ResetData 43](#_Toc101597492)

[3.3 Схема алгоритмов решения задачи по ГОСТ 19.701-90 43](#_Toc101597493)

[3.4 Графический интерфейс 44](#_Toc101597494)

[3.4.1 Описание графических компонентов формы FormMain 44](#_Toc101597495)

[3.4.2 Описание графических компонентов формы FormHist 46](#_Toc101597496)

[3.4.3 Описание графических компонентов формы FormTrig 47](#_Toc101597497)

[4 Тестирование программного средства 49](#_Toc101597498)

[5 Руководство по установке и использованию программного средства 50](#_Toc101597499)

[Заключение 51](#_Toc101597500)

[Список использованной литературы 52](#_Toc101597501)

# Введение

# Аналитический обзор литературы

## Обзор существующих аналогов

### Стандартный калькулятор Windows

Программное средство, изображённое на рисунке 1, целесообразно рассматривать как один из ведущих аналогов, поскольку оно поставляется вместе с операционной системой Windows.



Рисунок 1

Список выполняемых функций:

* ввод чисел;
* возможность редактирования и очистки введённого текста;
* выполнение различных операций над числами;
* работа с целыми и дробными числами;
* отображение истории действий.

Рассмотрим достоинства данного приложения:

* выполнение нескольких операций одновременно в соответствии с приоритетом;
* возможность работать как с градусами, так и с радианами;
* наличие прямых тригонометрических и гиперболических функций;
* наличие автоматической очистки поля при вводе нового числа.

Рассмотрим недостатки данного приложения:

* отсутствие некоторых прямых тригонометрических функций, таких как синус-верзус, косинус-верзус, гаверсинус, гаверкосинус, эксеканс, экскосеканс;
* отсутствие обратных тригонометрических функций;
* отсутствие быстрых степенных функций и двойного факториала;
* интуитивно сложный для понимания режим ввода чисел с плавающей точкой;
* неверное название десятичного логарифма, вводящее пользователя в заблуждение.

### Онлайн-калькулятор Google

Программное средство, изображённое на рисунке 2, целесообразно рассматривать как один из ведущих аналогов, поскольку оно встроено в поисковую страницу Google и показывается пользователю, когда он вводит арифметический запрос в поисковую строку.



Рисунок 2

Список выполняемых функций:

* Ввод чисел;
* Возможность редактирования и очистки введённого текста;
* Выполнение различных операций над числами;
* Работа с целыми и дробными числами;
* Отображение истории действий.

Рассмотрим достоинства данного приложения:

* Выполнение нескольких операций одновременно в соответствии с приоритетом;
* Возможность работать как с градусами, так и с радианами;
* Наличие прямых тригонометрических функций;
* Наличие автоматической очистки поля при вводе нового числа.

Рассмотрим недостатки данного приложения:

* отсутствие некоторых прямых тригонометрических функций, таких как синус-верзус, косинус-верзус, гаверсинус, гаверкосинус, эксеканс, экскосеканс;
* отсутствие обратных тригонометрических функций;
* отсутствие гиперболических функций;
* отсутствие быстрых степенных функций и двойного факториала;
* отсутствие ввода чисел с плавающей точкой;
* отсутствие очистки целого поля, необходимо стирать по одному символу;
* неверное название десятичного логарифма, вводящее пользователя в заблуждение.

### Онлайн-калькулятор «okcalc»

Программное средство, изображённое на рисунке 3, целесообразно рассматривать как один из ведущих аналогов, поскольку оно занимает лидирующее положение в поиске Google по запросу “инженерный калькулятор онлайн”.



Рисунок 3

Список выполняемых функций:

* ввод чисел;
* возможность редактирования и полной очистки введённого текста;
* выполнение арифметических операций;
* работа с целыми и дробными числами;

Рассмотрим достоинства данного приложения:

* выполнение нескольких операций одновременно в соответствии с приоритетом;
* наличие быстрых степенных функций.

Рассмотрим недостатки данного приложения:

* отсутствие прямых тригонометрических функций;
* отсутствие обратных тригонометрических функций;
* отсутствие гиперболических функций;
* отсутствие факториалов;
* отсутствие ввода чисел с плавающей точкой;
* отсутствие автоматической очистки поля при вводе нового числа;
* неверное название десятичного логарифма, вводящее пользователя в заблуждение;
* отсутствие истории действий.

### Онлайн-калькулятор «calculator888»

Программное средство, изображённое на рисунке 4 целесообразно рассматривать как один из ведущих аналогов, поскольку оно появляется на первой странице поиска Google по запросу “инженерный калькулятор онлайн”.



Рисунок 4

Список выполняемых функций:

* ввод чисел;
* возможность редактирования и полной очистки введённого текста;
* выполнение арифметических операций;
* работа с целыми и дробными числами;

Рассмотрим достоинства данного приложения:

* выполнение нескольких операций одновременно в соответствии с приоритетом;
* наличие быстрых степенных функций.
* наличие некоторых прямых и обратных тригонометрических функций.

Рассмотрим недостатки данного приложения:

* отсутствие некоторых прямых тригонометрических функций;
* отсутствие некоторых обратных тригонометрических функций;
* отсутствие гиперболических функций;
* отсутствие факториалов;
* отсутствие ввода чисел с плавающей точкой;
* отсутствие истории действий.

Очевидно, что стандартный калькулятор Windows имеет наименьшее количество недостатков, поэтому он будет являться ориентиром при разработке собственного программного средства.

### Онлайн-калькулятор «kontrolnaya-rabota»

Программное средство, изображённое на рисунке 5, целесообразно рассматривать как один из ведущих аналогов, поскольку оно появляется на первой странице поиска Google по запросу “инженерный калькулятор онлайн”.

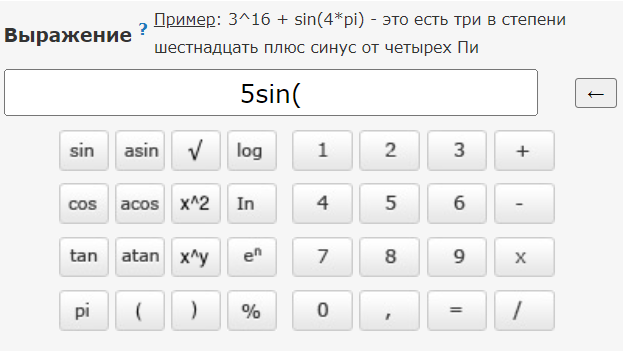


Рисунок 5

Список выполняемых функций:

* ввод чисел;
* возможность редактирования и полной очистки введённого текста;
* выполнение арифметических операций;
* работа с целыми и дробными числами;

Рассмотрим достоинства данного приложения:

* выполнение нескольких операций одновременно в соответствии с приоритетом;
* наличие быстрых степенных функций.
* наличие некоторых прямых и обратных тригонометрических функций;
* наличие режима градусов и радианов.

Рассмотрим недостатки данного приложения:

* отсутствие некоторых прямых тригонометрических функций;
* отсутствие некоторых обратных тригонометрических функций;
* отсутствие гиперболических функций;
* отсутствие факториалов;
* отсутствие ввода чисел с плавающей точкой;
* отсутствие истории действий.
* неудобная реализация операций над одним числом, требующая обязательного закрытия скобки

### Онлайн-калькулятор «kalkulyatoronlajn»

Программное средство, изображённое на рисунке 6, целесообразно рассматривать как один из ведущих аналогов, поскольку оно появляется на первой странице поиска Google по запросу “инженерный калькулятор онлайн”.



Рисунок 6

Список выполняемых функций:

* ввод чисел;
* возможность редактирования и полной очистки введённого текста;
* выполнение арифметических операций;
* работа с целыми и дробными числами;

Рассмотрим достоинства данного приложения:

* выполнение нескольких операций одновременно в соответствии с приоритетом;
* наличие быстрых степенных функций.
* наличие некоторых прямых тригонометрических функций;

Рассмотрим недостатки данного приложения:

* отсутствие некоторых прямых тригонометрических функций;
* отсутствие обратных тригонометрических функций;
* отсутствие гиперболических функций;
* отсутствие факториалов;
* отсутствие ввода чисел с плавающей точкой;
* отсутствие истории действий.
* неудобная реализация операций над одним числом, требующая обязательного закрытия скобки

Очевидно, что стандартный калькулятор Windows имеет наименьшее количество недостатков, поэтому он будет являться ориентиром при разработке собственного программного средства.

## Анализ методов и способов разработки

Предполагается, что данное средство, помимо выполнения арифметических и тригонометрических операций над числами, будет выполнять ряд дополнительных функций, таких как сохранение истории действий в текстовый файл, чтение истории действий из текстового файла, чтение и массовая операция над всеми числами из текстового файла.

В данном программном средстве будут использоваться пять основных структур данных:

* двунаправленные связные списки;
* файлы: текстовые и динамические.

### Связный список

Связный список — некоторая последовательность элементов, которые связаны друг с другом логически. Логический порядок прохождения элементов определяется с помощью ссылок, при этом он может не совпадать с физическим порядком размещения элементов в памяти компьютера.

Доступ к элементам списка осуществляется последовательно, т. е. чем дальше в структуре расположен элемент, тем дольше к нему по времени будет осуществляться доступ.

Список состоит из узлов (англ. nodes). Каждый узел включает две части: информационную (непосредственные данные, принадлежащие элементу) и ссылочную (указатель/ссылка на следующий и/или предыдущий узел).

В односвязном, или однонаправленном связном, списке (рис. 7) каждый узел содержит ссылку на следующий узел. Для последнего узла эта ссылка обычно является нулевой.



Рисунок 7

По односвязному списку можно передвигаться только в сторону конца списка. Узнать адрес предыдущего элемента, опираясь на содержимое текущего узла, невозможно.

В двусвязном, или двунаправленном связном, списке (рис. 8) ссылки в каждом узле указывают на предыдущий и на последующий узел.

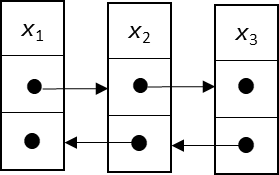


Рисунок 8

Как и односвязный список, двусвязный допускает только последовательный доступ к элементам, но при этом даёт возможность перемещения в обе стороны. В таком списке проще производить удаление и перестановку элементов, так как легко получить доступ ко всем элементам списка, ссылки которых направлены на изменяемый элемент.

При работе со списком вводятся дополнительные ссылки на первый и последний элемент списка. Будем называть их head («голова») и tail («хвост»).

Чаще всего узлы списка размещают в динамической памяти, при этом в качестве значений ссылок используются адреса узлов (рис. 9).

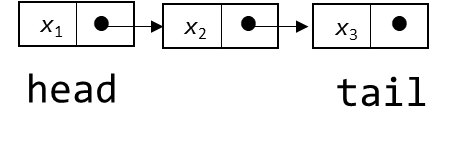


Рисунок 9

Альтернативный способ — использовать для хранения информации обычные массивы (рис. 10), тогда в качестве значений ссылок будут выступать индексы (порядковые номера элементов массива).

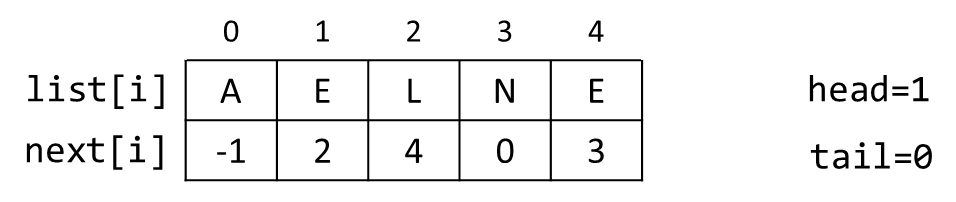


Рисунок 10

Операции вставки в конкретное место списка и удаления определённого элемента списка выполняются за O(1) при условии, что на вход даётся ссылка на узел (идущий перед точкой вставки или предшествующий узлу, который будет удалён). Если такая ссылка не предоставлена, то операции работают за O(n).

В то же время вставка в произвольное место динамического массива требует перемещения в среднем половины элементов, а в худшем случае — всех элементов. Хотя можно «удалить» элемент из массива за константное время, пометив его ячейку как «свободную», это вызовет фрагментацию, которая будет негативно влиять на скорость прохода по массиву.

В связный список может быть вставлено произвольное количество элементов, ограниченное только доступной памятью. Ранее вставленные элементы никуда не перемещаются, их адреса в памяти не меняются. В динамических массивах при вставке иногда происходит реаллокация; это дорогостоящая операция, которая может оказаться невозможной при высокой фрагментированности памяти (не удастся найти непрерывный блок памяти нужного размера, хотя небольшие свободные блоки будут доступны в достаточном количестве).

### Работа с текстовым файлом

Файловый тип — это произвольная последовательность элементов, длина которой заранее не определена, а конкретизируется в процессе выполнения программы. Это определение логического файла, т.е. того, который используется в программе (файл с точки зрения программиста). Физический файл (набор данных) — это именованная область памяти на внешнем носителе, в которой хранится некоторая информация (файл с точки зрения пользователя).

B Delphi существует два типа файлов:

* типизированные файлы;
* текстовые файлы;

Типизированные файлы связываются с файловыми переменными, объявленными как file of <Тип>. Файл считается состоящим из элементов, каждый из которых имеет тип «Тип>.

Текстовый файл представляет собой последовательность символов, однако он не эквивалентен файлу типа file of Char. Файл типа file of Char — это единая последовательность символов. Текстовые файлы связываются с файловыми переменными, принадлежащими стандартному типу TextFile.

Особенность текстовых файлов состоит в том, что содержащиеся в них символы разбиваются на строки. Строки могут быть различной длины (пустые в том числе). В конце каждой строки помещается специальный управляющий символ: возврат каретки (#13 или ^М - международное обозначение CR) и перехода новую строку (#10 или ^J - международное обозначение LF). С наличием этого маркера связана логическая функция Eoln (End of line).

Для доступа к отдельным элементам файлов в Delphi существуют специальные стандартные процедуры и функции. Их называют процедурами и функциями ввода-вывода. Обращение к ним осуществляется обычным образом.

* AssignFile: устанавливает ассоциативную связь между файловой переменной и внешним файлом;
* BlockRead: читает несколько записей из файла в переменную;
* BlockWrite: записывает несколько записей в файл;
* CloseFile: разрывает ассоциативную связь между файловой переменной и файлом;
* DeleteFile: удаляет файл с диска;
* DiskFree: определяет количество свободного места на диске;
* DiskSize: определяет размер указанного диска;
* Eof: определяет, стоит ли указатель в конце файла;
* Erase: удаляет файл, связанный с файловой переменной;
* DiskSize: определяет размер указанного диска;
* FileExists: проверяет, существует ли указанный файл;
* FileGetAttr: возвращает атрибуты указанного файла;
* FileSearch: производит поиск файла в нескольких заданных каталогах;
* FileSetAttr: устанавливает атрибуты файла;
* FindClose: высвобождает память, выделенную процедурой FindFirst;
* FindFirst: находит файл с заданным набором атрибутов по заданной маске в указанном каталоге;
* FindNext: производит повторный поиск файла;
* IOResult: возвращает статус ошибки последней выполненной операции ввода/вывода;
* Read: читает данные из файла;
* Rename: переименовывает файл;
* RenameFile: переименовывает файл;
* Reset: открывает существующий файл;
* Rewrite: создает новый файл и открывает его;
* Seek: устанавливает файловый указатель в заданную позицию;
* Write: записывает данные в файл.

Некоторые операции неприменимы к текстовым файлам:

* FilePos: возвращает текущую позицию файлового указателя;
* FileSize: возвращает размер файла в байтах, а для файлов типа Record - количество записей;
* Truncate: удаляет все записи в файле, находящиеся после текущей позиции.

# Постановка задачи

## Назначение разработки

В век информационных технологий самым популярным направлением является IT. Множество людей занимается созданием разнообразных приложений на различные устройства для упрощения жизни. И на сегодняшний день сложно представить жизнь без смартфонов, компьютеров, телевизоров и так далее.

Все эти достижения науки и техники невозможны без математики и физики. Инженеры-конструкторы используют редкие функции и проводят сложные вычисления, и каждая ошибка в них чревата поломками в будущем устройстве.

К сожалению, большинство инженерных калькуляторов не имеют всех необходимых для инженера функций. Поэтому было решено разработать такой калькулятор, который будет иметь все, даже самые редкие, функции, которые могут понадобиться в работе. Также необходимо наличие популярных частных случаев функций для экономии времени инженеров.

## Перечень функциональных требований

После анализа аналогов был составлен следующий список выполняемых функций:

1. Ввод чисел.

* ввод обычных чисел;
* ввод чисел с плавающей точкой.

1. Операции над одним числом.

* прямые и обратные тригонометрические функции, гиперболические функции;
* факториалы;
* быстрые степенные функции.

1. Операции над двумя числами.

* арифметические функции.

1. Операции над многими числами.

* перемножение и суммирование всех чисел из текстового файла.

## Входные и выходные параметры

Пользователь выбирает одну из функций, которую должна сделать программа (ввести число, выполнить операцию, загрузить данные из файла).

Результат будет представляться в зависимости от выбранной функции: отображение результата выражения, сохранение истории в файл.

## Состав и параметры технических и программных средств

Приложение должно функционировать на персональных компьютерах со следующими характеристиками:

* процессор Intel 80486DX 64MHz или лучше;
* оперативная память 16 MB 100 MHz SDRAM или лучше;
* накопитель HDD или SSD объемом 1 GB или больше.

Приложение «Инженерный калькулятор» должно функционировать в окружении операционной системы Windows 95 или новее.

В данном разделе указаны минимальные технические требования для запуска программного средства. Для эксплуатации в реальных могут потребоваться более мощные технические средства. Программное средство должно корректно функционировать на более мощном оборудовании.

# Разработка программного средства

## Описание алгоритмов

В таблице 1 представлены краткие описания алгоритмов, используемых для реализации программы.

Таблица 1 – Описание алгоритмов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N | Наименование алгоритма | Назначение алгоритма | Формальные параметры | Предполагаемый тип  реализации |
| 1 | Инициализация формы «Hummel009's Calculator» | Загрузка основного окна программы, инициализация изначальных данных и флагов |  |  |
| 2 | Инициализация формы «Trigonometry» | Загрузка окна программы со всеми тригонометрическими функциями |  |  |
| 3 | Инициализация формы «History» | Загрузка окна программы с историей действий |  |  |
| 4 | ConvertSF  (FInp) | Преобразование строки в число с формированием результата в Result, обновление флага ошибки | FInp - получает значение от фактического параметра | Функция;  Result – возвращаемый параметр |
| 5 | btnEqClick  (Sender) | Считывание второго операнда операций над двумя операндами, вызов операции Display() | Sender – получает адрес от фактического параметра | Процедура |
| 6 | ZeroException  (FOp) | Выполнение тригонометрических функций, имеющих разрыв в нуле | FOp – получает защищённый адрес от фактического параметра | Процедура |
| 7 | OneException  (FOp) | Выполнение тригонометрических функций с областью определения от -1 до 1 | FOp – получает защищённый адрес от фактического параметра | Процедура |

Продолжение Таблицы 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N | Наименование алгоритма | Назначение алгоритма | Формальные параметры | Предполагаемый тип  реализации |
| 8 | NoException  (FOp) | Выполнение тригонометрических функций с бесконечной областью определения | FOp – получает защищённый адрес от фактического параметра | Проце-дура |
| 9 | FactException  (FOp; FInt) | Вычисление факториала либо, если введён неверный факториал, обновление флага ошибки | FOp, FInt – получают защищённый адрес от фактического параметра | Проце-дура |
| 10 | LimException  (FOp; FInt) | Вычисление результата операции над одним операндом либо, если превышен лимит Integer, обновление флага ошибки | FOp, FInt – получают защищённый адрес от фактического параметра | Проце-дура |
| 11 | NegException  (FOp) | Вычисление результата операции над одним операндом либо, если введено отрицательное число, обновление флага ошибки | FOp – получает защищённый адрес от фактического параметра | Проце-дура |
| 12 | TwoNumbers  (FOp) | Считывание первого операнда и сохранение операции, требующей ввод второго операнда | FOp – получает защищённый адрес от фактического параметра | Процедура |
| 13 | PlaceSymbol  (FSym) | Добавление символа на экран с различными проверками, блокирующими добавление символа или очищающими экран при такой надобности. | FSym – получает защищённый адрес от фактического параметра | Процедура |

Продолжение Таблицы 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N | Наименование алгоритма | Назначение алгоритма | Формальные параметры | Предпо-лагаемый тип  реализа-ции |
| 14 | btnClearClick  (Sender) | Удаление одного символа с конца, либо, если на экране было сообщение об ошибке, очистка экрана | Sender – получает адрес от фактического параметра | Процедура |
| 15 | btnHistClick (Sender) | Открытие формы «History» и загрузка данных из типизированного файла | Sender – получает адрес от фактического параметра | Проце-дура |
| 16 | btnBulkClick  (Sender) | Загрузка чисел из текстового файла и их суммирование/перемножение в зависимости от присвоенной ранее операции | Sender – получает адрес от фактического параметра | Проце-дура |
| 17 | btnClearFileClick (Sender) | Очистка истории в поле вывода и типизированном файле, закрытие формы | Sender – получает адрес от фактического параметра | Проце-дура |
| 18 | Calculate (FRes) | Выбор формулы и вычисление результата в зависимости от присвоенной ранее операции | FRes – получает значение от фактического параметра | Проце-дура |
| 19 | ResetData (FMem) | Зануление результатов и данных, возвращение операции и флага ошибки к изначальному состоянию | FMem – получает значение от фактического параметра | Проце-дура |
| 20 | SaveData() | Запись истории вычислений в типизированный файл | Отсутствуют | Проце-дура |
| 21 | Display() | Отображение результата или сообщения об ошибке на экране | Отсутствуют | Проце-дура |

Продолжение Таблицы 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N | Наименование алгоритма | Назначение алгоритма | Формальные параметры | Предпола-гаемый тип  реализации |
| 22 | btn0Click (Sender) | Вывод числа на экран | Sender – получает адрес от фактического параметра | Проце-дура |
| 23 | btnVersinClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Проце-дура |
| 25 | btn1Click (Sender) | Вывод числа на экран | Sender – получает адрес от фактического параметра | Процедура |
| 26 | btn2Click (Sender) | Вывод числа на экран | Sender – получает адрес от фактического параметра | Процедура |
| 27 | btn2PowerClick (Sender) | Вывод числа на экран | Sender – получает адрес от фактического параметра | Процедура |
| 28 | btn3Click (Sender) | Вывод числа на экран | Sender – получает адрес от фактического параметра | Процедура |
| 29 | btn4Click (Sender) | Вывод числа на экран | Sender – получает адрес от фактического параметра | Процедура |
| 30 | btn5Click (Sender) | Вывод числа на экран | Sender – получает адрес от фактического параметра | Процедура |
| 31 | btn6Click (Sender) | Вывод числа на экран | Sender – получает адрес от фактического параметра | Процедура |
| 32 | btn7Click (Sender) | Вывод числа на экран | Sender – получает адрес от фактического параметра | Процедура |
| 33 | btn8Click (Sender) | Вывод числа на экран | Sender – получает адрес от фактического параметра | Процедура |
| 34 | btn9Click (Sender) | Вывод числа на экран | Sender – получает адрес от фактического параметра | Процедура |

Продолжение Таблицы 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N | Наименование алгоритма | Назначение алгоритма | Формальные параметры | Предпола-гаемый тип  реализации |
| 35 | btnAcosClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 36 | btnAcscClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 37 | btnActgClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 38 | btnAscClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 39 | btnAsinClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 40 | btnAtgClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 41 | btnBackClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 42 | btnChClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 43 | btnClearAllClick (Sender) | Очистка экрана | Sender – получает адрес от фактического параметра | Процедура |
| 44 | btnCommaClick (Sender) | Вывод точки на экран | Sender – получает адрес от фактического параметра | Процедура |

Продолжение Таблицы 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N | Наименование алгоритма | Назначение алгоритма | Формальные параметры | Предпола-гаемый тип  реализации |
| 45 | btnCosClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 46 | btnCscClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 47 | btnCschClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 48 | btnCtgClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 49 | btnCthClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 50 | btnDFactorialClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 51 | btnDivideClick (Sender) | Выполнение операции над двумя операндами | Sender – получает адрес от фактического параметра | Процедура |
| 52 | btnEilerClick (Sender) | Вывод числа Эйлера на экран | Sender – получает адрес от фактического параметра | Процедура |
| 53 | btnExcscClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 54 | btnExpClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |

Продолжение Таблицы 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N | Наименование алгоритма | Назначение алгоритма | Формальные параметры | Предпола-гаемый тип  реализации |
| 55 | btnExscClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 56 | btnFactorialClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 57 | btnFloatClick (Sender) | Ввод числа с плавающей точкой | Sender – получает адрес от фактического параметра | Процедура |
| 58 | btnHavercosClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 59 | btnHaversinClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 60 | btnLgClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 61 | btnLnClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 62 | btnVercosClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Процедура |
| 63 | btnMinusClick (Sender) | Выполнение операции над двумя операндами | Sender – получает адрес от фактического параметра | Процедура |
| 64 | btnMultipleClick (Sender) | Выполнение операции над двумя операндами | Sender – получает адрес от фактического параметра | Процедура |

Продолжение Таблицы 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N | Наименование алгоритма | Назначение алгоритма | Формальные параметры | Предпо-лагае-мый тип  реализации |
| 65 | btnPeeClick (Sender) | Вывод числа Пи на экран | Sender – получает адрес от фактического параметра | Проце-дура |
| 66 | btnPercentClick (Sender) | Выполнение операции над двумя операндами | Sender – получает адрес от фактического параметра | Проце-дура |
| 67 | btnPlusClick (Sender) | Выполнение операции над двумя операндами | Sender – получает адрес от фактического параметра | Проце-дура |
| 68 | btnPosNegClick (Sender) | Добавление символьного минуса на экран, чтобы иметь возможность вводить отрицательные числа | Sender – получает адрес от фактического параметра | Проце-дура |
| 69 | btnPower2Click (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Проце-дура |
| 70 | btnPower3Click (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Проце-дура |
| 71 | btnPowerYClick (Sender) | Выполнение операции над двумя операндами | Sender – получает адрес от фактического параметра | Проце-дура |
| 72 | btnScClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Проце-дура |
| 73 | btnSchClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Проце-дура |

Продолжение Таблицы 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N | Наименова-ние алгоритма | Назначение алгоритма | Формальные параметры | Предпола-гаемый тип  Реализа-ции |
| 74 | btnShClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Проце-дура |
| 75 | btnSinClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Проце-дура |
| 77 | btnTgClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Проце-дура |
| 78 | btnThClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Проце-дура |
| 79 | btnTrigClick (Sender) | Выполнение операции над одним операндом | Sender – получает адрес от фактического параметра | Проце-дура |

## Структура данных

### Структура типов программы

При разработке программного средства была использована динамическая структура данных «Линейный двунаправленный список»

Таблица 2 – Структура типов программы

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| TDisp | Record  Inp1, Inp2, Res, Op, Disp: String;  End; | Тип, предназначенный для хранения первого и второго операнда, а также результата в строковом виде для сохранения в типизированный файл |
| TMem | Record  Inp1, Inp2, Res: Real;  Op: TOp;  End; | Тип, предназначенный для хранения первого и второго операнда, а также результата |

Продолжение Таблицы 2

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| TOp | (ENULL, EARCCOS, EARCCTG, EARCSIN, EARCTG, ECOS, ECTG, EDIVIDE, EFACTORIAL, EMINUS, EMULTIPLE, EPERCENT, EPLUS, EPOWER, ESIN, ESQRT, ETG, ESQUARE, ECUBE,  ELG, ELN, ECH, ESH, ETH, ECTH, ETEN, EBACK, EDFACTORIAL, EEXP, ETWO, ESC, ECSC, EARCSC, EARCCSC, ESCH, ECSCH, EVERSIN, EVERCOS, EHAVERSIN, EHAVERCOS, EEXSC, EEXCSC) | Тип-перечисление, содержащий все виды операций, включая одну нейтральную (ENULL) |
| Tline | ^Eline | Тип, предназначенный для построения линейного двунаправленного списка, содержащего числа;  Data – число  Next, Prev – ссылки на следующее и предыдущее звено |
| Eline | Record  Data: Real;  Next, Prev: Tline;  End; |

### Структура данных программы

Таблица 3 – Структура данных программы

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| E | Real | Число Эйлера |
| P | Real | Число Пи |
| GOpView | Array[TOp] Of String | Внешний вид операции при выводе истории действий |
| GFact1 | Array[0..12] | Значения факториалов |
| GFact2 | Array[0..19] | Значения двойных факториалов |
|  |  |  |

Продолжение Таблицы 3

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| GHist1 | Set Of TOp | Операции над одним операндом, имеющие скобки при отображении истории действий |
| GHist2 | Set Of TOp | Операции над одним операндом, не имеющие скобки и расположенные справа при отображении истории действий |
| GHist3 | Set Of TOp | Операции над одним операндом, не имеющие скобки и расположенные слева при отображении истории действий |
| GDisp | TDisp | Сохранённые операнды, результат и операция в строковом виде для последующего отображения в истории действий |
| GMem | TMem | Сохранённые операнды, результат и операция |
| GLine | Integer | Текущий номер строки в истории действий |
| GError | Boolean | Переменная, предназначенная для вывода на экран сообщения об ошибке ввода |
| GClear | Boolean | Переменная, предназначенная для очистки экрана при вводе нового числа после завершения вычислений |

### Структура данных алгоритма SaveData

Таблица 4 – Структура данных программы SaveData()

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| LFile | File Of TDisp | Типизированный файл с историей | Локальный |

### Структура данных алгоритма btn0Click

Таблица 5 – Структура данных программы btn0Click(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма ConvertSF

Таблица 6 – Структура данных программы ConvertSF(FInp)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| FInp | String | Сохраняемая в историю строка | Формальный |
| LInp | Real | Конвертируемое число | Локальный |
| LLim | Integer | Верхний лимит степени числа с плавающей точкой | Локальный |
| LPower | String | Степень числа с плавающей точкой в строковом виде | Локальный |

### Структура данных алгоритма btnEqClick

Таблица 7 – Структура данных программы btnEqClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |
| LInp | String | Считываемое число | Локальный |

### Структура данных алгоритма ZeroException

Таблица 8 – Структура данных программы ZeroException(FOp)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| FOp | TOp | Присваиваемая операция | Формальный |
| LInp | String | Считываемая строка | Локальный |

### Структура данных алгоритма OneException

Таблица 9 – Структура данных программы OneException(FOp)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| FOp | TOp | Присваиваемая операция | Формальный |
| LInp | String | Считываемая строка | Локальный |

### Структура данных алгоритма NoException

Таблица 10 – Структура данных программы NoException(FOp)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| FOp | TOp | Присваиваемая операция | Формальный |
| LInp | String | Считываемая строка | Локальный |

### Структура данных алгоритма FactException

Таблица 11 – Структура данных программы FactException(FOp, FInt)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| FOp | TOp | Присваиваемая операция | Формальный |
| FInp | Integer | Ограничение ввода | Формальный |
| LInp | String | Считываемое число | Локальный |

### Структура данных алгоритма LimException

Таблица 12 – Структура данных программы LimException(FOp, FInt)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| FOp | TOp | Присваиваемая операция | Формальный |
| FInp | Extended | Ограничение ввода | Формальный |
| LInp | String | Считываемое число | Локальный |

### Структура данных алгоритма NegException

Таблица 13 – Структура данных программы NegException(FOp)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| FOp | TOp | Присваиваемая операция | Формальный |
| LInp | String | Считываемое число | Локальный |

### Структура данных алгоритма TwoNumbers

Таблица 14 – Структура данных программы TwoNumbers(FOp)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| FOp | TOp | Присваиваемая операция | Формальный |
| LInp | String | Считываемое число | Локальный |

### Структура данных алгоритма PlaceSymbol

Таблица 15 – Структура данных программы PlaceSymbol(FOp)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| FSym | String | Добавляемый символ | Формальный |
| LInp | String | Считываемая строка | Локальный |
| LAdd | String | Добавляемый символ | Локальный |

### Структура данных алгоритма btnClearClick

Таблица 16 – Структура данных программы btnClearClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |
| LInp | String | Считываемая строка | Локальный |
| LLen | Integer | Длина строки | Локальный |

### Структура данных алгоритма btnHistClick

Таблица 17 – Структура данных программы btnHistClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |
| LLine | TDisp | Запись из типизированного файла | Локальный |
| LFile | File Of TDisp | Типизированный файл с историей | Локальный |

### Структура данных алгоритма btnBulkClick

Таблица 18 – Структура данных программы btnBulkClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |
| LFile | TextFile | Количество строк в истории | Локальный |
| LData | String | Считываемая из файла строка | Локальный |
| LLine1, LLine2 | TLine | Ссылка на звено двунаправленного связного списка | Локальный |
| LRes | Real | Результат вычислений | Локальный |

### Структура данных алгоритма btn1Click

Таблица 19 – Структура данных программы btn1Click(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btn2Click

Таблица 20 – Структура данных программы btn2Click(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btn3Click

Таблица 21 – Структура данных программы btn3Click(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btn4Click

Таблица 22 – Структура данных программы btn4Click(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btn5Click

Таблица 23 – Структура данных программы btn5Click(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btn6Click

Таблица 24 – Структура данных программы btn6Click(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnVersinClick

Таблица 25 – Структура данных программы btnVersinClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btn7Click

Таблица 26 – Структура данных программы btn7Click(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btn8Click

Таблица 27 – Структура данных программы btn8Click(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btn9Click

Таблица 28 – Структура данных программы btn9Click(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnCommaClick

Таблица 29 – Структура данных программы btnCommaClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnSinClick

Таблица 30 – Структура данных программы btnSinClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnCosClick

Таблица 31 – Структура данных программы btnCosClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnTgClick

Таблица 32 – Структура данных программы btnTgClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnCtgClick

Таблица 33 – Структура данных программы btnCtgClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnAsinClick

Таблица 34 – Структура данных программы btnAsinClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnAcosClick

Таблица 35 – Структура данных программы btnAcosClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnAtgClick

Таблица 36 – Структура данных программы btnAtgClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnActgClick

Таблица 37 – Структура данных программы btnActgClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btn10PowerClick

Таблица 38 – Структура данных программы btn10PowerClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btn2PowerClick

Таблица 39 – Структура данных программы btn2PowerClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnAcscClick

Таблица 40 – Структура данных программы btnAcscClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnAscClick

Таблица 41 – Структура данных программы btnAscClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnBackClick

Таблица 42 – Структура данных программы btnBackClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnChClick

Таблица 43 – Структура данных программы btnChClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnClearAllClick

Таблица 44 – Структура данных программы btnClearAllClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnCscClick

Таблица 45 – Структура данных программы btnCscClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnCschClick

Таблица 46 – Структура данных программы btnCschClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnCthClick

Таблица 47 – Структура данных программы btnCthClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnDfactorialClick

Таблица 48 – Структура данных программы btnDfactorialClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnDivideClick

Таблица 49 – Структура данных программы btnDivideClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnEilerClick

Таблица 50 – Структура данных программы btnEilerClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnExscClick

Таблица 51 – Структура данных программы btnExscClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnExpClick

Таблица 52 – Структура данных программы btnExpClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnExcscClick

Таблица 53 – Структура данных программы btnExcscClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnFactorialClick

Таблица 54 – Структура данных программы btnFactorialClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnFloatClick

Таблица 55 – Структура данных программы btnFloatClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnHavercosClick

Таблица 56 – Структура данных программы btnHavercosClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnHaversinClick

Таблица 57 – Структура данных программы btnHaversinClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnLgClick

Таблица 58 – Структура данных программы btnLgClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnLnClick

Таблица 59 – Структура данных программы btnLnClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### С Структура данных алгоритма btnVercosClick

Таблица 60 – Структура данных программы btnVercosClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### труктура данных алгоритма btnMinusClick

Таблица 61 – Структура данных программы btnMinusClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnMultipleClick

Таблица 62 – Структура данных программы btnMultipleClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnPeeClick

Таблица 63 – Структура данных программы btnPeeClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnPercentClick

Таблица 64 – Структура данных программы btnPercentClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnPlusClick

Таблица 65 – Структура данных программы btnPlusClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnPosNegClick

Таблица 66 – Структура данных программы btnPosNegClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnPower2Click

Таблица 67 – Структура данных программы btnPower2Click(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnPower3Click

Таблица 68– Структура данных программы btnPower3Click(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnPowerYClick

Таблица 69 – Структура данных программы btnPowerYClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnScClick

Таблица 70 – Структура данных программы btnScClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnSchClick

Таблица 71 – Структура данных программы btnSchClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnShClick

Таблица 72 – Структура данных программы btnShClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnSqrtClick

Таблица 73 – Структура данных программы btnSqrtClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnThClick

Таблица 74 – Структура данных программы btnThClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnTrigClick

Таблица 75 – Структура данных программы btnTrigClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |

### Структура данных алгоритма btnClearFileClick

Таблица 76 – Структура данных программы btnClearFileClick(Sender)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| Sender | TObject | Объект | Формальный |
| LFile | File Of TDisp | Типизированный файл с историей | Локальный |

### Структура данных алгоритма Calculate

Таблица 77 – Структура данных программы Calculate(FRes)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| FRes | Real | Результат | Формальный |

### Структура данных алгоритма Display

Таблица 78 – Структура данных программы Display()

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| TRes | Real | Результат | Локальный |

### Структура данных алгоритма ResetData

Таблица 79 – Структура данных программы ResetData(FMem)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| FMem | TMem | Операнды, операция и результат | Формальный |

## Схема алгоритмов решения задачи по ГОСТ 19.701-90

Placeholder

## Графический интерфейс

Для организации графического интерфейса было использовано 3 формы: FormMain, FormHist и FormTrig. Взаимосвязь между формами представлена на рисунке 11.

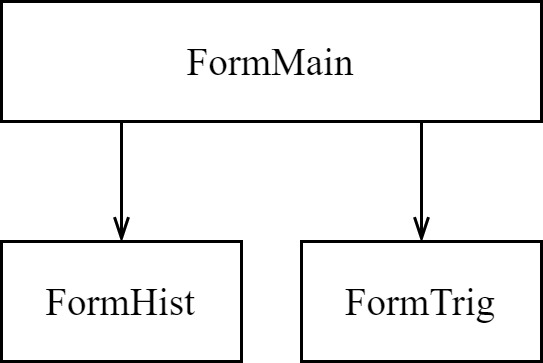


Рисунок 11

### Описание графических компонентов формы FormMain

Форма FormMain – это главная форма с набором кнопок. Пользователь может ввести число, осуществить операцию над числами, импортировать текстовый файл с числами или открыть две другие формы. Данная форма имеет вид, представленный на рисунке 12.

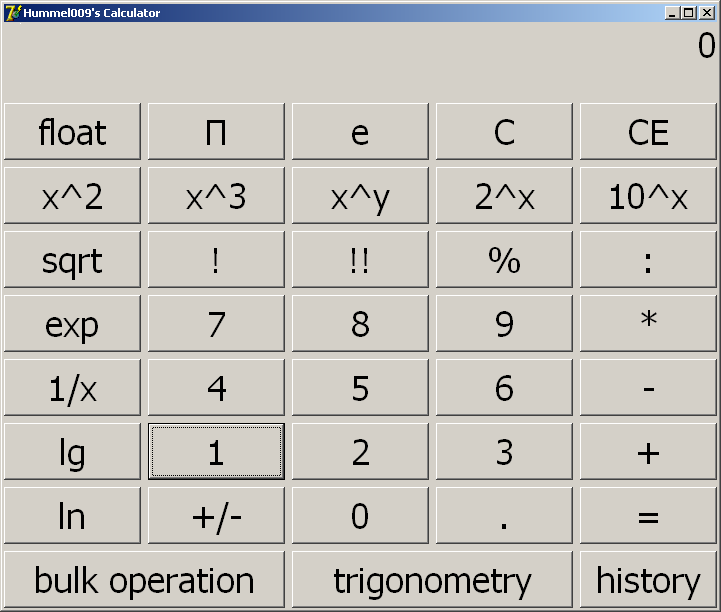


Рисунок 12 - Вид формы FormMain при запуске программы

Составляющие формы FormMain:

* текстовое поле «lblRes», на котором отображается результат;
* кнопки «.», «0», «1», «2», «3», «4», «5», «6», «7», «8», «9», выводящие одноимённый символ на экране;
* кнопка «%», присваивающая операцию «Х процентов от Y»;
* кнопка «:», присваивающая операцию «Х поделить на Y»;
* кнопка «\*», присваивающая операцию «Х умножить на Y»;
* кнопка «x^y», присваивающая операцию «Х в степени Y»;
* кнопка «-», присваивающая операцию «Х минус Y»;
* кнопка «+», присваивающая операцию «Х плюс Y»;
* кнопка «=», считывающая второе число, выполняющая присвоенную операцию и выводящая результат операции на экран;
* кнопка «sqrt», присваивающая операцию «корень из X» и выводящая результат операции на экран;
* кнопка «1/x», присваивающая операцию «двойной факториал X» и выводящая результат операции на экран;
* кнопка «lg», присваивающая операцию «двойной факториал X» и выводящая результат операции на экран;
* кнопка «+/-», используемая для ввода отрицательных чисел;
* кнопка «x^2», выполняющая операцию «X в квадрате» и выводящая результат операции на экран;
* кнопка «10^x», выполняющая операцию «десять в степени X» и выводящая результат операции на экран;
* кнопка «exp», выполняющая операцию «экспонента от X» и выводящая результат операции на экран;
* кнопка «lg», выполняющая операцию «факториал X» и выводящая результат операции на экран;
* кнопка «lg», выполняющая операцию «десятичный логарифм из X» и выводящая результат операции на экран;
* кнопка «ln», выполняющая операцию «двоичный логарифм из X» и выводящая результат операции на экран;
* кнопка «x^3», выполняющая операцию «X в кубе» и выводящая результат операции на экран;
* кнопка «2^x», выполняющая операцию «два в степени X» и выводящая результат операции на экран;
* кнопка «float», выводящая плавающую точку на экран ;
* кнопка «ce», очищающая экран;
* кнопка «c», стирающая один символ на экране;
* кнопка «e», выводящая число Эйлера на экран;
* кнопка «П», выводящая число Пи на экран;
* кнопка «history», открывающая форму «Form2» и загружающая данные из файла;
* кнопка «trigonometry», открывающая форму «Form3»;
* кнопка «bulk operation», загружающая текстовый файл с числами и выполняющая операцию сложения/перемножения этих чисел.

### Описание графических компонентов формы FormHist

Форма FormHist – это вспомогательная форма для показа истории вычислений. Пользователь может просмотреть историю, сохранить её в файл и загрузить её из файла. Данная форма имеет вид, представленный на рисунке 13.

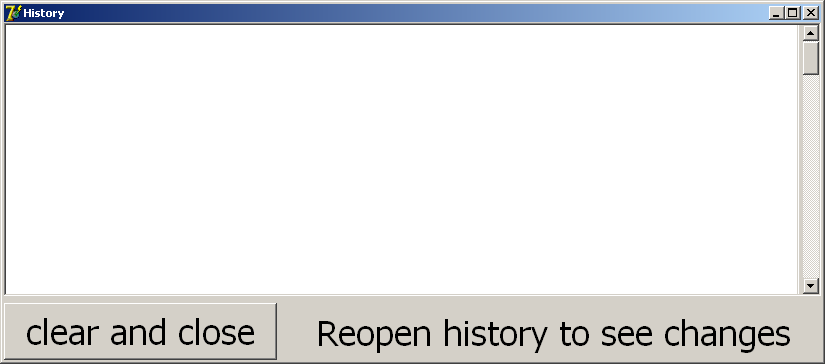


Рисунок 13 - Вид формы FormHist

Составляющие формы FormHist:

* поле «mmoHistory», где отображается история действий;
* кнопка «Clear and close» для очистки поля и файла с историей.

### Описание графических компонентов формы FormTrig

Форма FormTrig – это вспомогательная форма, где сгруппированы все тригонометрические и гиперболические функции. Данная форма имеет вид, представленный на рисунке 14.



Рисунок 14 - Вид формы FormTrig

Составляющие формы FormTrig:

* кнопка «sin», выполняющая операцию «синус X» и выводящая результат операции на экран;
* кнопка «cos», выполняющая операцию «косинус X» и выводящая результат операции на экран;
* кнопка «tg», выполняющая операцию «тангенс X» и выводящая результат операции на экран;
* кнопка «ctg», выполняющая операцию «котангенс X» и выводящая результат операции на экран;
* кнопка «arcsin», выполняющая операцию «арксинус X» и выводящая результат операции на экран;
* кнопка «arccos», выполняющая операцию «арккосинус X» и выводящая результат операции на экран;
* кнопка «arctg», выполняющая операцию «арктангенс X» и выводящая результат операции на экран;
* кнопка «arcctg», выполняющая операцию «арккотангенс X» и выводящая результат операции на экран;
* кнопка «sec», выполняющая операцию «секанс X» и выводящая результат операции на экран;
* кнопка «cosec», выполняющая операцию «косеканс X» и выводящая результат операции на экран;
* кнопка «arcsc», выполняющая операцию «арксеканс X» и выводящая результат операции на экран;
* кнопка «arccsc», выполняющая операцию «арккосеканс X» и выводящая результат операции на экран;
* кнопка «sh», выполняющая операцию «гиперболический синус X» и выводящая результат операции на экран;
* кнопка «ch», выполняющая операцию «гиперболический косинус X» и выводящая результат операции на экран;
* кнопка «th», выполняющая операцию «гиперболический тангенс X» и выводящая результат операции на экран;
* кнопка «cth», выполняющая операцию «гиперболический котангенс X» и выводящая результат операции на экран;
* кнопка «sch», выполняющая операцию «гиперболический секанс X» и выводящая результат операции на экран;
* кнопка «csch», выполняющая операцию «гиперболический косеканс X» и выводящая результат операции на экран;
* кнопка «versin», выполняющая операцию «синус-верзус X» и выводящая результат операции на экран;
* кнопка «vercos», выполняющая операцию «косинус-верзус X» и выводящая результат операции на экран;
* кнопка «hav», выполняющая операцию «гаверсинус X» и выводящая результат операции на экран;
* кнопка «hac», выполняющая операцию «гаверкосинус X» и выводящая результат операции на экран;
* кнопка «exsc», выполняющая операцию «экссеканс X» и выводящая результат операции на экран;

кнопка «excsc», выполняющая операцию «экскосеканс X» и выводящая результат операции на экран.

# Тестирование программного средства

placeholder

# Руководство по установке и использованию программного средства

placeholder

# Заключение

В процессе выполнения курсового проекта была проанализирована предметная область, рассмотрены существующие аналоги и выявлены их преимущества и недостатки. В качестве языка разработки использовался Delphi.

На этапе проектирования были разработаны блок-схема алгоритма работы программы. Было разработано программное средство инженерный калькулятор с возможностью обработки выражений, работы с константами, получения подробной информации по работе с приложением.

Проведено тестирование работоспособности разработанной программной части. Поставленная цель была выполнена в полном объеме, работоспособность подтверждена тестированием программного средства. Разработано руководство пользователя.

# Список использованной литературы

[1] Глухова, Л.А. Основы алгоритмизации и программирования: учебное пособие / Л.А. Глухова, В.В. Бахтизин. – Минск: БГУИР, 2003. – 72 с.

[2] Серебряная, Л.В. Структуры и алгоритмы обработки данных: учеб.-метод. пособие / Л.В. Серебряная, И. М. Марина. – Минск: БГУИР, 2013. – 51 с.

[3] Глухова, Л.А. Основы алгоритмизации и программирования.

Лабораторный практикум: учеб.-метод. пособие. В 4 ч. Ч. 4/ Л. А. Глухова, Е.П. Фадеева, Е.Е. Фадеева. – Минск: БГУИР, 2012. – 58 с.

[4] Фленов, М.Е. Библия Delphi. – 3-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2011. – 688 с.

[5] Фаронов, В.В. Delphi 6. Учебный курс.-М.: Издатель Молгачева С.В., 2001. – 672 с.

[6] Глухова, Л.А. Основы алгоритмизации и программирования: учебное пособие ч.2 / Л.А. Глухова. – Минск: БГУИР, 2006. – 177 с.