

Министерство науки и высшего образования Российской Федерации

федеральное государственное автономное

образовательное учреждение высшего образования

«Самарский национальный исследовательский университет

имени академика С.П. Королева»

Институт информатики и кибернетики

Кафедра технической кибернетики

## **Краткая документация по проекту**

### **«Math Monster»**

Выполнили:

Дробин Р. Р., группа 6210.

Козлов Н.М., группа 6210.

Долгих Д. Ю., группа 6210.

Степанов Н. А., группа 6210.

Самара, 2022

## Техническое задание

### Аннотация проекта

Math Monster - многофункциональный калькулятор, предназначенный для решения задач основных разделов математики.

В процессе обучения приходилось пользоваться другими многофункциональными калькуляторами: Wolfram Alpha, Desmos. Часто их работа и алгоритмы вызывали интерес. И с объявлением проектной деятельности в начале 4 семестра идея о написании подобного приложения, которое будет востребовано и интересно нам, сразу засела в светлых умах некоторых членов нашей команды. Вместе с этим появилась возможность проверить себя, подкрепить полученные в университете знания.

Изначально наша команда ставила себе следующие задачи:

- Написание модулей основных разделов:
  1. Алгебра.
  2. Математический анализ.
  3. Геометрия.
  4. Дискретная математика .
- Минимальный GUI.

В качестве дополнительных задач:

- Дополнение к GUI красивой анимации, эффектов, иконок к каждому разделам.
- Краткие выкладки по необходимой теории для каждой задачи.

По итогу, нам удалось выполнить все задачи "минимум" и "максимум". А также под конец работы удалось расширить блок дискретной математики, написав помимо графов небольшой модуль комбинаторики, добавить различные компоненты для облегчения работы пользователя с нашим приложением, например, ToolTip.

Подробный список реализованных по итогу функциональных задач, которые детальнее представлены ниже:

- Алгебра:

- Операции с матрицами(основные арифметические операции, характеристики матрицы, LU и QR разложение).
- Операции с многочленами (основные арифметические, НОД, НОК, нахождение корней).
- Теория чисел(НОД и НОК, операции над комплексными числами).
- Дискретная математика:

Теория графов:

- поиск кратчайших путей, центр, радиус, медиана,
- компоненты связности,
- гамильтонов цикл,
- эйлеров путь/цикл,
- минимальное остовное дерево неорграфа,
- раскраска графа;

Комбинаторика:

- перестановки с повторениями и без них;
- сочетания с повторениями и без них;
- размещения с повторениями и без них;

- Матанализ:
  - производная (численное и аналитическое дифференцирование);
  - собственный интеграл;
  - сумма ряда;
  - локальные экстремумы;
  - нахождение значения функции в точке;
- Геометрия:
  - Построение графиков функции (явное задание),
  - уравнение касательной,
  - операции над векторами на плоскости и в пространстве,
  - углы и расстояния между точками/прямыми/плоскостями,
  - определение типа КВП и ПВП

## **Состав команды**

Дробин Роман Равильевич - тимлидер, программист блока “Алгебра”, компоновка отдельных модулей в приложение, идея структуры и общего дизайна приложения.

Козлов Никита Максимович - автор идеи, программист блока “Математический анализ”, тестер, backend “Комбинаторики”, работа с документацией.

Долгих Данила Юрьевич - программист блока “Теория графов”, frontend “Комбинаторики”, работа с документацией.

Степанов Никита - программист блока «Геометрия», «Алгебра многочленов» и «Векторная алгебра», работа с документацией.

## **Введение**

Калькулятор был, есть и всегда будет актуальным инструментом, помогающим при счёте. Технологии развиваются, методы изменяются, но суть всегда остаётся прежней – упростить жизнь, избавив себя от необходимости длинных и нудных вычислений. Наше приложение это не какой-то там калькулятор, который умеет складывать  $2+2$ , это многофункциональное приложение, в котором используются современнейшие методы, написанные нами самостоятельно, без использования готовых решений, для вычисления далеко не базовых задач. Оно зиждется на четырех столпах математики, и способно быстро, качественно и удобно получить результат буквально в пару кликов!

Особую актуальность это приложение будет иметь среди студентов Самарского Университета и для нас самих, поскольку львиная доля алгоритмов сопутствует программе обучения бакалавров.

## **Особенности разработки**

### **Технические особенности и реализация**

Программа была написана в среде разработки Visual Studio на языке программирования C#. Для визуализации приложения был применён встроенный интерфейс WinForms. Рассмотрим подробнее для каждого раздела математики стек примененных технологий:

- Математический анализ. Для преобразования строки в обрабатываемую компьютером было реализовано несколько алгоритмов, идейно повторяющих Обратную Польскую Запись и конкретно алгоритм сортировочной станции в случае нахождения производной. В алгоритме нахождения значения функции в точке были использованы указатели в качестве входных параметров методов, что позволило поэлементно считать функцию, дабы впоследствии преобразовать её в понимаемый машиной код. Также алгоритм нахождения определенного интеграла основан на методе Симпсона. Производная, локальные экстремумы, локальные минимумы и максимумы, а также нахождение суммы ряда найдены по общеизвестным правилам/определениям;

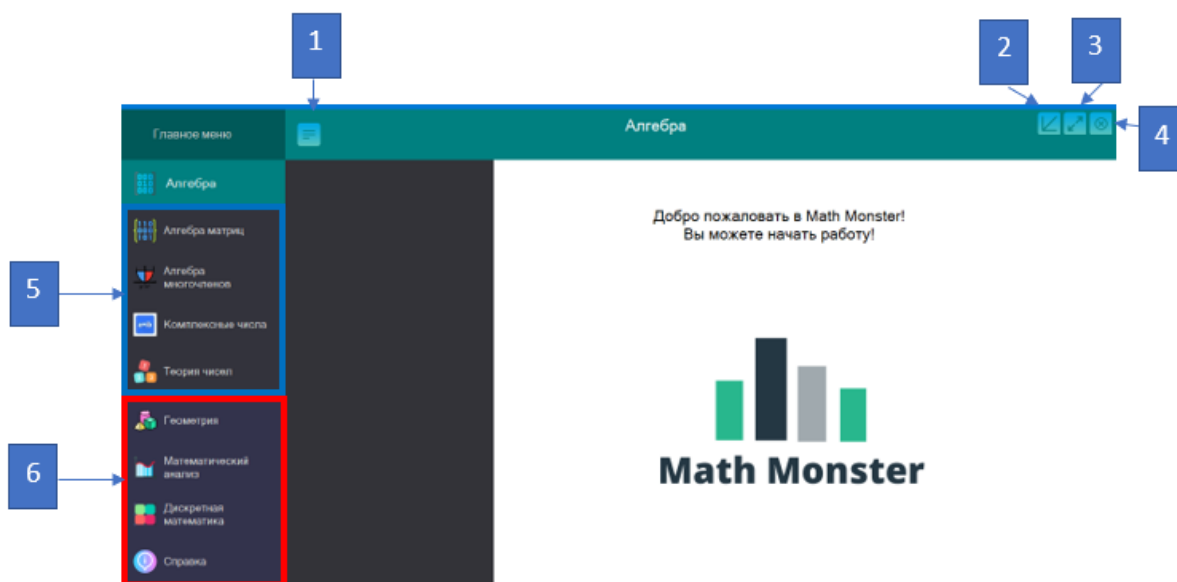
- В теории графов пришлось реализовывать рисование вершин и ребер с помощью пространства имен `System.Drawing`.
- В комбинаторике использовался тип `BigInteger` из пространства имен `System.Numerics`, что позволило значительно увеличить диапазон значений (до  $9000!$ (факториал)), а также `System.Linq` для наиболее оптимального нахождения факториала.
- Алгебра. Для некоторых методов был использован `List` из пространства имен `System.Collections.Generic`, но в основном были написаны собственные структуры данных такие, как `Matrix`(матрица), `Complex`(комплексное число), `Number`(особое число, которое помимо самого значения числа, хранит число вхождений в разложении на простые множители).
- Алгебра полиномов. Для представления полиномов в виде определенной структуры использовался собственный класс `Polynom`. Некоторые Арифметические операции реализованы в собственном статическом классе `OperationsOnPolinoms`. Остальные операции и поиск корней в самом классе `Polynom`.
- Геометрия. Для реализации отдельных методов использовались собственно-написанные классы `Vector`, `Matrix` и другие, определенные в классе `Algebra`. Функционал некоторых методов позволяет строить графики функций, для чего была использована компонента `Chart` и пространство имён `System.Windows.Forms.DataVisualization` для работы с ней. Также некоторые методы реализовывались с помощью собственного статического класса `OperationsOnVectors`. Основной используемый тип данных для работы - `Double`.

Для командного взаимодействия использовалась социальная сеть ВКонтакте, а также программы `Zoom` и `Discord`, где мы показывали друг другу на какой стадии разработки находится код. Хранение и обмен кодом проводилось через `Git` и `GitHub`.

## Внешний вид и взаимодействие с пользователем (руководство пользователя)

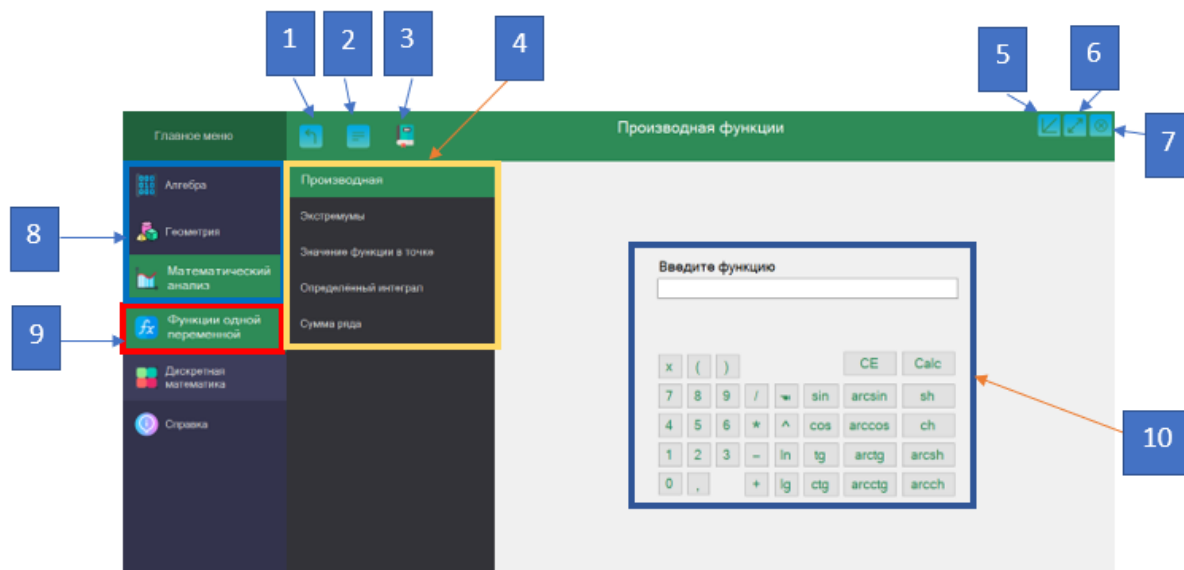
### Начало работы

В данном приложении собрано огромное множество функций и для их корректной работы необходимо соблюдать этикет ввода.



- 1) Скрыть главное меню
- 2) «Свернуть»
- 3) Развернуть на полный экран
- 4) «Заккрыть»
- 5) Подразделы
- 6) Разделы

Слева находится главное меню, в котором можно наблюдать кнопки основных 4 разделов, при нажатии на которые (Геометрия - исключение) открываются подразделы, выделенные другим цветом.



- 1) Вернуться в главное меню
- 2) Скрыть главное меню
- 3) Открыть справку для данного метода
- 4) Методы подраздела
- 5) «Свернуть»
- 6) Развернуть на полный экран
- 7) «Закрывать»
- 8) Разделы
- 9) Подразделы
- 10) Форма выбранной функции

При нажатии на некоторые кнопки из подраздела, чуть правее основного меню будет открываться блок функций, каждая из которых решает определенную задачу. При выборе такой функции в основной панели будет открываться форма, приглашающая ко вводу данных и соответствующим расчетам.

Особенностью является подсветка всех нажатых на данный момент кнопок, причём своим уникальным цветом для каждого раздела. До нажатия на кнопку из блока функций можно было менять свой выбор обычным нажатием на кнопку из главного меню или на кнопку подраздела, однако после нажатия на кнопку из блока функций, все эти кнопки блокируются, и на верхней панели появляется кнопка со стрелочкой,



как раз при нажатии на которую, мы сможем вернуться в главное меню, разблокировать все прочие кнопки и продолжить работу.

На верхней панели также находится кнопка сокрытия меню, подменю и блоков функций. Также при нажатии на определённый метод из блока функций (в основном на первую кнопку по порядку) в верхней панели появляется третья кнопка с книжкой, которая отвечает за теоретические выкладки к текущему блоку. Эти выкладки включают минимальную необходимую теорию по каждой функции и помогут понять, что вообще делает тот или иной метод.

## Правила пользования

### *Блок “Математический анализ”*

Корректный ввод очень важен для дальнейших вычислений, потому пришлось обрабатывать все возможные исключительные ситуации, до которых мог бы догадаться пользователь. Примеры ошибок и их обработка:

- Если введены некорректные данные при указании диапазона (должны быть int или double)

The screenshot displays a mathematical calculator application. On the left, a small error dialog box is open with the text "Введите корректные данные!" (Enter correct data!) and an "OK" button. The main interface has a title "Введите функцию" (Enter function) above a text input field containing "x". Below this, it says "Введите [a;b]:" (Enter [a;b]:) with two input fields containing "p" and "n". To the right, under the heading "Результат" (Result), there is a text box showing "Input error". At the bottom is a keypad with buttons for variables (x, (, ), CE, Calc), digits (7, 8, 9, 4, 5, 6, 1, 2, 3, 0), operators (/, \*, ^, -, +, ,), and functions (sin, cos, tg, ctg, ln, lg, π).

- Если введена некорректная функция, или выражение, не являющееся функцией, или же не соблюден баланс скобок

Введите функцию

Введите пределы интегрирования

$\int$

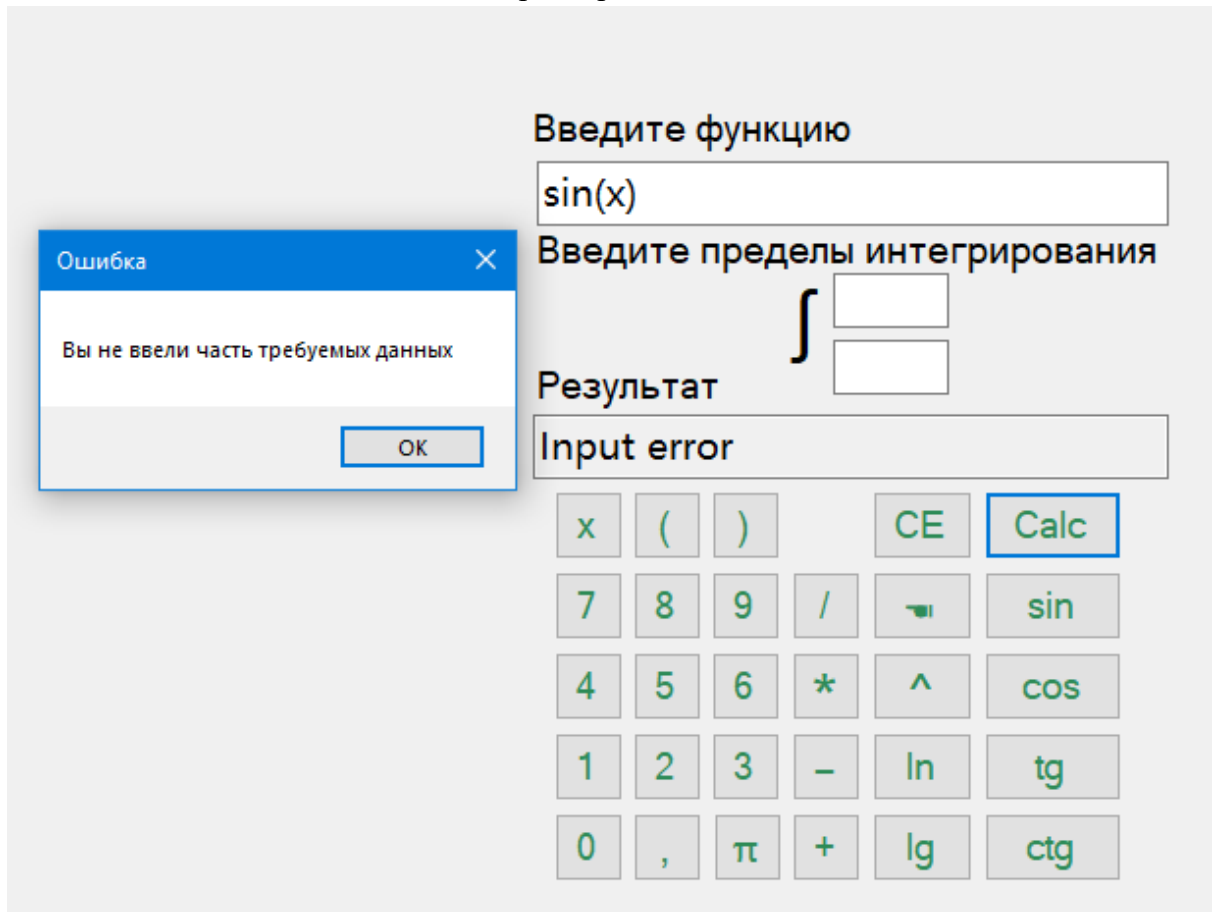
$\int$

Error of syntactic check of expression, balance of bracket is not correct.  
String: sin(  
Uncorrect symbol: \0  
Position in string: 4

OK

0 ,  $\pi$  + lg ctg

- Если введены не все необходимые параметры



### Производная

- Пользователь должен ввести функцию.
- Переменной является только "x".
- Доступны лишь те функции, что имеются на панели управления за исключением  $\exp()$ , остальные функции алгоритм будет выплёвывать и выдавать ошибку.
- Обязательно соблюдение пунктуации: все скобки должны быть парными, между двумя операндами должен быть только один оператор, допускается и запятая, и точка при вводе чисел с плавающей точкой.
- После ввода функции при нажатии на кнопку «Calc» калькулятор вычислит производную и выведет её в появившемся текстовом поле.

Пример корректной работы:

Введите функцию

Результат

x	(	)	CE						Calc
7	8	9	/	←	sin	arcsin	sh		
4	5	6	*	^	cos	arccos	ch		
1	2	3	-	ln	tg	arctg	arcsh		
0	,		+	lg	ctg	arcctg	arcch		

### Сумма ряда

- Пользователь должен ввести сходящийся ряд.
- Переменной является только “x”.
- Доступны лишь те функции, что имеются на панели управления, остальные функции алгоритм будет выплёвывать и выдавать ошибку.
- Функция “!” отображается в текстовом поле как fact(), потому её следует вводить перед, а не после параметров функции.
- Обязательно соблюдение пунктуации: все скобки должны быть парными, между двумя операндами должен быть только один оператор, допускается и запятая, и точка при вводе чисел с плавающей точкой.
- После ввода функции при нажатии на кнопку «Calc» калькулятор вычислит сумму ряда и выведет её в появившемся текстовом поле.

Пример корректной работы:

Введите ряд

$\sum$

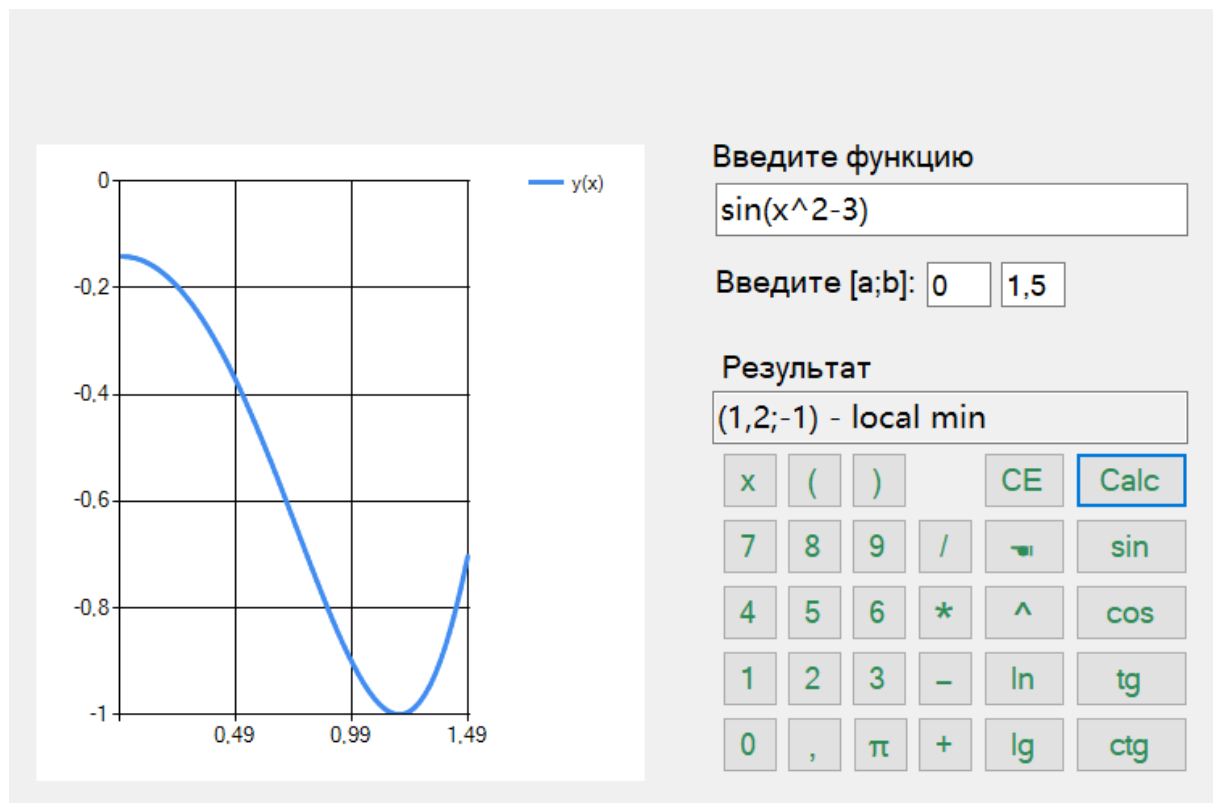
Результат

x	(	)	!	CE	Calc
7	8	9	/	$\frac{\square}{\square}$	sin
4	5	6	*	^	cos
1	2	3	-	ln	tg
0	,	$\pi$	+	lg	ctg

### Локальный экстремум

- Пользователь должен ввести функцию, а также интервал, на котором определён один локальный экстремум.
- Переменной является только “x”.
- Все входные поля обязательно должны быть заполнены.
- Доступны лишь те функции, что имеются на панели управления, остальные функции алгоритм не распознает и будет выдавать ошибку.
- Обязательно соблюдение пунктуации: все скобки должны быть парными, между двумя операндами должен быть только один оператор, допускается и запятая, и точка при вводе чисел с плавающей точкой.
- После ввода функции при нажатии на кнопку «Calc» калькулятор вычислит точку локального экстремума, выведет её в появившемся текстовом поле и укажет, является ли она точкой локального минимума/максимума функции.

Пример корректной работы:



### Определённый интеграл

- Пользователь должен ввести функцию, а также интервал, на котором определена функция.
- Переменной является только “x”.
- Все входные поля обязательно должны быть заполнены.
- Доступны лишь те функции, что имеются на панели управления, остальные функции алгоритм не распознает и будет выдавать ошибку.
- Обязательно соблюдение пунктуации: все скобки должны быть парными, между двумя операндами должен быть только один оператор, допускается и запятая, и точка при вводе чисел с плавающей точкой.
- После ввода функции при нажатии на кнопку «Calc» калькулятор вычислит определённый интеграл и выведет его значение в появившемся текстовом поле.

Пример корректной работы:

The screenshot shows a web-based calculator interface. At the top, there is a label "Введите функцию" (Enter the function) above a text input field containing "2\*x". Below this is a label "Введите пределы интегрирования" (Enter the limits of integration) above a large integral symbol  $\int$ . To the right of the integral symbol are two stacked input fields: the top one contains "2" and the bottom one contains "1". Below the integral symbol is a label "Результат" (Result) above a text input field containing "2,99933333333322". At the bottom of the interface is a numeric keypad with buttons for digits 0-9, decimal point, pi, plus, minus, multiply, divide, power, and inverse. There are also buttons for trigonometric functions: sin, cos, tg, and ctg. A "CE" button is located to the left of the "Calc" button, which is highlighted with a blue border.

### Значение функции в точке

- Пользователь должен ввести функцию, а также точку  $x$ , в которой будет искомое значение.
- Переменной является только "x".
- Все входные поля обязательно должны быть заполнены.
- Доступны лишь те функции, что имеются на панели управления, остальные функции алгоритм будет выплёвывать и выдавать ошибку.
- Обязательно соблюдение пунктуации: все скобки должны быть парными, между двумя операндами должен быть только один оператор, допускается и запятая, и точка при вводе чисел с плавающей точкой.
- После ввода функции при нажатии на кнопку «Calc» калькулятор вычислит значение функции в точке и выведет его значение в появившемся текстовом поле.

Пример корректной работы:

Введите функцию

Введите значение x

Результат

x	(	)		CE	Calc
7	8	9	/		sin
4	5	6	*	^	cos
1	2	3	-	ln	tg
0	,	π	+	lg	ctg

### ***Блок “Геометрия”***

Сразу стоит отметить, что при некорректном вводе пользователь будет осведомлен об этом. Примеры ошибок и их обработка:



- Отсутствие значения в поле ввода:

Определение кривой второго порядка.

☒ На плоскости  
☐ В пространстве

Функция

Кривая второго порядка

$x^2$  +   $y^2$  +   $xy$  +   $x$  +   $y$  +

Внимание! ✕

Введите параметры

ОК

Результат

Выполнить

- Всё кроме целого числа или числа с плавающей точкой:

Расстояние от точки до прямой.

☒ На плоскости  
☐ В пространстве

Функция

Параметры

Точка: X:  Y:

$y(x) =$    $x +$

Ошибка ✕

✖ Введите число с запятой

ОК

Результат

Выполнить

- Отсутствие выбора формы ввода, если их несколько:

Уравнение прямой.

Функция

Внимание! Выберите формат ввода

ОК

Результат

На плоскости

В пространстве

Параметры

По 2 точкам

X1: 3 Y1: 4

X2: 2 Y2: 3

По точке и вектору

A: X: Y:

n: X: Y:

Выполнить

- Введение нулевых прямых / плоскостей / векторов

Расстояние между прямыми и плоскостями.

На плоскости

В пространстве

Функция

Введите ненулевую плоскость

2 прямые

L1: x - = y - = z -

L2: x - = y - = z -

Прямая и плоскость

S: x + y + z +

L: x - = y - = z -

2 плоскости

S1: 0 x + 0 y + 0 z + 2

S2: 0 x + 0 y + 1 z + 2

Выполнить

График функции на плоскости и уравнения касательной и нормальной к нему

## График функции

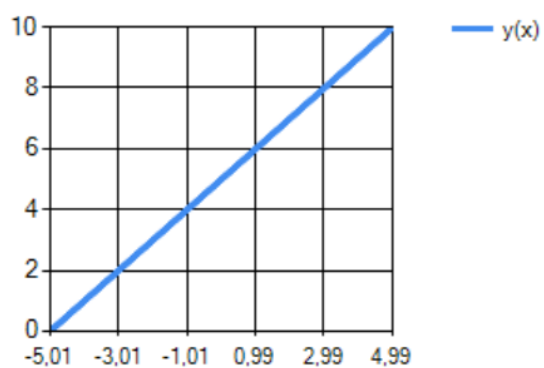
### Правила пользования

- Ввод функции осуществляется с помощью клавиатуры с неизвестной “х”.
- Коэффициентами или константами в функции должны быть целые числа или числа с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- Также пользователь может выбрать шаг построения, который будет являться численным параметром кол-ва построенных точек, и границы, в которых будет построен график функции, соблюдая некоторые правила:
  1. Введенные границы и шаг должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
  2. Левая граница а должна быть меньше правой границы b.
- Если не указывается хотя бы один из дополнительных параметров, то они не будут учитываться при построении графика функции.
- После ввода функции (и возможных дополнительных параметров) при нажатии на кнопку «Выполнить» калькулятор построит график функции.

График функции одной переменной.

- ☒ На плоскости  
☐ В пространстве

Функция



Параметры

$y(x) = x + 5$

Ось X:

Интервал: [     ;     ]

Шаг:    

Абсцисса точки касания  $x_0 =$

Уравнение касательной и нормали

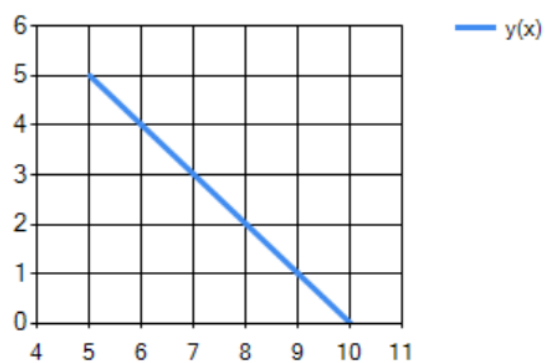
$$y(x) = x + 5$$

Выполнить

График функции одной переменной.

- ☒ На плоскости  
☐ В пространстве

Функция



Параметры

$y(x) = -1 \cdot x + 10$

Ось X:

Интервал: [ 5 ; 10 ]

Шаг: 1

Абсцисса точки касания  $x_0 =$

Уравнение касательной и нормали

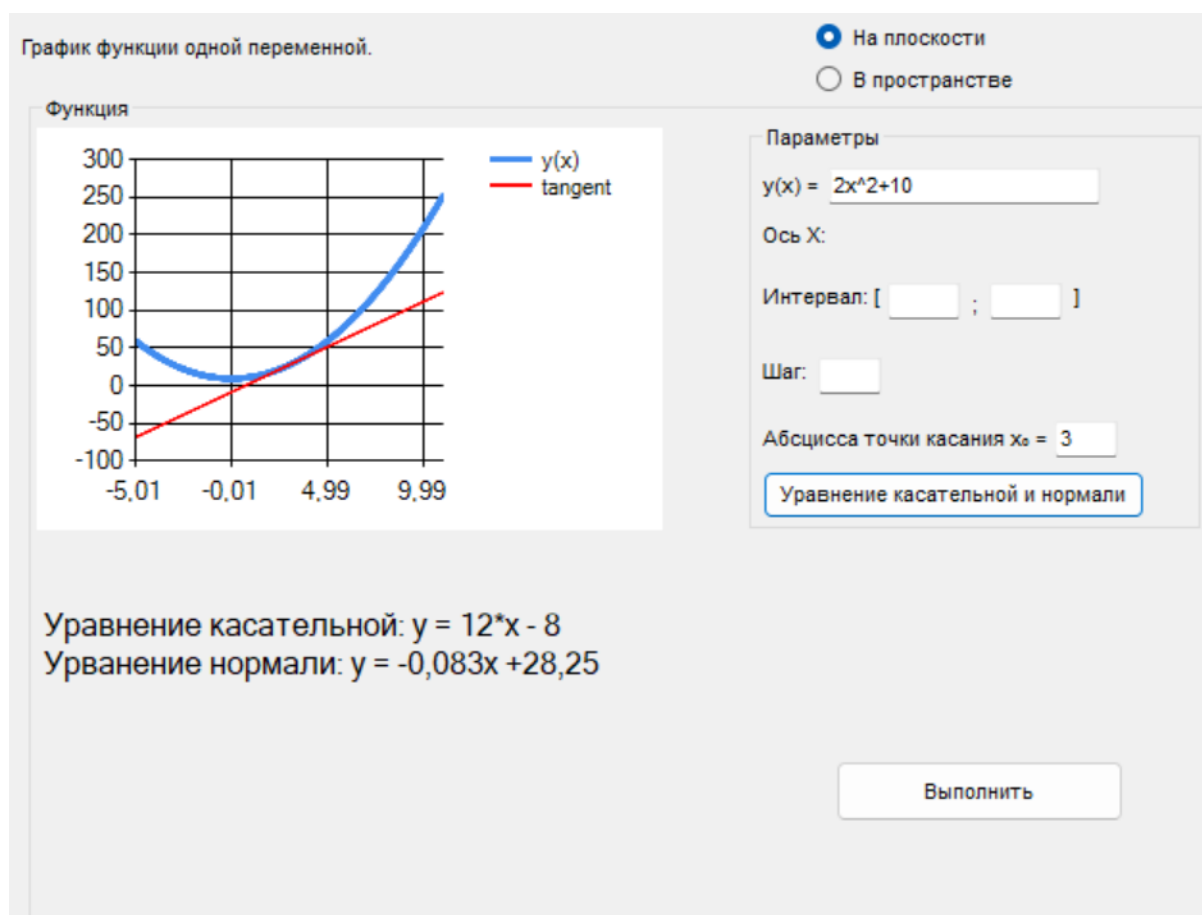
$$y(x) = -1 \cdot x + 10$$

Выполнить

## Уравнения касательной и нормали к графику функции

### Правила пользования

- Для получения уравнений касательной и нормали к графику функции необходимо, чтобы была введена какая-либо функция и абсцисса точки касания.
- Пользователь должен вводить только целые числа или числа с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода абсциссы точки касания, функции (и возможных дополнительных параметров) при нажатии на кнопку «Уравнение касательной и нормали» калькулятор построит график касательной и вычислит уравнения касательной и нормали к графику функции.



## Уравнение прямой в пространстве

### Уравнение прямой в пространстве по двум точкам

- Пользователь должен ввести две точки с координатами (x, y, z)
- Значения координат должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).

· После ввода двух точек при нажатии на кнопку «Выполнить» калькулятор построит график этой прямой и выведет её уравнения.

Уравнение прямой.

☐ На плоскости  
☒ В пространстве

Параметры

По 2 точкам

☒ X1: 3 Y1: 21 Z1: 5  
X2: 1 Y2: 3 Z2: 3

По точке и вектору

☐ A: X: 3 Y: 1 Z: 5  
n: X: 4 Y: 54 Z: 3

-0.5x -1.5 = -0.056y -1.167 = -0.5z -2.5

Выполнить

### Уравнение прямой на плоскости по точке и направляющему вектору

· Пользователь должен ввести точку с координатами (x, y, z) и направляющий вектор с соответствующими координатами вектора нормали к этой прямой.

· Значения координат должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).

· После ввода точки и направляющего вектору прямой при нажатии на кнопку «Выполнить» калькулятор построит график этой прямой и выведет её уравнения.

Уравнение прямой.

☐ На плоскости  
☒ В пространстве

Функция

Параметры

☐ По 2 точкам

X1: 3 Y1: 21 Z1:

X2: 1 Y2: 3 Z2:

☒ По точке и вектору

A: X: 3 Y: 1 Z: 5

n: X: 4 Y: 54 Z: 3

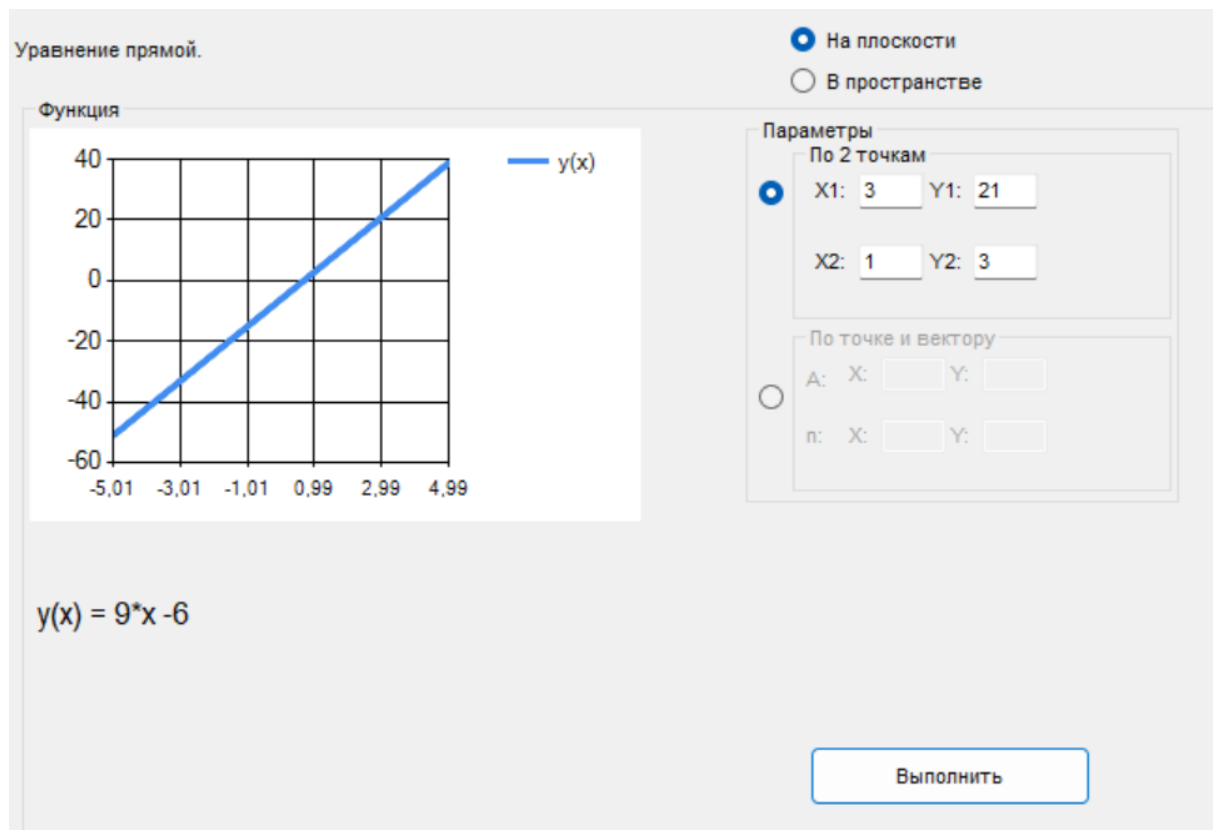
$0.25x + 0.75 = 0.019y + 0.019 = 0.333z + 1.667$

Выполнить

### Уравнение прямой на плоскости

#### Уравнение прямой на плоскости по двум точкам

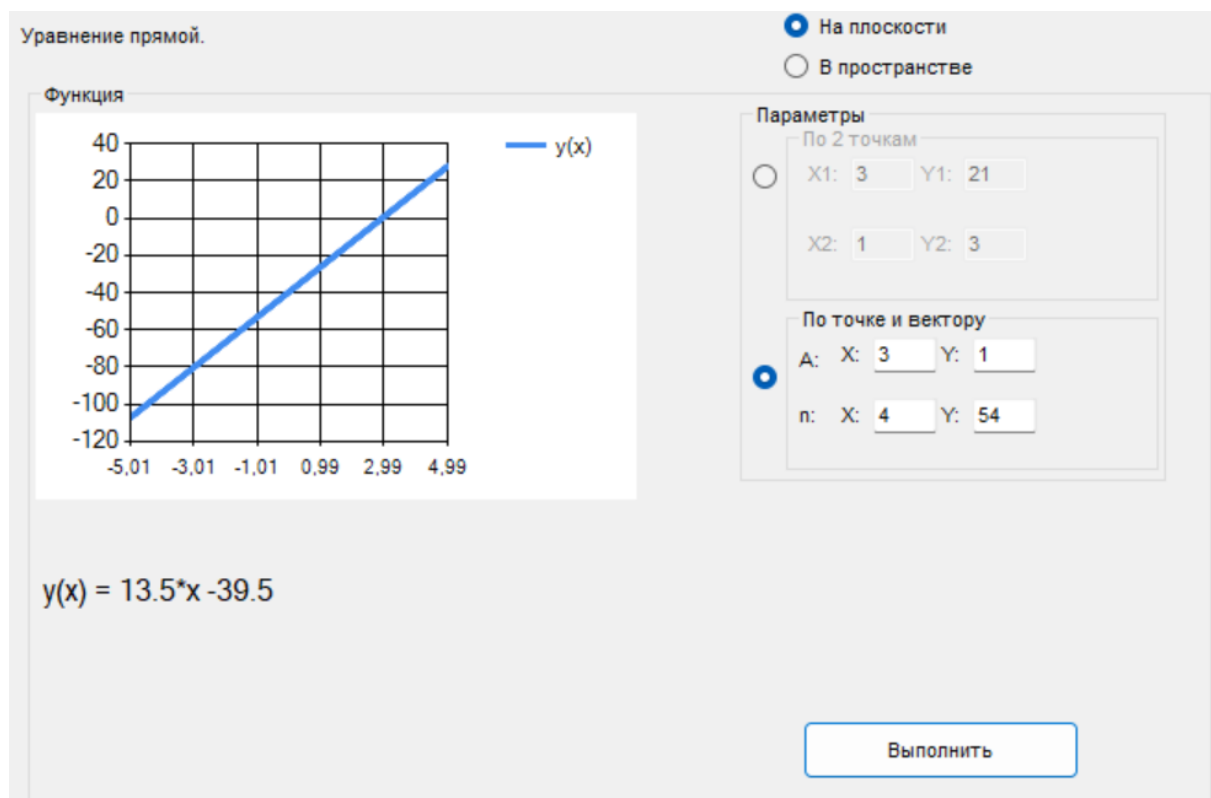
- Пользователь должен ввести две точки с координатами (x, y)
- Значения координат должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода двух точек при нажатии на кнопку «Выполнить» калькулятор построит график этой прямой и выведет её уравнения.



### Уравнение прямой на плоскости по точке и направляющему вектору

- Пользователь должен ввести точку с координатами (x, y) и направляющий вектор с соответствующими координатами вектора нормали к этой прямой.
- Значения координат должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода точки и направляющего вектора прямой при нажатии на кнопку «Выполнить» калькулятор построит график этой прямой и выведет её уравнения.





## Уравнение плоскости в пространстве

### Уравнение плоскости по трём точкам в пространстве

- Пользователь должен ввести три точки с координатами (x, y, z)
- Значения координат должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода трёх точек при нажатии на кнопку «Выполнить» калькулятор выведет уравнение этой плоскости.

Уравнение плоскости.

☐ На плоскости  
☒ В пространстве

Функция

Параметры

☒ По 3 точкам

X1:  Y1:  Z1:   
X2:  Y2:  Z2:   
X3:  Y3:  Z3:

☐ По точке и нормали

A: X:  Y:  Z:   
n: X:  Y:  Z:

$5x - 1y - 7z + 18 = 0$

Выполнить

### Уравнение плоскости по точке и нормали в пространстве

- Пользователь должен ввести точку с координатами (x0, y0, z0) и нормаль (A, B, C).
- Значения координат должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода точки и нормали при нажатии на кнопку «Выполнить» калькулятор выведет уравнение этой плоскости.

Уравнение плоскости.

☐ На плоскости  
☒ В пространстве

Функция

Параметры

☐ По 3 точкам

X1:	<input type="text" value="4"/>	Y1:	<input type="text" value="3"/>	Z1:	<input type="text" value="5"/>
X2:	<input type="text" value="1"/>	Y2:	<input type="text" value="2"/>	Z2:	<input type="text" value="3"/>
X3:	<input type="text" value="3"/>	Y3:	<input type="text" value="5"/>	Z3:	<input type="text" value="4"/>

☒ По точке и нормали

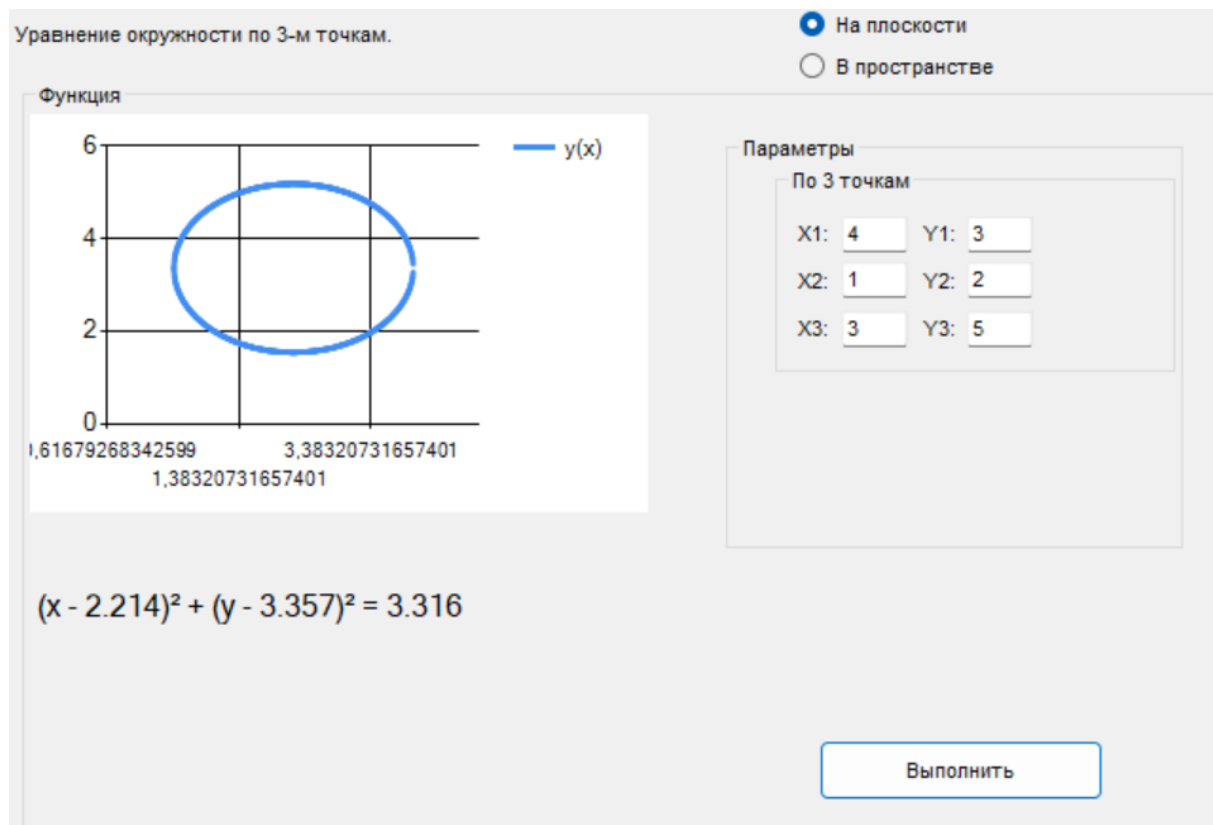
A:	X: <input type="text" value="6"/>	Y: <input type="text" value="86"/>	Z: <input type="text" value="0"/>
n:	X: <input type="text" value="1"/>	Y: <input type="text" value="82"/>	Z: <input type="text" value="2"/>

$1x + 82y + 2z - 7058 = 0$

Выполнить

### Уравнение окружности по трём точкам на плоскости

- Пользователь должен ввести три точки с координатами (x, y)
- Значения координат должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).
- После ввода трёх точек при нажатии на кнопку «Выполнить» калькулятор построит график этой окружности и выведет её уравнения.



### Расстояние между точкой и прямой на плоскости

- Пользователь должен ввести координаты точки и уравнение прямой угловым коэффициентом  $k$  и свободным членом  $b$ .
- Значения коэффициентов должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода точки и прямой при нажатии на кнопку «Выполнить» калькулятор определяет расстояние между точкой и прямой.

Расстояние от точки до прямой.

☒ На плоскости  
☐ В пространстве

Функция

Параметры

Точка: X:  Y:

$y(x) =$  $x +$

Расстояние: 7,77817459305202

### Расстояние между точкой и прямой в пространстве

- Пользователь должен ввести координаты точки и каноническое уравнение прямой.
- Значения коэффициентов должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода точки и прямой при нажатии на кнопку «Выполнить» калькулятор определяет расстояние между точкой и прямой.

Расстояние от точки до прямой / плоскости.

☐ На плоскости  
☒ В пространстве

Функция

Параметры

Точка и прямая

М: (  ;  ;  )

☒ L:  $x -$  $= y -$  $= z -$

---

Точка и плоскость

☐ M: (  ;  ;  )  
 S:  x +  y +  z +

Расстояние: 44,574

Выполнить

### Расстояние между точкой и плоскостью в пространстве

- Пользователь должен ввести координаты точки и уравнение плоскости.
- Значения коэффициентов должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода точки и плоскости при нажатии на кнопку «Выполнить» калькулятор определяет расстояние между точкой и плоскости.

Расстояние от точки до прямой / плоскости.

☐ На плоскости  
☒ В пространстве

Функция

Параметры

Точка и прямая

☐ M: ( 80 ; 73 ; 0 )  
 L:  $x - 2 = y - 9 = z - 8$   
           2       3       1

Точка и плоскость

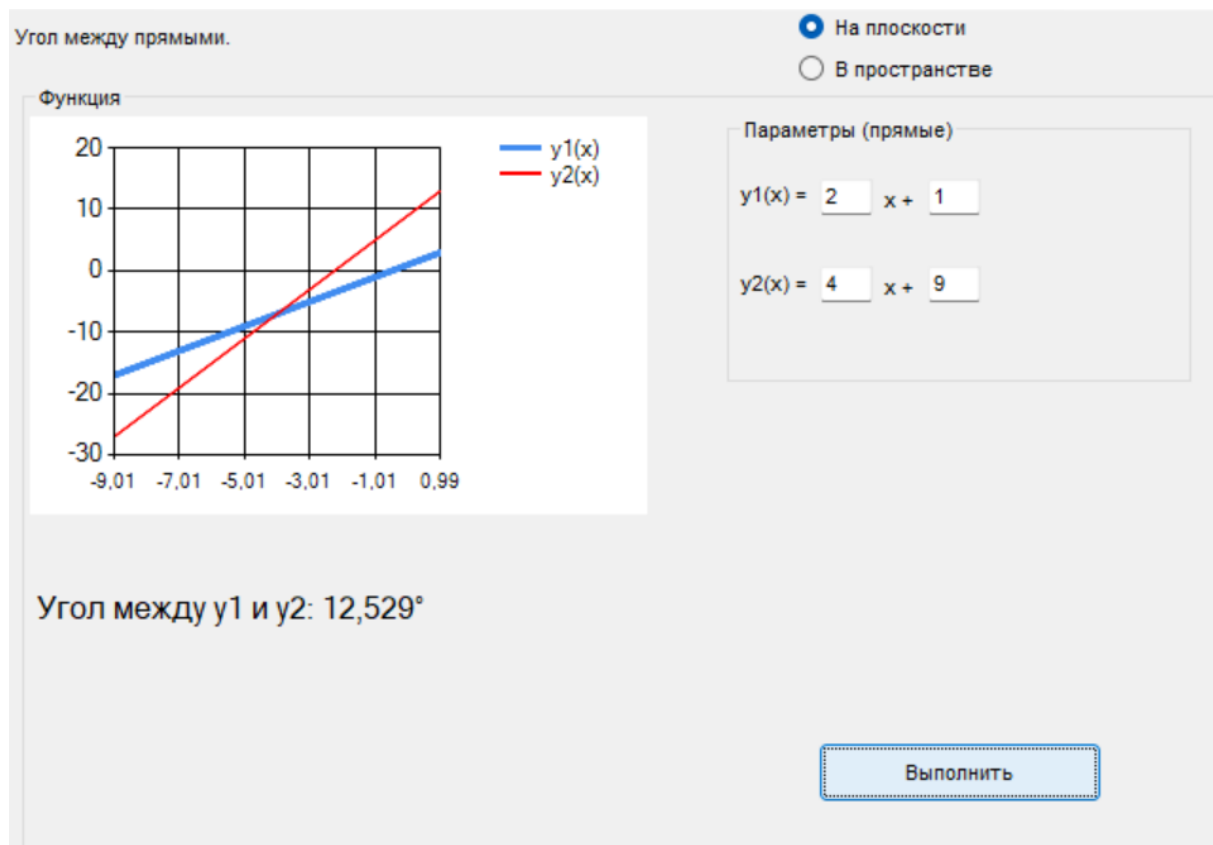
☒ M: ( 10 ; 6 ; 5 )  
 S: 2 x + -2 y + 2 z + 0

Расстояние: 5,196

Выполнить

### Угол между прямыми на плоскости

- Пользователь должен ввести две прямые, т.е. их коэффициенты  $k_1$ ,  $b_1$ ,  $k_2$ ,  $b_2$  соответственно, где  $k$  – угловой коэффициент, а  $b$  – свободный член.
- Значения коэффициентов должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода двух прямых при нажатии на кнопку «Выполнить» калькулятор определяет расстояние между ними.



## Вычисление углов

### Угол между прямыми в пространстве

- Пользователь должен ввести две прямые, заданных каноническими уравнениями.
- Значения коэффициентов в уравнениях должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).
- После ввода двух прямых при нажатии на кнопку «Выполнить» калькулятор определяет угол между ними.



Углы между прямыми и плоскостями.

Функция

☐ На плоскости  
☒ В пространстве

2 прямые

L1:  $x - 68 = y - 8 = z - 62$

☒
 $\frac{4}{2} = \frac{-2}{1} = \frac{-3}{3}$

L2:  $x - 8 = y - 72 = z - -1$

Прямая и плоскость

☐
 S:  $x + y + z +$   
 L:  $x - = y - = z -$

2 плоскости

☐
 S1:  $x + y + z +$   
 S2:  $x + y + z +$

Угол = 98,562°

Выполнить

### Угол между прямой и плоскостью в пространстве

- Пользователь должен ввести каноническое уравнение прямой и уравнение плоскости.
- Значения коэффициентов в уравнениях должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).
- После ввода прямой и плоскости при нажатии на кнопку «Выполнить» калькулятор определяет угол между ними.

Углы между прямыми и плоскостями.

Функция

☐ На плоскости  
☒ В пространстве

2 прямые

L1:  $x - 68 = y - 8 = z - 62$

☐
 $\frac{4}{2} = \frac{-2}{1} = \frac{-3}{3}$

L2:  $x - 8 = y - 72 = z - -1$

Прямая и плоскость

☒
 S:  $4x + 2y + 3z + 2$   
 L:  $x - 1 = y - 2 = z - 3$   
 $\frac{4}{4} = \frac{5}{5} = \frac{4}{4}$

2 плоскости

☐
 S1:  $\square x + \square y + \square z + \square$   
 S2:  $\square x + \square y + \square z + \square$

Угол = 69,171°

Выполнить

### Угол между плоскостями в пространстве

- Пользователь должен ввести две плоскости с помощью уравнений.
- Значения коэффициентов в уравнениях должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).
- После ввода двух плоскостей при нажатии на кнопку «Выполнить» калькулятор определяет угол между ними.

Углы между прямыми и плоскостями.

Функция

☐ На плоскости  
☒ В пространстве

2 прямые

L1:  $x - 68 = y - 8 = z - 62$

☐
 $\frac{4}{2} = \frac{-2}{1} = \frac{-3}{3}$

L2:  $x - 8 = y - 72 = z - -1$

Прямая и плоскость

☐
 S:  $4x + 2y + 3z + 2$   
 L:  $x - 1 = y - 2 = z - 3$

2 плоскости

☒
 S1:  $7x + 69y + 20z - 1$   
 S2:  $2x + 3y + 3z + 4$

Угол = 33,902°

Выполнить

## Расстояние между прямыми на плоскости

### Правила пользования

- Пользователь должен ввести две прямые, т.е. их коэффициенты  $k_1$ ,  $b_1$ ,  $k_2$ ,  $b_2$  соответственно, где  $k$  – угловой коэффициент, а  $b$  – свободный член.
- Значения коэффициентов должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода двух прямых при нажатии на кнопку «Выполнить» калькулятор определяет расстояние между ними.

Расстояние между двумя прямыми.

☒ На плоскости

☐ В пространстве

Функция

Параметры (прямые)

$$y_1(x) = 7x + 2$$

$$y_2(x) = 7x + 14$$

Расстояние: 8,48528137423857

Выполнить

Расстояние между двумя прямыми.

☒ На плоскости

☐ В пространстве

Функция

Параметры (прямые)

$$y_1(x) = 7x + 2$$

$$y_2(x) = 1x + 3$$

Расстояние: 0. Прямые пересекаются

Выполнить

## Определение типа поверхности второго порядка

- Пользователь должен ввести коэффициенты поверхности в пространстве.
- Значения коэффициентов должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода коэффициентов при нажатии на кнопку «Выполнить» калькулятор определяет вид поверхности.

Определение поверхности второго порядка.

☐ На плоскости  
☒ В пространстве

Функция

Кривая второго порядка

$79x^2 + 6y^2 + 82xy + 0x + 0y - 2$   
 $+ 3z^2 - 3xz - 3yz + 7z$

$t_1: 88$   
 $t_2: -956,5$   
 $k_1: 0$   
 $k_2: 0$   
 $\delta: -3627,75$   
 $\Delta: 22041,25$

Однополостный гиперболоид

Выполнить

## Операции с векторами на плоскости

### Скалярное произведение

- Ввод 2-х векторов осуществляется по координатам (x, y).
- Пользователь должен вводить только целые числа или числа с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода двух векторов при нажатии на кнопку «Выполнить» калькулятор посчитает скалярное произведение

Операции с векторами на плоскости.

☒ На плоскости  
☐ В пространстве

Функция

Векторы

☒ Скалярное произведение

n1: { 8 ; 9 }

n2: { -2 ; 1 }

☐ Угол между векторами

n1: { ; }

n2: { ; }

Скалярное произведение: -7

Выполнить

### Угол между векторами

- Ввод 2-х векторов осуществляется по координатам (x, y).
- Пользователь должен вводить только целые числа или числа с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).
- После ввода двух векторов при нажатии на кнопку «Выполнить» калькулятор посчитает угол между векторами

Операции с векторами на плоскости.

☒ На плоскости  
☐ В пространстве

Функция

Векторы

☐ Скалярное произведение

n1: { 8 ; 9 }  
n2: { -2 ; 1 }

☒ Угол между векторами

n1: { 8 ; 8 }  
n2: { 2 ; 0 }

Угол между векторами: 45,001°

Выполнить

## Операции с векторами в пространстве

### Скалярное произведение

- Ввод 2-х векторов осуществляется по координатам (x, y, z).
- Пользователь должен вводить только целые числа или числа с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).
- После ввода двух векторов при нажатии на кнопку «Выполнить» калькулятор посчитает скалярное произведение.

Операции с векторами в пространстве.

☐ На плоскости  
☒ В пространстве

Функция

Векторы

☒ Скалярное произведение  
 n1: { 8 ; 9 ; 6 }  
 n2: { -2 ; 1 ; 6 }

☐ Угол между векторами  
 n1: { 8 ; 8 ; }  
 n2: { 2 ; 0 ; }

☐ Векторное произведение  
 n1: { ; ; }  
 n2: { ; ; }

☐ Смешанное произведение  
 n1: { ; ; }  
 n2: { ; ; }  
 n3: { ; ; }

Скалярное произведение: 29

Выполнить

### Угол между векторами

- Ввод 2-х векторов осуществляется по координатам (x, y, z).
- Пользователь должен вводить только целые числа или числа с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).
- После ввода двух векторов при нажатии на кнопку «Выполнить» калькулятор посчитает угол между векторами.



Операции с векторами в пространстве.

☐ На плоскости  
☒ В пространстве

Функция

Векторы

Скалярное произведение

☐ n1: { 8 ; 9 ; 6 }

n2: { -2 ; 1 ; 6 }

Векторное произведение

☐ n1: { ; ; }

n2: { ; ; }

Угол между векторами

☒ n1: { 4 ; 4 ; 9 }

n2: { 5 ; 4 ; -3 }

Смешанное произведение

☐ n1: { ; ; }

n2: { ; ; }

n3: { ; ; }

Угол между векторами: 83,125°

Выполнить

### Векторное произведение

- Ввод 2-х векторов осуществляется по координатам (x, y, z).
- Пользователь должен вводить только целые числа или числа с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).
- После ввода двух векторов при нажатии на кнопку «Выполнить» калькулятор посчитает векторное произведение.

Операции с векторами в пространстве.

☐ На плоскости  
☒ В пространстве

Функция

Векторы

☐ Скалярное произведение  
 n1: {  ;  ;  }  
 n2: {  ;  ;  }

☐ Угол между векторами  
 n1: {  ;  ;  }  
 n2: {  ;  ;  }

☒ Векторное произведение  
 n1: {  ;  ;  }  
 n2: {  ;  ;  }

☐ Смешанное произведение  
 n1: {  ;  ;  }  
 n2: {  ;  ;  }  
 n3: {  ;  ;  }

Векторное произведение: { 31 22 -67 }

Выполнить

### Смешанное произведение

- Ввод 3-х векторов осуществляется по координатам (x, y, z).
- Пользователь должен вводить только целые числа или числа с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).
- После ввода трех векторов при нажатии на кнопку «Выполнить» калькулятор посчитает смешанное произведение.

Операции с векторами в пространстве.

☐ На плоскости  
☒ В пространстве

Функция

Векторы

☐ Скалярное произведение  
 n1: {  ;  ;  }  
 n2: {  ;  ;  }

☐ Угол между векторами  
 n1: {  ;  ;  }  
 n2: {  ;  ;  }

☐ Векторное произведение  
 n1: { 8 ; 7 ; 6 }  
 n2: { 5 ; -4 ; 1 }

☒ Смешанное произведение  
 n1: { 6 ; 8 ; -3 }  
 n2: { 0 ; 7 ; 27 }  
 n3: { 7 ; 2 ; 1 }

Смешанное произведение: 1377

Выполнить

### Определение типа кривой второго порядка

- Пользователь должен ввести коэффициенты кривой на плоскости.
- Значения коэффициентов должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода коэффициентов при нажатии на кнопку «Выполнить» калькулятор определяет вид кривой.

Определение кривой второго порядка.

☒ На плоскости  
☐ В пространстве

Функция

Кривая второго порядка

$x^2$  +   $y^2$  +   $xy$  +   $x$  +   $y$  +

$\tau$ : 7  
 $\delta$ : 0  
 $\Delta$ : -1,75

Парабола

## Вычисление расстояний

### Расстояние между прямыми в пространстве

- Пользователь должен ввести две прямые, заданных каноническими уравнениями.
- Значения коэффициентов в уравнениях должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).
- После ввода двух прямых при нажатии на кнопку «Выполнить» калькулятор определяет расстояние между ними.

Расстояние между прямыми и плоскостями.

☐ На плоскости

☒ В пространстве

Функция

Расстояние: 0,352

2 прямые

L1:  $x - 6 = y - 8 = z - 6$

☒

$\frac{x-6}{2} = \frac{y-8}{-2} = \frac{z-6}{0}$

L2:  $x - 9 = y - 62 = z - 9$

$\frac{x-9}{8} = \frac{y-62}{9} = \frac{z-9}{1}$

Прямая и плоскость

☐

S:  $x + y + z +$

L:  $x - y = z -$

$\frac{x-}{ } = \frac{y-}{ } = \frac{z-}{ }$

2 плоскости

☐

S1:  $x + y + z +$

S2:  $x + y + z +$

Выполнить

Расстояние между прямыми и плоскостями.

☐ на плоскости

☒ В пространстве

Функция

Расстояние: 0

2 прямые

L1:  $x - 6 = y - 8 = z - 6$

☒

$\frac{x-6}{2} = \frac{y-8}{-2} = \frac{z-6}{0}$

L2:  $x - 6 = y - 8 = z - 6$

$\frac{x-6}{3} = \frac{y-8}{3} = \frac{z-6}{3}$

Прямая и плоскость

☐

S:  $3x + 0y + 8z + 72$

L:  $x - 2 = y - 3 = z - 4$

$\frac{x-2}{-3} = \frac{y-3}{12} = \frac{z-4}{3}$

2 плоскости

☐

S1:  $3x + 3y + 2z + 1$

S2:  $3x + 3y + 2z + 2$

Выполнить

## Расстояние между прямой и плоскостью в пространстве

- Пользователь должен ввести каноническое уравнение прямой и уравнение плоскости.
- Значения коэффициентов в уравнениях должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).
- После ввода прямой и плоскости при нажатии на кнопку «Выполнить» калькулятор определяет расстояние между ними.

Расстояние между прямыми и плоскостями.

Функция

☐ На плоскости  
☒ В пространстве

2 прямые

L1:  $x - 6 = y - 8 = z - 6$

☐  $\frac{x-6}{2} = \frac{y-8}{-2} = \frac{z-6}{0}$

L2:  $x - 9 = y - 62 = z - 9$

$\frac{x-9}{8} = \frac{y-62}{9} = \frac{z-9}{1}$

Прямая и плоскость

☒ S:  $3x + 0y + 8z + 72$

L:  $x - 2 = y - 3 = z - 4$

$\frac{x-2}{-3} = \frac{y-3}{12} = \frac{z-4}{3}$

2 плоскости

☐ S1:  $\square x + \square y + \square z + \square$

S2:  $\square x + \square y + \square z + \square$

Выполнить

Расстояние: 12,875

## Расстояние между плоскостями в пространстве

- Пользователь должен ввести две плоскости с помощью уравнений.
- Значения коэффициентов в уравнениях должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).
- После ввода двух плоскостей при нажатии на кнопку «Выполнить» калькулятор определяет расстояние между ними.

Расстояние между прямыми и плоскостями.

☐ На плоскости

☒ В пространстве

Функция

Эти плоскости не параллельны

2 прямые

L1:  $x - 6 = y - 8 = z - 6$

☐

$\frac{x-6}{2} = \frac{y-8}{-2} = \frac{z-6}{0}$

L2:  $x - 9 = y - 62 = z - 9$

$\frac{x-9}{8} = \frac{y-62}{9} = \frac{z-9}{1}$

Прямая и плоскость

☐

S:  $3x + 0y + 8z + 72$

L:  $x - 2 = y - 3 = z - 4$

$\frac{x-2}{-3} = \frac{y-3}{12} = \frac{z-4}{3}$

2 плоскости

☒

S1:  $-4x + 3y + 2z + 1$

S2:  $3x + 4y + 1z + 2$

Выполнить

Расстояние между прямыми и плоскостями.

☐ На плоскости

☒ В пространстве

Функция

Расстояние между S1 и S2: 0,213

2 прямые

L1:  $x - 6 = y - 8 = z - 6$

☐

$\frac{x-6}{2} = \frac{y-8}{-2} = \frac{z-6}{0}$

L2:  $x - 9 = y - 62 = z - 9$

$\frac{x-9}{8} = \frac{y-62}{9} = \frac{z-9}{1}$

Прямая и плоскость

☐

S:  $3x + 0y + 8z + 72$

L:  $x - 2 = y - 3 = z - 4$

$\frac{x-2}{-3} = \frac{y-3}{12} = \frac{z-4}{3}$

2 плоскости

☒

S1:  $3x + 3y + 2z + 1$

S2:  $3x + 3y + 2z + 2$

Выполнить

## ***Блок “Дискретная математика”***

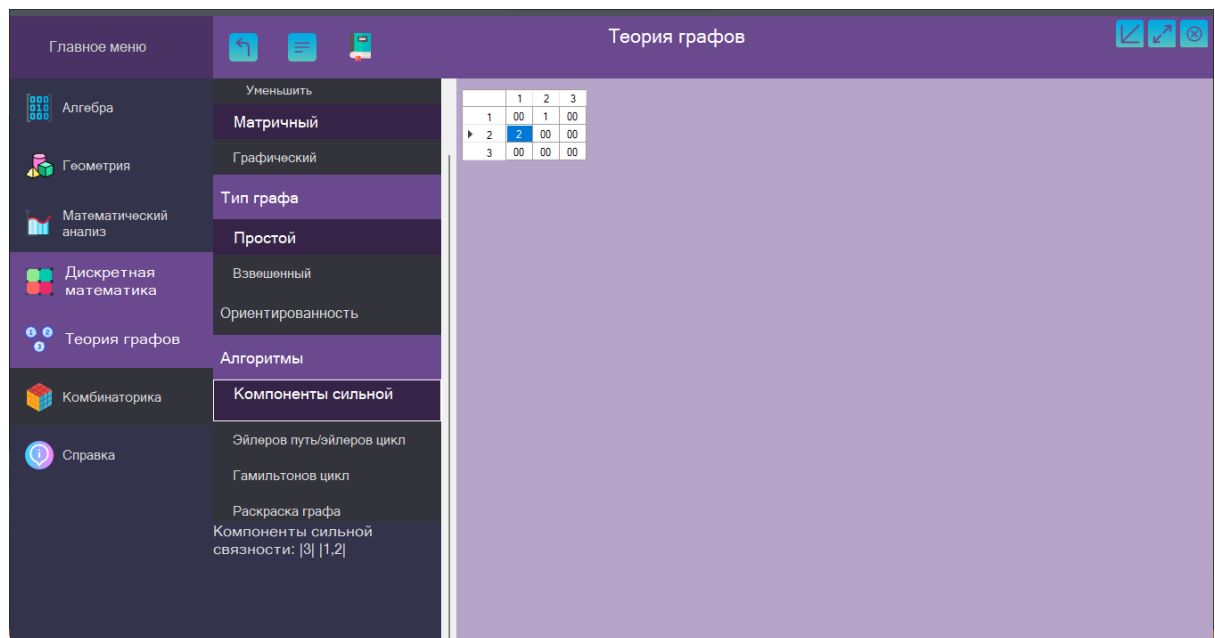
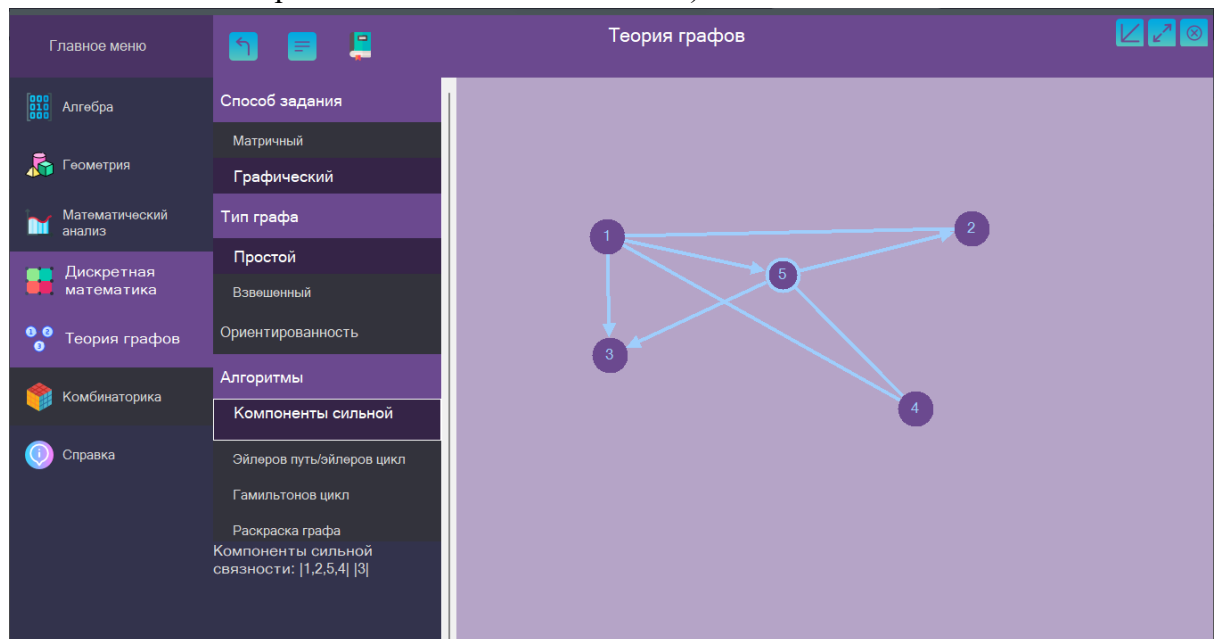
### ***Подраздел “Теория графов”***

- Пользователь выбирает “Способ задания” и “Тип графа”.
- **При Графическом способе задания, чтобы появился доступ к алгоритмам необходимо добавить хотя бы одну вершину!**
- При “Графическом” способе задания и выборе “Простого” типа графа пользователь может строить граф графически: ЛКМ на свободное от вершины место - добавление вершины к графу (вершины не могут накладываться друг на друга), ЛКМ на вершину - выделение вершины, ЛКМ на выделенную вершину - снять выделение с вершины, ЛКМ при выделенной вершине по другой из вершин - добавление ребра, ПКМ при выделенной вершине по другой из вершин - удаление ребра, SHIFT+ЛКМ - удаление указанной вершины.
- При “Графическом” способе задания и выборе “Взвешенного” типа графа граф строится аналогично “Простому типу” за исключением того, что добавить можно только одно ребро, а изменить вес: выделить вершину и ПКМ по вершине, у которой хотите поменять вес.
- При “Графическом” способе задания и выборе “Ориентированности” (по умолчанию задана “Неориентированная” ориентированность), можно динамически изменять тип ориентации дуги.
- При “Матричном” способе задания и выборе одного из типов графа можно увеличить и уменьшить размер матрицы (кнопки “Увеличить” и “Уменьшить” соответственно). Если дуги из вершины не существует, тогда задается значения “00”.
- После ввода матрицы либо графического задания графа можно выбрать нужный вам алгоритм (кнопка “Алгоритмы”)
- Чтобы можно было добавлять вершины графически или вводить в матрицу необходимо выбрать пункты в разделе “Способ задания” и “Тип графа”.

### **Компоненты сильной связности**



После нажатия в нижнем окне слева(под алгоритмами) появится ответ(“||” данные символы отделяют разные компоненты связности).



**Эйлеров путь/цикл**

После нажатия в нижнем окне слева(под алгоритмами) появится ответ: если первая и последняя вершины совпадают, то это Эйлеров цикл,если нет, то это эйлеров путь.

Главное меню

- Алгебра
- Геометрия
- Математический анализ
- Дискретная математика
- Теория графов
- Комбинаторика
- Справка

Теория графов

Способ задания

- Матричный
- Графический

Тип графа

- Простой
- Взвешенный
- Ориентированность

Алгоритмы

- Компоненты сильной связности
- Эйлеров путь/эйлеров
- Гамильтонов цикл
- Раскраска графа

Эйлеров путь/цикл: 1=>3=>4=>2=>1

Главное меню

- Алгебра
- Геометрия
- Математический анализ
- Дискретная математика
- Теория графов
- Комбинаторика
- Справка

Теория графов

Уменьшить

Матричный

Графический

Тип графа

- Простой
- Взвешенный
- Ориентированность

Алгоритмы

- Компоненты сильной связности
- Эйлеров путь/эйлеров
- Гамильтонов цикл
- Раскраска графа

Эйлеров путь/цикл: 2=>1=>3

	1	2	3
1	00	1	1
2	1	00	00
3	1	00	00

## Гамильтонов цикл

После нажатия в нижнем окне слева(под алгоритмами) появится ответ путь.

The image displays two screenshots of a software application titled "Теория графов" (Graph Theory). The interface includes a sidebar menu with categories like "Главное меню", "Алгебра", "Геометрия", "Математический анализ", "Дискретная математика", "Теория графов", "Комбинаторика", and "Справка". The "Теория графов" section is expanded, showing options for "Способ задания" (Matrix, Graphical), "Тип графа" (Simple, Weighted, Directed), and "Алгоритмы" (Strongly connected components, Eulerian path/cycle, Hamiltonian cycle, Graph coloring).

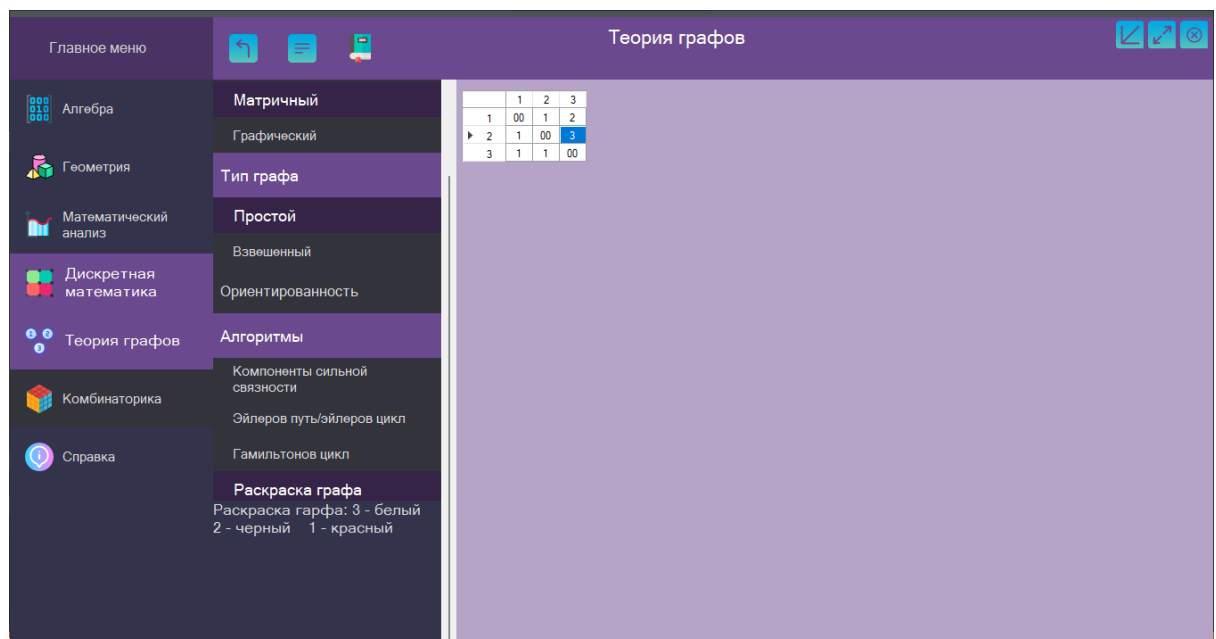
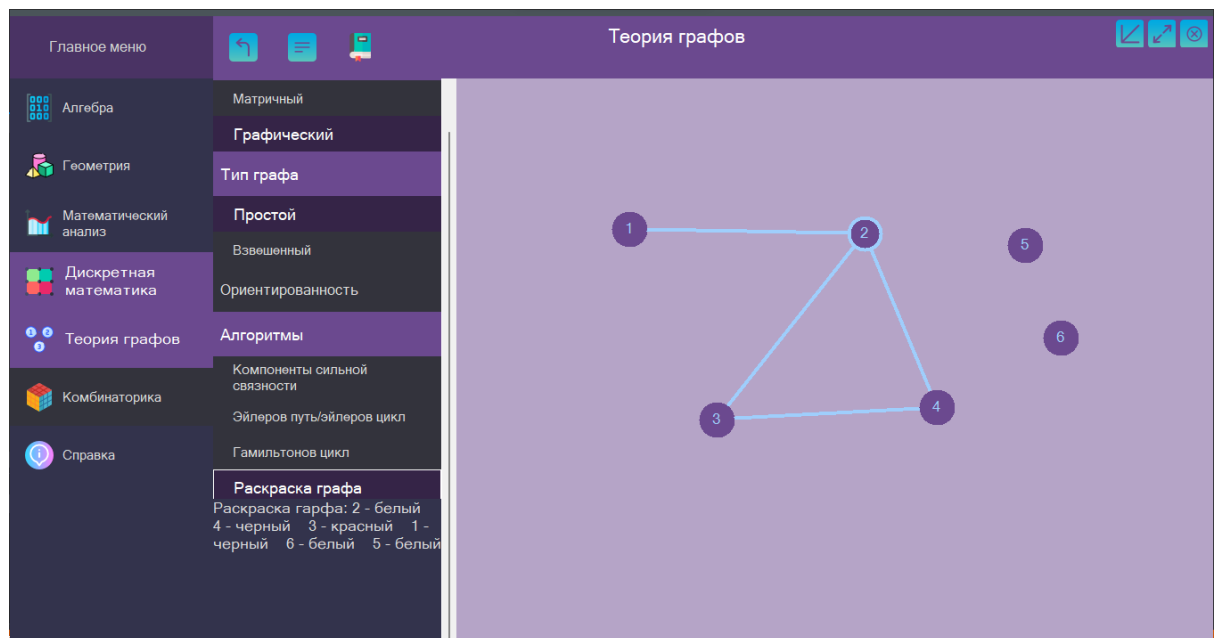
**Top Screenshot:** The "Алгоритмы" section is selected, showing "Гамильтонов цикл" (Hamiltonian cycle) with the result: "Раскраска графа Гамильтонов цикл: 1=>4=>2=>3=>1". The main area displays a directed graph with 4 nodes (1, 2, 3, 4) and edges forming a cycle: 1 → 4 → 2 → 3 → 1.

**Bottom Screenshot:** The "Алгоритмы" section is selected, showing "Гамильтонов цикл" (Hamiltonian cycle) with the result: "Раскраска графа Гамильтонов цикл: 1=>2=>3=>1". The main area displays a matrix view of the graph, which is a 3x3 adjacency matrix for a directed graph with 3 nodes (1, 2, 3):

	1	2	3
1	00	1	1
2	1	00	1
3	1	1	00

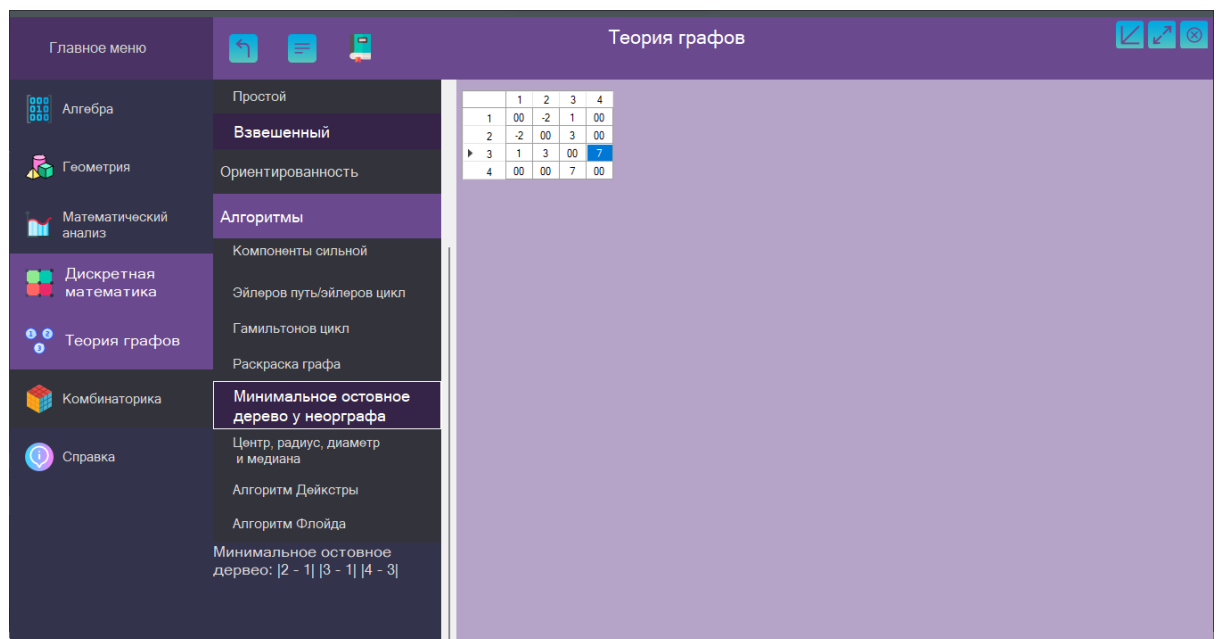
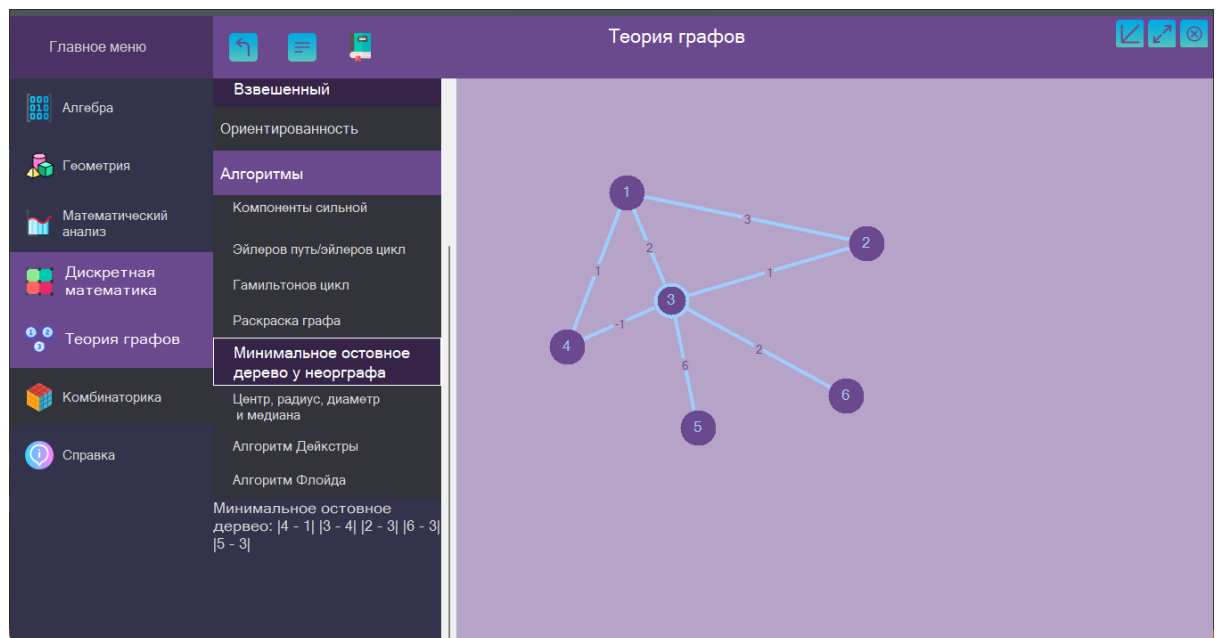
## Раскраска графа

После нажатия в нижнем окне слева(под алгоритмами) появится ответ: первая цифра - это номер вершины, за ней через "-" идет ее цвет. Реализована с помощью жадного алгоритма, поэтому может не выдать минимального числа цветов в ответе, а выдать раскраску с небольшим количеством цветов.



### Минимальное остовное дерево у неграфа (для взвешенного типа)

После нажатия в нижнем окне слева(под алгоритмами) появится ответ: ребра, из которых состоит минимальный остов(в “||” содержится номера вершин, то есть ребро). Реализован с помощью алгоритма Прима.



### Центр, радиус, диаметр и медиана графа (для взвешенного типа)

После нажатия в нижнем окне слева(под алгоритмами) появится ответ. Если в графе существует отрицательный цикл, то выведется ошибка, т. к. минимальные расстояния находятся по алгоритму Флойда.

Главное меню

Алгебра

Геометрия

Математический анализ

Дискретная математика

Теория графов

Комбинаторика

Справка

Теория графов

Простой

Взвешенный

Ориентированность

Алгоритмы

Компоненты сильной

Эйлеров путь/эйлеров цикл

Гамильтонов цикл

Раскраска графа

Минимальное остовное дерево у неорграфа

Центр, радиус, диаметр

Алгоритм Дейкстры

Алгоритм Флойда

Центр : 3 , Радиус: 7, Диаметр: 9, Медиана: 3

	1	2	3	4
1		00	3	1 00
2			00	3 00
3		1	2	00 7
4		00	00	7 00

Главное меню

Алгебра

Геометрия

Математический анализ

Дискретная математика

Теория графов

Комбинаторика

Справка

Теория графов

Ориентированный

Неориентированный

Алгоритмы

Компоненты сильной

Эйлеров путь/эйлеров цикл

Гамильтонов цикл

Раскраска графа

Минимальное остовное дерево у неорграфа

Центр, радиус, диаметр

Алгоритм Дейкстры

Алгоритм Флойда

Центр : 2 5 , Радиус: 3, Диаметр: 4, Медиана: 1

### Алгоритм Дейкстры, Флойда (для взвешенного типа)

При графическом задании нужно выделить две вершины, для которых необходимо найти минимальный путь, чтобы снять выделение и добавлять вершины дальше необходимо нажать на “Алгоритмы” либо на любой другой алгоритм. При матричном - написать номера данных вершин через пробел. В ответе: первое число - длина минимального пути, второе - кратчайший путь.

Главное меню

Алгебра

Геометрия

Математический анализ

Дискретная математика

Теория графов

Комбинаторика

Справка

Ориентированность

Алгоритмы

Компоненты сильной

Эйлеров путь/эйлеров цикл

Гамильтонов цикл

Раскраска графа

Минимальное остовное дерево у неорграфа

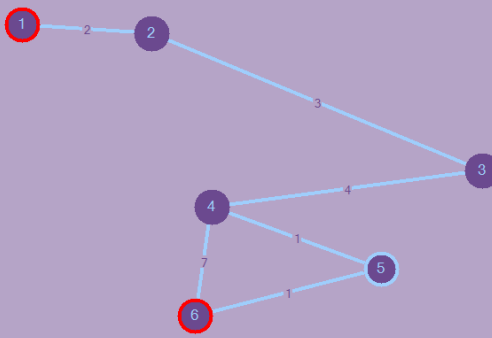
Центр, радиус, диаметр и медиана

Алгоритмы Дейкстры

Алгоритм Флойда

Алгоритм Дейкстры: 11 1-2-3 -4-5-6

Теория графов



Главное меню

Алгебра

Геометрия

Математический анализ

Дискретная математика

Теория графов

Комбинаторика

Справка

Неориентированный

Алгоритмы

Компоненты сильной

Эйлеров путь/эйлеров цикл

Гамильтонов цикл

Раскраска графа

Минимальное остовное дерево у неорграфа

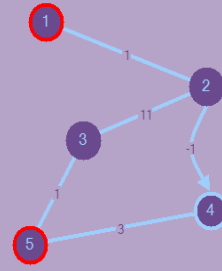
Центр, радиус, диаметр и медиана

Алгоритмы Дейкстры

Алгоритм Флойда

Алгоритм Флойда: 3 1-2-4 -5

Теория графов



Главное меню

Алгебра

Геометрия

Математический анализ

Дискретная математика

Теория графов

Комбинаторика

Справка

Неориентированный

Алгоритмы

Компоненты сильной

Эйлеров путь/эйлеров цикл

Гамильтонов цикл

Раскраска графа

Минимальное остовное дерево у неорграфа

Центр, радиус, диаметр и медиана

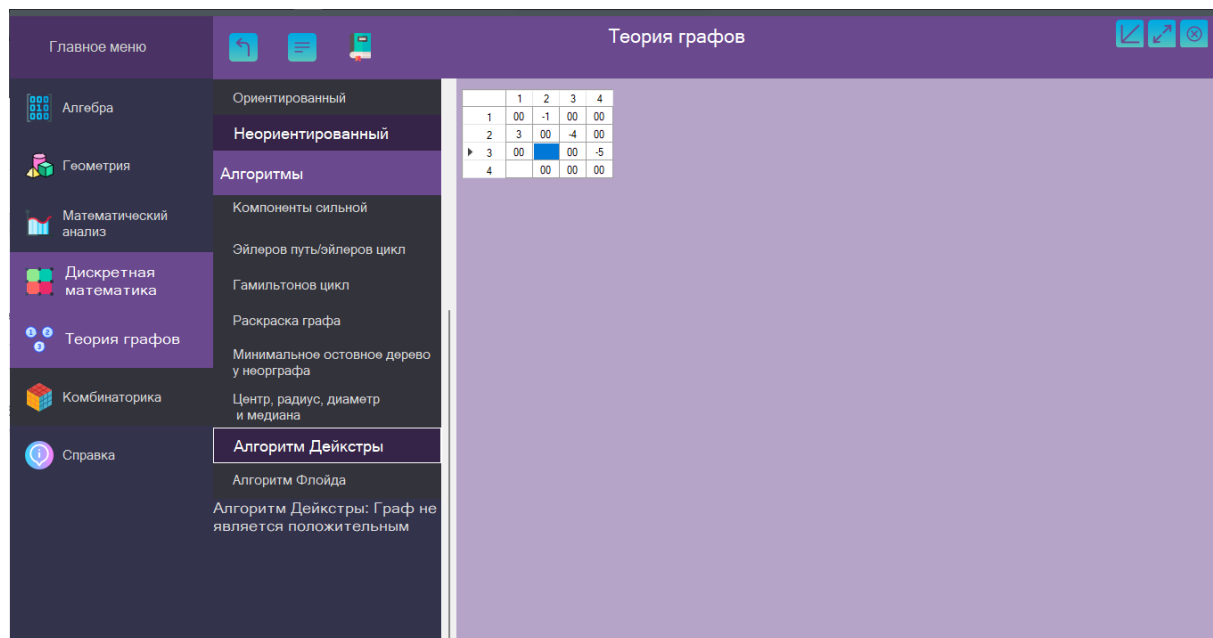
Алгоритмы Дейкстры

Алгоритм Флойда

Алгоритм Флойда: -10 1-2 -3-4

Теория графов

	1	2	3	4
1	00	-1	00	00
2	3	00	-4	00
3	00		00	-5
4		00	00	00



### ***Подраздел “Комбинаторика”***

После выбора нужной формулы и ввода нужно нажать “Вычислить” в нижнем левом углу. Если ввод был некорректен, то в ответ выведется “Ошибка”, если же результат превышает 9000!, то выведется “Слишком большое число”



Главное меню

Алгебра

Геометрия

Математический анализ

Дискретная математика

Теория графов

Комбинаторика

Справка

Вычислить

Комбинаторика

Сочетания

Без повторений

С повторениями

Размещения

Перестановки

$C_{5}^3 =$

10

Главное меню

Алгебра

Геометрия

Математический анализ

Дискретная математика

Теория графов

Комбинаторика

Справка

Вычислить

Комбинаторика

Сочетания

Без повторений

С повторениями

Размещения

Перестановки

$C_{5}^3 =$

35

Главное меню

Алгебра

Геометрия

Математический анализ

Дискретная математика

Теория графов

Комбинаторика

Справка

Вычислить

Комбинаторика

Сочетания

Размещения

Без повторениями

С повторениями

Перестановки

$$A_{7,4} = 840$$

Главное меню

Алгебра

Геометрия

Математический анализ

Дискретная математика

Теория графов

Комбинаторика

Справка

Вычислить

Комбинаторика

Сочетания

Размещения

Без повторениями

С повторениями

Перестановки

$$A_{7,4} = 5764801$$

Главное меню

Алгебра

Геометрия

Математический анализ

Дискретная математика

Теория графов

Комбинаторика

Справка

Сочетания

Размещения

Без повторениями

С повторениями

Перестановки

Вычислить

Комбинаторика

$$A_{7,4} = 2401$$

Главное меню

Алгебра

Геометрия

Математический анализ

Дискретная математика

Теория графов

Комбинаторика

Справка

Сочетания

Размещения

Перестановки

Без повторений

С повторениями

Вычислить

Комбинаторика

$$P_{81} = 5797126020747$$

Главное меню

Алгебра

Геометрия

Математический анализ

Дискретная математика

Теория графов

Комбинаторика

Справка

Сочетания

Размещения

Перестановки

Без повторений

С повторениями

Вычислить

Комбинаторика

$$P_{1237} = 102960$$

## Блок “Алгебра”

### Подраздел “Алгебра матриц”

#### Арифметические операции

##### Сумма, разность, произведение

При нажатии на кнопку “Арифметические операции” открывается форма, где показаны первый операнд и второй операнд, по умолчанию матрицы второго порядка заполненные нулями. Для каждой матрицы внизу есть три кнопки, позволяющие увеличить, уменьшить размер и очистить матрицу соответственно.

Первый операнд	Второй операнд								
<table><tr><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td></tr></table>	0	0	0	0	<table><tr><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td></tr></table>	0	0	0	0
0	0								
0	0								
0	0								
0	0								
<div><div>+</div><div>−</div><div>🗑</div></div>	<div><div>+</div><div>−</div><div>🗑</div></div>								

В окошки ввода можно вводить только комбинации из цифр, знаков {+, -} и запятой(именно запятой, а не точки). При нарушении этих правил ввода возникает исключение.


Первый операнд

3	qwe
+12	2.2

Второй операнд

12	4
3	2

Ошибка

 Введите число с запятой

ОК

Также исключение возникает, если у операндов размеры не совпадают:


Первый операнд

2	1	3
2	1	3
2	1	3

Второй операнд

5	2
6	-33

Ошибка

 Матрицы разных размеров

ОК

Примеры правильной работы арифметических операций приведены ниже:

Первый операнд		Второй операнд		Сумма	
3	-3,12	-12	4	-9	0,88
+12	2,21	3,012	2,01	15,012	4,22

Первый операнд		Второй операнд		Разность	
3	-3,12	-12	4	15	-7,12
+12	2,21	3,012	2,01	8,988	0,2

Первый операнд		Второй операнд		Произведение	
1	2	5	2	17	-64
3	4	6	-33	39	-126

## LU разложение

Исключение выбрасывается при вводе вырожденной матрицы, т.е. той, у которой детерминант равен нулю. Для класса таких матриц LU-разложения не существует.

Входная матрица


1	2	3
4	5	6
7	8	9

Результат: LU-разложение

1	2	3
0	-3	-6
0	0	2

1	0	0
4	1	0
4	2	1

Ошибка



LU-разложения не существует, так как матрица вырождена

OK


Корректность ввода также обрабатывается, как упоминалось выше, для любых функций. Для вещественных чисел используется также запятая, а не точка! (В дальнейшем это демонстрироваться уже не будет).

Входная матрица

е	х	е
z	i	p
г	а	г

Результат:

Ошибка


Введите число с запятой


ОК

Входная матрица

0.2	1.2	3.2
0.1	0.2	0.3
0.1	-0.1	0.3

Результат:

Ошибка


Введите число с запятой

ОК

Пример правильной работы функции LU-разложения приведен ниже. После ввода матрицы справа появляются две матрицы: первая матрица - L, вторая матрица - U.

Входная матрица

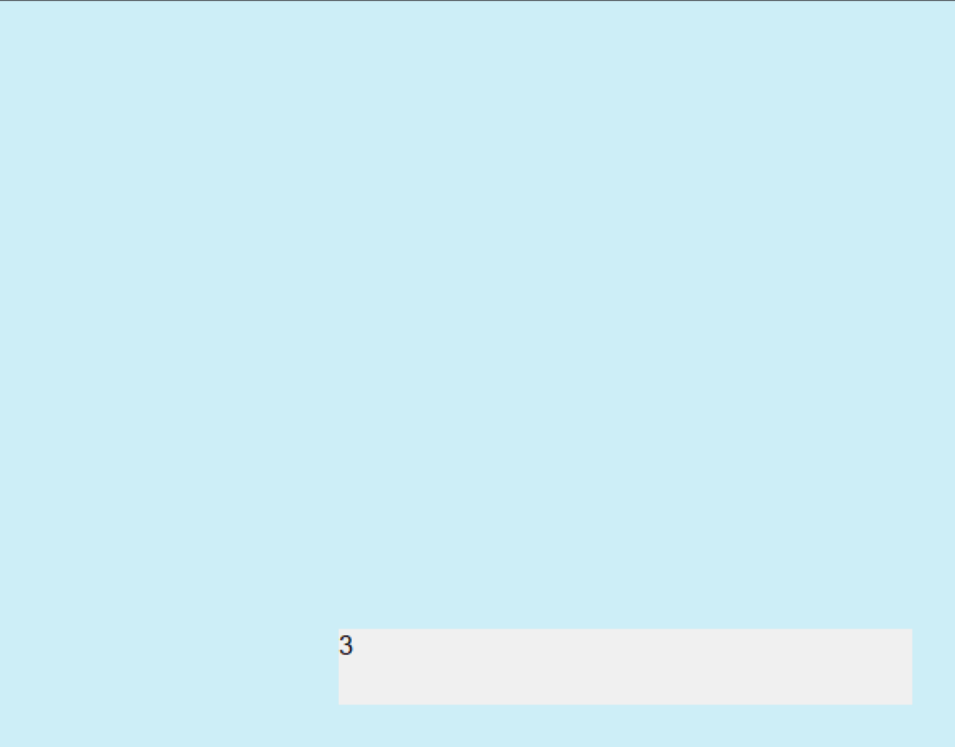
1	2	3
4	5	6
4	2	2

Результат: LU-разложение

1	2	3
0	-3	-6
0	0	2

1	0	0
4	1	0
4	2	1

# Определитель

Входная матрица			Результат	Определитель
1	2	3		3
4	5	6		
7	8	8		

## Ступенчатый вид

Входная матрица			Результат	Ступенчатый вид	
1	2	3	1	2	3
4	5	6	0	-3	-6
7	8	8	0	0	-1



Ранг

Входная матрица

Результат

Ранг

1	2	3
1	2	3
7	8	8

2

Определенность матрицы

Входная матрица

Результат

Определённость матрицы

1	2	3
0	2	3
0	0	1

Матрица положительно определена

Транспонированная матрица

Входная матрица			Результат			Транспонированная матрица		
1	2	3		1	0	0		
0	2	3		2	2	0		
0	0	1		3	3	1		

Обратная матрица

Входная матрица			Результат			Обратная матрица		
1	2	3		-1	1	0		
2	2	3		-5	5,5	-1,5		
4	0	1		4	-4	1		

Пример правильного нахождения обратной матрицы приведен выше. Ниже приведен пример, когда выбрасывается исключение необратимости матрицы в силу равенства нулю ее детерминанта.

Входная матрица

1	2	3
-2	-4	-6
1	2	1

Результат

Обратная матрица

Ошибка

✖

Матрица необратима

OK

Степень матрицы

Входная матрица			Результат			Степень матрицы 3		
-1	2	-3		-468	576	-684		
4	-5	6		1062	-1305	1548		
-7	8	-9		-1656	2034	-2412		


Пример правильного нахождения степени матрицы приведен выше. Степень должна быть целым положительным числом, иначе выбрасывается исключение, как в примере ниже при вводе степени -4.

Входная матрица

1	2	-3
6	5	4
-8	-9	7

Результат

Ошибка



Степень должна быть целым положительным числом

OK

Введите степень: -4

## Умножение на число

Входная матрица			Результат			Умножение на число -2,32		
1	2	3	-2,32	-4,64	-6,96			
4	5	6	-9,28	-11,6	-13,92			
-1	8	9	2,32	-18,56	-20,88			

Введите степень: 
  
Введите число:

### ***Подраздел “Алгебра многочленов”***

Сразу стоит отметить, что при некорректном вводе пользователь будет осведомлен об этом. Если оставить поле без значения, то это поле будет считаться с значением ноль. Примеры ошибок и их обработка:

- Ввод нулевого полинома:

### Калькулятор многочленов

0

$x^6 +$

0

$x^5 +$

0

$x^4 +$

0

$x^3 +$

0

$x^2 +$

0

$x +$

0

▼

$x^5 +$

$x^4 +$

$x^3 +$

$x^2 +$

$x +$

=

Результат

### Калькулятор многочленов

$x^6 +$

$x^5 +$

$x^4 +$

$x^3 +$

$x^2 +$

$x +$

▼

$x^5 +$

$x^4 +$

$x^3 +$

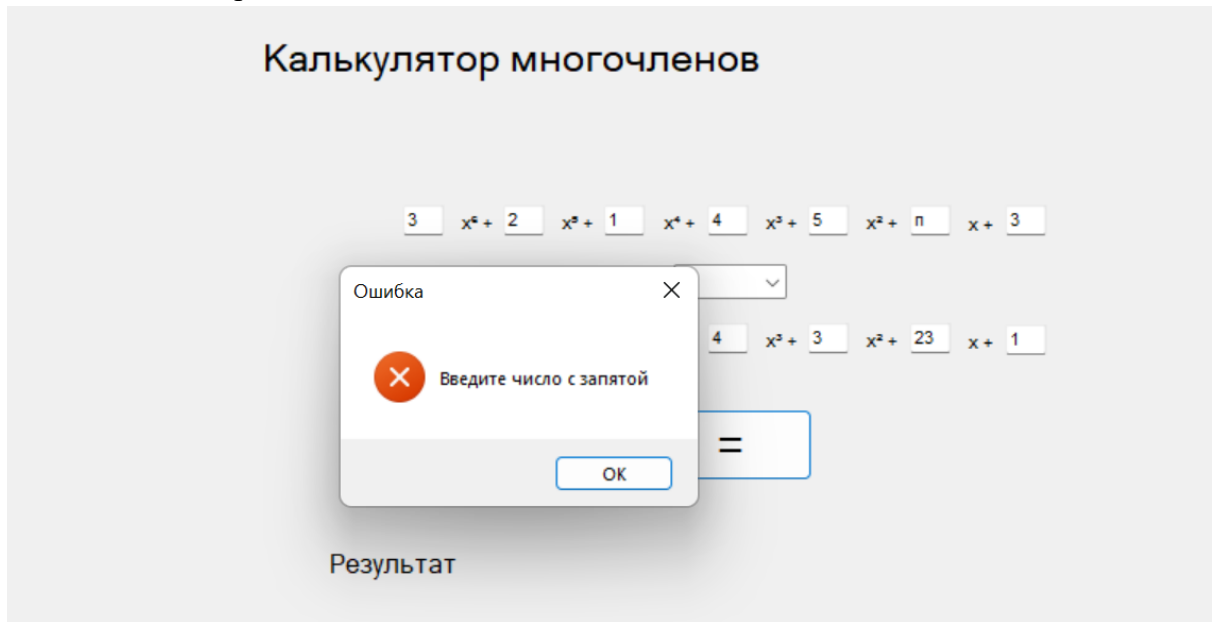
$x^2 +$

$x +$

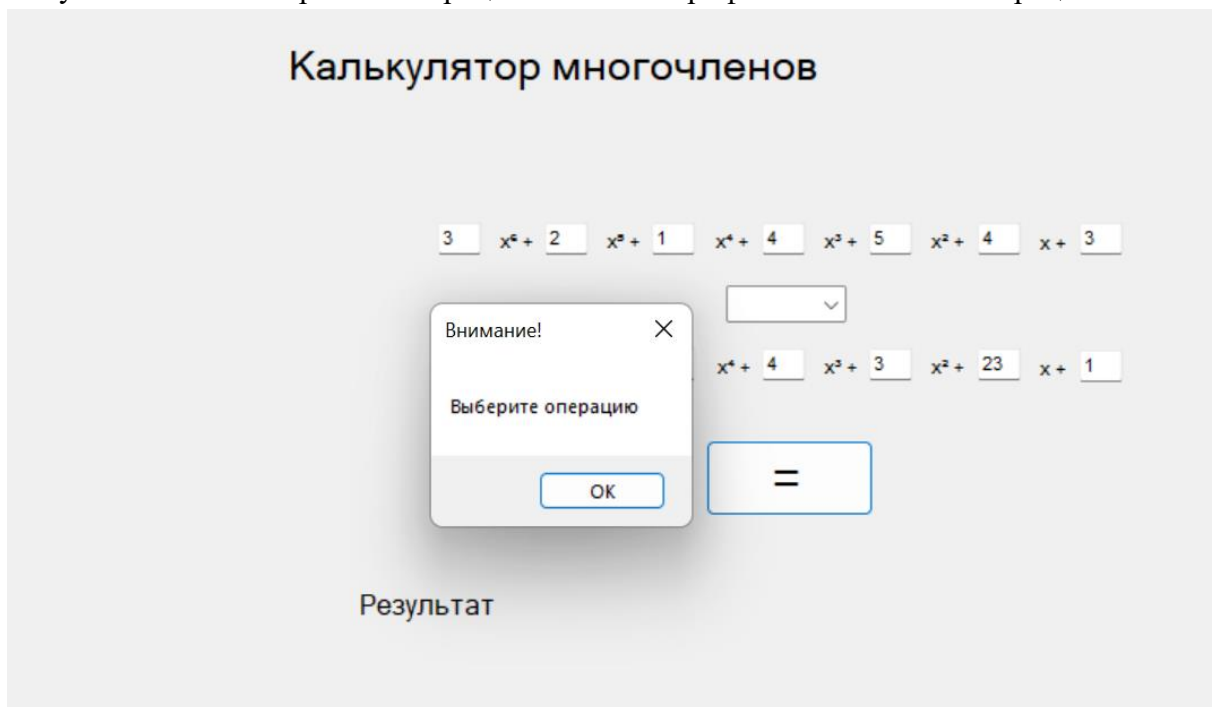
=

Результат

- Ввод всего кроме целого числа или числа с плавающей точкой:



- Отсутствие выбора операции в арифметических операциях:



- Введение полиномов, степень которых не 2, 3 или 4:

### Арифметические операции над полиномами

- Пользователь должен ввести коэффициенты двух полиномов при соответствующих степенях  $x$ . При этом если не заполнять поля коэффициента, калькулятор автоматически распознает это как 0.
- Нельзя проводить никакие операции, если хотя бы один полином нулевой.

- Также необходимо выбрать тип операции, которая будет производиться над полиномами.
- Максимальная степень полинома – шестая.
- Значения коэффициентов полиномов должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки запятую).
- После ввода двух полиномов и выбора операции при нажатии на кнопку «=» калькулятор выводит результат операции над полиномами.

### Сложение

**Калькулятор многочленов**

$x^6 +$    $x^5 +$    $x^4 +$    $x^3 +$    $x^2 +$    $x +$

$x^6 +$    $x^5 +$    $x^4 +$    $x^3 +$    $x^2 +$    $x +$

$7x^6 + 4x^5 + 4x^4 - 6x^3 + 8x^2 - 21x + 4$

### Вычитание

## Калькулятор многочленов

$x^6 +$    $x^5 +$    $x^4 +$    $x^3 +$    $x^2 +$    $x +$

-

$x^6 +$    $x^5 +$    $x^4 +$    $x^3 +$    $x^2 +$    $x +$

=

$26x^6 - 2x^4 + 4x^3 - 3x^2 + 25x + 2$

### Умножение

## Калькулятор многочленов

$x^6 +$    $x^5 +$    $x^4 +$    $x^3 +$    $x^2 +$    $x +$

x

$x^6 +$    $x^5 +$    $x^4 +$    $x^3 +$    $x^2 +$    $x +$

=

$4x^{10} + 6x^9 + 8x^8 + 18x^7 - 42x^6 + 26x^5 - 83x^4 + 10x^3 - 37x^2 - 67x + 3$



## Деление

### Калькулятор многочленов

$x^6 +$    $x^5 +$    $x^4 +$    $x^3 +$    $x^2 +$    $x +$

$\div$

$x^6 +$    $x^5 +$    $x^4 +$    $x^3 +$    $x^2 +$    $x +$

$=$

Частное:  $1,2x^2 - 0,4x + 3,6$

Остаток:  $-23,2x^2 + 38,4x - 0,6$

## НОД

### Калькулятор многочленов

$x^6 +$    $x^5 +$    $x^4 +$    $x^3 +$    $x^2 +$    $x +$

$\text{НОД}$

$x^6 +$    $x^5 +$    $x^4 +$    $x^3 +$    $x^2 +$    $x +$

$=$

НОД:  $1x - 1$

## НОК

Калькулятор многочленов

$x^6 +$    $x^5 +$    $x^4 +$    $x^3 +$    $x^2 +$    $x +$

НОК

$x^6 +$    $x^5 +$    $x^4 +$    $x^3 +$    $x^2 +$    $x +$

НОК:  $0,25x^6 + 0,6875x^5 + 1,25x^4 + 1,875x^3 + 1,25x^2 + 0,6875x + 0,25$

## Поиск корней полиномов

- Пользователь должен ввести коэффициенты полинома при соответствующих степенях  $x$ . При этом если не заполнять поля коэффициента, калькулятор автоматически распознает это как 0.
- Нельзя найти корни многочлена если его степени отлична от 2-ой, 3-ей, 4-ой.
- Значения коэффициентов полиномов должны быть целыми числами или числами с плавающей точкой (при вводе чисел с плавающей точкой использовать вместо точки **запятую**).

После ввода полинома при нажатии на кнопку « $\Rightarrow$ » калькулятор вычисляет корни полинома.

## Уравнение 2-ой степени

Поиск корней полинома (2, 3, 4 степени).

$$\square x^6 + \square x^5 + \square x^4 + \square x^3 + \underline{4} x^2 + \underline{2} x + \underline{-34}$$

=

x<sub>1</sub>: 2,676

x<sub>2</sub>: -3,17617497767991

Уравнение 3-ей степени

Поиск корней полинома (2, 3, 4 степени).

$$\square x^6 + \square x^5 + \square x^4 + \underline{5} x^3 + \underline{4} x^2 + \underline{2} x + \underline{-34}$$

=

$5x^3 + 4x^2 + 2x - 34$

x<sub>1</sub>: 1,60150257080649

x<sub>2</sub>: -1,20075128540325

x<sub>3</sub>: 1,67457722770479

### Уравнение 4-ой степени(вещественные)

Поиск корней полинома (2, 3, 4 степени).

$$\boxed{\phantom{0}} x^4 + \boxed{\phantom{0}} x^3 + \boxed{6} x^2 + \boxed{-43} x + \boxed{42} x^2 + \boxed{1} x + \boxed{-6}$$

$$=$$

x<sub>1</sub>: 6  
x<sub>2</sub>: 1  
x<sub>3</sub>: 0,5  
x<sub>4</sub>: -0,333

### Подраздел “Комплексные числа”

Для операций с комплексными числами, отдельно вводятся вещественные и мнимые части в соответствующие окошки. Обработка некорректного ввода присутствует так же, как и в подразделе “Алгебра матриц”.

#### Степень комплексного числа

Введённое число	Степень	Результат	
2,2 + -3 i	3	-48,752+-16,56i	<button>Вычислить</button>

Выше представлен пример правильного вычисления степени комплексного числа. Степень должна быть целым положительным числом, иначе выбрасывается исключение, которое можно видеть на примере ниже при вводе степени -8.

Введённое число	Степень	Результат
2,2 + -3 i	-8	

Ошибка

Степень должна быть целым положительным числом

OK

### Алгебраический корень из комплексного числа

Введённое число	Степень корня	Результат
0 + 1 i	1	(1) 0+1i

Вычислить

Введённое число	Степень корня	Результат
-1 + 2 i	4	(1) 1,0681+0,5955i (2) -0,5955+1,0681i (3) -1,0681+-0,5955i (4) 0,5955+-1,0681i

Вычислить

Выше представлены примеры правильного вычисления корня из комплексного числа. Степень должна быть целым положительным числом, иначе выбрасывается исключение, которое можно видеть на примере ниже при вводе степени -3.

Введённое число	Степень корня	Результат
3 + 3 i	-3	

Вычислить

Ошибка

Степень должна быть целым положительным числом

OK

## Модуль комплексного числа

Введённое число	Результат
<input type="text" value="8"/> + <input type="text" value="6"/> i	<input type="text" value="10"/>
<input type="button" value="Вычислить"/>	

Введённое число	Результат
<input type="text" value="4"/> + <input type="text" value="-4"/> i	<input type="text" value="5,65685424949238"/>
<input type="button" value="Вычислить"/>	

## Сумма, разность и произведение комплексных чисел

Для вычисления арифметической операции нужно ввести два комплексных числа в соответствующих окошках и выбрать одну из 4 операций. В вышеперечисленных трех функциях исключительных ситуаций кроме некорректного ввода не возникает. Ниже приведены примеры правильной работы этих функций.

Первый операнд	Второй операнд	Сумма
<input type="text" value="0,1"/> + <input type="text" value="-2,3"/> i	<input type="text" value="3"/> + <input type="text" value="0,25"/> i	<input type="text" value="3,1+-2,05i"/>

Первый операнд	Второй операнд	Разность
<input type="text" value="0,1"/> + <input type="text" value="-2,3"/> i	<input type="text" value="3"/> + <input type="text" value="0,25"/> i	<input type="text" value="-2,9+-2,55i"/>

Первый операнд	Второй операнд	Произведение
<input type="text" value="0,1"/> + <input type="text" value="-2,3"/> i	<input type="text" value="3"/> + <input type="text" value="0,25"/> i	<input type="text" value="0,875+-6,875i"/>

## Частное комплексных чисел

Исключение возникает при делении на нулевое комплексное число, т.е. на число мнимое и вещественная части которого одновременно равны нулю. В остальных случаях исключений не возникает. Ниже приведены примеры корректной работы функции и исключительной ситуации.

Первый операнд	Второй операнд	Частное
0,1 + -2,3 i	3 + 0,25 i	-0,0303+-0,7641i

Первый операнд	Второй операнд	Частное
0,1 + -2,3 i	0 + 0 i	

Ошибка

Деление невозможно

ОК

## Подраздел “Теория чисел”

### Число делителей

Введённое число	Число делителей
1296	24

Выше приведен пример корректного вычисления числа делителей. Введенное число должно быть положительным, начиная с единицы (единица не представляет интереса в этом плане), иначе возникает исключение: пример ниже.

Введённое число	Число делителей
-888	

Ошибка

Введите положительное число

ОК

Введённое число	Число делителей
1	

Ошибка

Делителей нет

ОК

Введённое число	Число делителей
0	

Ошибка  
Делителей нет  
OK

## Разложение на простые множители

Аналогичная ситуация с разложением на простые множители.

Введённое число	Разложение на простые множители
1296	$2^4 * 3^4$

Введённое число	Разложение на простые множители
-888	

Ошибка  
Введите положительное число  
OK

Введённое число	Разложение на простые множители
1	

Ошибка  
Единицу или ноль нельзя разложить на простые множители  
OK

Введённое число	Разложение на простые множители
0	

Ошибка  
Единицу или ноль нельзя разложить на простые множители  
OK



## Перевод рационального числа в цепную дробь

По правилам математики число является рациональным, если его можно представить в виде отношения целого числа к натуральному числу, но в этой функции знаменатель может быть отрицательным - в таком случае значение знаменателя конвертируется в положительное значение умножением дроби на -1 внутри программы.

Чтобы получить цепную дробь нужно ввести два числа: целый числитель и целый ненулевой знаменатель. В ответе мы получим последовательность целых чисел, в которой первое число может быть отрицательным (отвечает за целую часть дроби) и отделяется точкой с запятой, а все последующие строго положительные и отделяются запятой. Как упоминалось выше исключительная ситуация (помимо некорректного ввода) возникает только при попытке ввести нулевой знаменатель. Ниже приведены различные вариации вводимых рациональных чисел и соответствующие им цепные последовательности, а также исключительная ситуация.

Дробь	Результат
<input type="text" value="-3"/>	<input type="text" value="[-1; 1, 6, 3, ]"/>
<input type="text" value="22"/>	
<input type="button" value="Вычислить"/>	

Дробь	Результат
<input type="text" value="222"/>	<input type="text" value="[ 10; 1, 1, 3, ]"/>
<input type="text" value="21"/>	
<input type="button" value="Вычислить"/>	

Дробь	Результат
<input type="text" value="55"/>	<input type="text" value="[ 0; 22, 2, 1, 1, 1, 1, 2, ]"/>
<input type="text" value="1231"/>	
<input type="button" value="Вычислить"/>	

Дробь	Результат
<input type="text" value="1296"/> <input type="text" value="0"/>	<div>Вычислить</div>

Ошибка

✖ Введите ненулевой знаменатель

ОК

## Функция Эйлера

Для корректного вычисления функции Эйлера нам нужно ввести целое положительное число, иначе выбрасывается исключение. Примеры приведены ниже.

Точка	Значение функции Эйлера
<input type="text" value="0"/>	<div>Вычислить</div>

Ошибка

✖ Введите целое положительное число

ОК

Точка	Значение функции Эйлера
<input type="text" value="-44"/>	<div>Вычислить</div>

Ошибка

✖ Слишком маленькое или слишком большое число

ОК

Точка	Значение функции Эйлера
<input type="text" value="444"/>	<input type="text" value="144"/> <div>Вычислить</div>

## НОД и НОК

При вычислении наибольшего общего делителя и наименьшего общего кратного нам нужно ввести два целых числа и нажать на соответствующую функцию. Исключительная ситуация возникает если одно из вводимых чисел равно нулю. Примеры корректного ввода и исключений приведены ниже.

Первое число	Второе число	НОД
1	0	

Ошибка

✖ Некорректные аргументы

ОК

Первое число	Второе число	НОД
0	2	

Ошибка

✖ Некорректные аргументы

ОК

Первое число	Второе число	НОД
-7	-5	35

Первое число	Второе число	НОД
296	32	8

Первое число	Второе число	НОД
77	131	1


## Диофантово уравнение

Для нахождения корней диофантова уравнения нам нужно ввести соответствующие коэффициенты  $a$ ,  $b$ ,  $c$  уравнения. Коэффициенты могут быть целыми ненулевыми числами. Ответ является суммой частного и общего решений, записанного в строку типа `string`. Примеры корректного ввода и исключений приведены ниже.

Введённое уравнение	Результат
<input type="text" value="1"/> x + <input type="text" value="2"/> y = <input type="text" value="3"/>	$x = -3 + 2k$ $y = 3 + -1k$

Введённое уравнение	Результат
<input type="text" value="-3"/> x + <input type="text" value="3"/> y = <input type="text" value="-26"/>	Решений нет

Введённое уравнение	Результат
<input type="text" value="-9"/> x + <input type="text" value="4"/> y = <input type="text" value="99"/>	$x = 99 + 4k$ $y = 198 + 9k$

Введённое уравнение	Результат
<input type="text" value="0"/> x + <input type="text" value="0"/> y = <input type="text" value="1"/>	<div> <div>Ошибка</div> <div>  Введите ненулевые целые числа </div> <div>OK</div> </div>

## Проблемы в реализации проекта

*Дробин Роман:* При разработке блока алгебры незначительную проблему составлял алгоритм приведения к ступенчатому виду (GaussStepwise), поскольку от него зависело множество остальных методов. Также при заполнении матриц, я сначала использовал компоненты TextBox, и для каждого размера матрицы создавал панель с этими компонентами и открывал нужную, что оказалось весьма неэффективно. Поэтому я решил совместно с программистом блока “Теория графов” использовать более удобный компонент DataGridView. Более весомые проблемы возникли при компоновке разделов помимо алгебры. Например, при разработке интерфейса на компьютере и при тестировании проекта на каком-либо ноутбуке возникала проблема того, что некоторые компоненты на ноутбуке не влезали на экран, и нарушалась логика программы. Далее, при вставке форм, которые были написаны моими коллегами, возникали трудности: что-то не отображалось, что то отображалось не там, но в итоге все получилось. Также трудность возникла при загрузке теории в виде pdf-файла, так как для открытия конкретного файла нужно было задавать абсолютный путь, но так как при переносе проекта на другое устройство, абсолютный путь отличался, мы решили использовать относительный путь при помощи пространства имен System.IO.Directory и метода “GetCurrentDirectory”.

*Козлов Никита:* Производная. Камень преткновения и спонсор моих бессонных ночей. Основная сложность была в том, чтобы написать такой парсер входной строки, который правильно и чётко бы определял операции, обрабатывал ошибочные входные данные, а также не путал скобочки приоритетности операций со скобочками функции. Кроме того, отдельно много внимания я удостоил операции возведения в степень, которая, в отличие от своих собратьев, считается обратным ходом. В результате оказалось, что это не я дурак, а лыжи не едут, ведь “ $x^{2^2}$ ” любая машина распознает как “ $x^4$ ” и это совершенно верно с точки зрения ОПЗ. (Чтобы машина поняла, что требовалась степень в степени, нужно обязательно расставить скобки “ $(x^2)^2$ ”)

Для нахождения значения функции в точке был написан совершенно другой парсер, который фактически является базовым калькулятором с некоторыми дополнительными возможностями. Чтобы функцию превратить в выражение я использовал самовыдуманный кост... метод, именуемый StringToDouble, который преобразовывал любые вхождения переменной “x” в число.

*Долгих Данила:* Основной проблемой в реализации теории графов было отрисовка и нахождение пересечения ребер с вершинами, что решалось учетом смещения центра (в чем и возникала проблема) вершин при отрисовке и решение пары квадратных уравнений. Также можно было бы удачнее на уровне классов реализовать разделение графа на взвешенный и простой. В комбинаторике же проблема возникала при выходе из диапазона типа `double`, из-за чего использовался класс `BigInteger` из пространства `System.Numerics`, не имеющий ограничений в диапазоне значений.

*Степанов Никита:* При разработке функционала геометрии глобальных проблем не было, трудности вызвала только реализация отдельных методов, например, расстояние между прямыми, где необходимо различать параллельные и скрещивающиеся. Сложнее было реализовывать методы блока “Алгебра полиномов”, т.к. в некоторых ситуациях вылетали ошибки, для обработки которых GUI был не готов. Пришлось добавить исключительные ситуации в самих методах, написанных ранее. Также возникали проблемы с выводом результатов. В конце возникла проблема с вводом некорректных значений, но она легко исправилась добавлением обработчика исключений.

## **Заключение**

Если Вы это читаете, значит наше время уже не зря было потрачено. А кроме шуток, как результат, можно выделить готовое приложение, способное вычислять, решать, визуализировать. Наше приложение, конечно же, не заменяет полноценные математические пособия, т.е. учебники, справочники и т.д., но будет очень полезно для проверки собственных вычислений, а также для решения неких смешанных задач из разных математических дисциплин.

Хочется отметить, что написанные нами функции хоть и явно разграничены разделами, на самом деле комбинируются и взаимно дополняют друг друга. Поэтому такое взаимодействие и непосредственное вовлечение весьма прокачали наши навыки программирования. Один из наших коллег сказал, что при помощи библиотеки `SymPy` можно было бы реализовать производную функции в две строчки, а не в 1500, что мы имеем сейчас, а так же мы могли бы очень упростить реализацию блока “Алгебра многочленов” и некоторые другие, используя библиотеку `SymPy`. Конечно, мы не хотели подчистую красть чьи-то решения, реализации функций, поэтому львиная доля наших

алгоритмов написана собственноручно. Потому был выбран C#, в котором готовых алгоритмов во встроенных библиотеках согласно нашим задачам попросту не было, но было всё, чтобы их написать, а не Python. Изначально C# был для нас чистым листом, некой солянкой из Java и C++, но благодаря тому, что он очень приятный, а главное очень понятный для тех кто раньше с ним не работал, мы очень быстро привыкли к его особенностям и фишкам. Обилие .NET и встроенных библиотек делает этот язык поистине волшебным.

Мы чётко и реалистично поставили дедлайны каждого этапа проекта, однако, как часто и бывает, немного “забили” на проект, так как времени для поиска новых идей, методов особо не было (иногда свою роль в этом играло отсутствие мотивации ввиду неисправленных багов на протяжении недель, а то и месяцев). А плотно заниматься проектом мы начали как раз последний месяц и совсем не учли необходимость оставить время на полноценное тестирование, что вышло нам боком даже при финальной сдаче. Поэтому хотелось бы посоветовать нам на будущее, все-таки стараться делать все равномерно и не оставлять снежный ком напоследок.

На самом деле, все были увлечены проектом, своими собственными задачами, потому на выходе мы имеем самодовольные морды и неплохое, на наш скромный взгляд, приложение. Делать что-то своими руками, то, что тебе правда нравится (математику мы очень любим), а особенно попробовать запрограммировать алгоритмы нахождения чего угодно, весьма мотивирует. Это был увлекательный, интересный а, главное, полезный опыт работы в команде.

Мы все научились чему-то новому по завершении этого проекта: познакомились с новым языком, улучшили навыки разработки оконных приложений, изучили нюансы и особенности работы с формами, добавлением данных, и создания собственного дизайна и стиля приложения. Мы также улучшили свои коммуникативные навыки, навыки работы в команде, и поняли, что значит нести ответственность за свой код перед своими единомышленниками.