

**Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Факультет безопасности информационных технологий

Дисциплина:

«Алгоритмы и структуры данных»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №0

«Решение кубических уравнений»

Выполнила:

Дынина Е.А., студент группы N3249

(подпись)

Проверил:

Ерофеев С.А.

(отметка о выполнении)

(подпись)

Санкт-Петербург

2025 г.

СОДЕРЖАНИЕ

Содержание	2
Постановка задачи	3
Техническая задача.....	4
Входные данные	4
Пролеточные данные	4
Выходные данные	5
Используемые функции.....	6
Блок-схема.....	8
Код	10
Тестирование.....	10
Заключение.....	16

ПОСТАНОВКА ЗАДАЧИ

Цель работы — написать программу для решения кубического уравнения вида $ax^3 + bx^2 + cx + d = 0$.

Для выполнения цели поставлены следующие задачи:

- Изучить способы решения кубических уравнений;
- Создать блок-схему алгоритма решения с учетом всех возможных ситуаций;
- Написать программу на языке Python с интерфейсом для пользователей.

ТЕХНИЧЕСКАЯ ЗАДАЧА

Входные данные

На вход в программу через консольное приложение пользователь подает коэффициенты кубического уравнения.

Таблица 1 – Входные данные

Название переменной	Тип в Python	Диапазон типа	Значение
a	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Коэффициент при x^3
b	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Коэффициент при x^2
c	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Коэффициент при x^1
d	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Коэффициент при x^0

Промежуточные данные

В ходе работы программы задействованы промежуточные переменные, указанные в таблице 2.

Таблица 2 – Промежуточные данные

Название переменной	Тип в Python	Диапазон типа	Значение
D	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Дискриминант квадратного уравнения $D = c^2 - 4bd$
r	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Замена $r = \frac{b}{a}$ для приведения уравнения $ax^3 + bx^2 + cx + d = 0$ к каноническому виду $x^3 + rx^2 + sx + t = 0$
s	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Замена $s = \frac{c}{a}$ для приведения уравнения $ax^3 + bx^2 + cx + d = 0$ к каноническому виду $x^3 + rx^2 + sx + t = 0$
t	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Замена $t = \frac{d}{a}$ для приведения уравнения $ax^3 + bx^2 + cx + d = 0$ к каноническому виду $x^3 + rx^2 + sx + t = 0$
p	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Замена $p = \frac{3s-r^2}{3}$ для приведения канонического вида уравнения

			$x^3 + rx^2 + sx + t = 0$ к виду $y^3 + py + q = 0$
q	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Замена $q = \frac{2r^2}{27} - \frac{rs}{3} + t$ для приведения канонического вида уравнения $x^3 + rx^2 + sx + t = 0$ к виду $y^3 + py + q = 0$
W	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Дискриминант кубического уравнения $W = \left(\frac{p}{3}\right)^3 + \left(\frac{q}{2}\right)^2$
u	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Вспомогательная переменная $u = \sqrt{-\frac{q}{2} + \sqrt{D}}$ для вычисления корней уравнения при $W \geq 0$
v	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Вспомогательная переменная $v = \sqrt{-\frac{q}{2} - \sqrt{D}}$ для вычисления корней уравнения при $W \geq 0$
s	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Вспомогательная переменная $s = \sqrt{\frac{p^3}{27}}$ для вычисления корней уравнения при $W < 0$
phi	float	$[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$	Вспомогательная переменная $phi = \arccos\left(-\frac{q}{2s}\right)$ для вычисления корней уравнения при $W < 0$

Выходные данные

При $a = b = c = 0$ программа должна выводить сообщение «Решение уравнения — любое число» при $d = 0$ или «Нет решений уравнения» при $d \neq 0$.

При $a = b = 0, c \neq 0$ программа выводит решение уравнения $x = -\frac{d}{c}$ типа float (диапазон $[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$).

При $a = 0, b \neq 0$ уравнение решается с помощью дискриминанта квадратного уравнения. Если $D > 0$, на выводе мы получаем два решения уравнения типа float (диапазон $[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$):

$$x_{1,2} = \frac{-c \pm \sqrt{D}}{2b}.$$

Если $D < 0$, на выводе мы получаем два решения уравнения типа `complex` (вещественная и мнимая часть типа `float` с диапазоном $[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$):

$$x_{1,2} = \frac{-c \pm \sqrt{|D|}i}{2b}.$$

Если $D = 0$, на выводе мы получаем одно двукратное решение уравнения типа `float` (диапазон $[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$):

$$x_1 = x_2 = \frac{-c}{2b}.$$

При $a \neq 0$ кубическое уравнение решается методом Кардано. После замен, приведенных в пункте «Промежуточные данные», рассчитывается дискриминант W , указанный там же.

Если $W \geq 0$, программа находит корни уравнения как

$$x_1 = u + v - \frac{r}{3},$$

$$x_{2,3} = h \pm gi.$$

При $g = 0$ на выход подаются три вещественных числа типа `float` (диапазон $[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$), иначе одно вещественное типа `float` (диапазон $[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$) и два комплексных типа `complex` (вещественная и мнимая часть типа `float` с диапазоном $[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$).

Если $W < 0$, программа находит промежуточные переменные s, phi , далее считает и выводит три вещественных решения уравнения типа `float` (диапазон $[-1,7 \cdot 10^{308}, 1,7 \cdot 10^{308}]$):

$$x_1 = 2\sqrt[3]{s} \cdot \cos\left(\frac{phi}{3}\right) - \frac{r}{3}$$

$$x_2 = 2\sqrt[3]{s} \cdot \cos\left(\frac{phi}{3} + \frac{2\pi}{3}\right) - \frac{r}{3}$$

$$x_3 = 2\sqrt[3]{s} \cdot \cos\left(\frac{phi}{3} + \frac{4\pi}{3}\right) - \frac{r}{3}$$

Используемые функции

- **math.cbrt()** — возведение числа в куб;
- **math.sqrt()** — возведение числа в квадрат;

- **math.cos()** — возвращение косинуса от числа;
- **complex()** — создание комплексного числа;
- **.imag** — возвращение мнимой части комплексного числа;
- **.real** — возвращение действительной части комплексного числа;
- **round()** — округление числа до определенного количества цифр после запятой.

БЛОК-СХЕМА

Рассмотрим блок-схему придуманного алгоритма, где рассмотрены все возможные комбинации коэффициентов кубического уравнения и для каждой найдены свои значения корней уравнения.

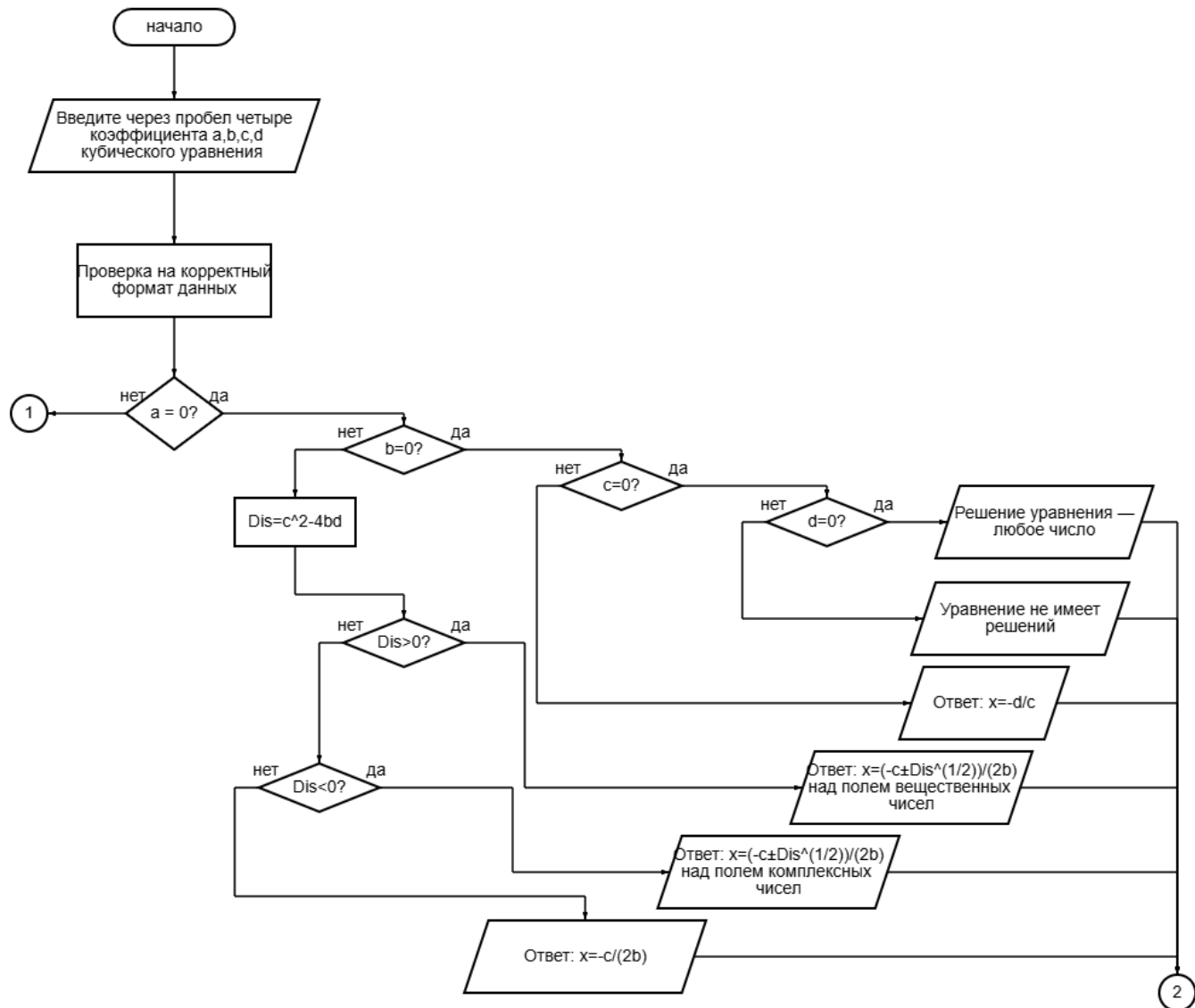


Рисунок 1 – Первая часть блок-схемы

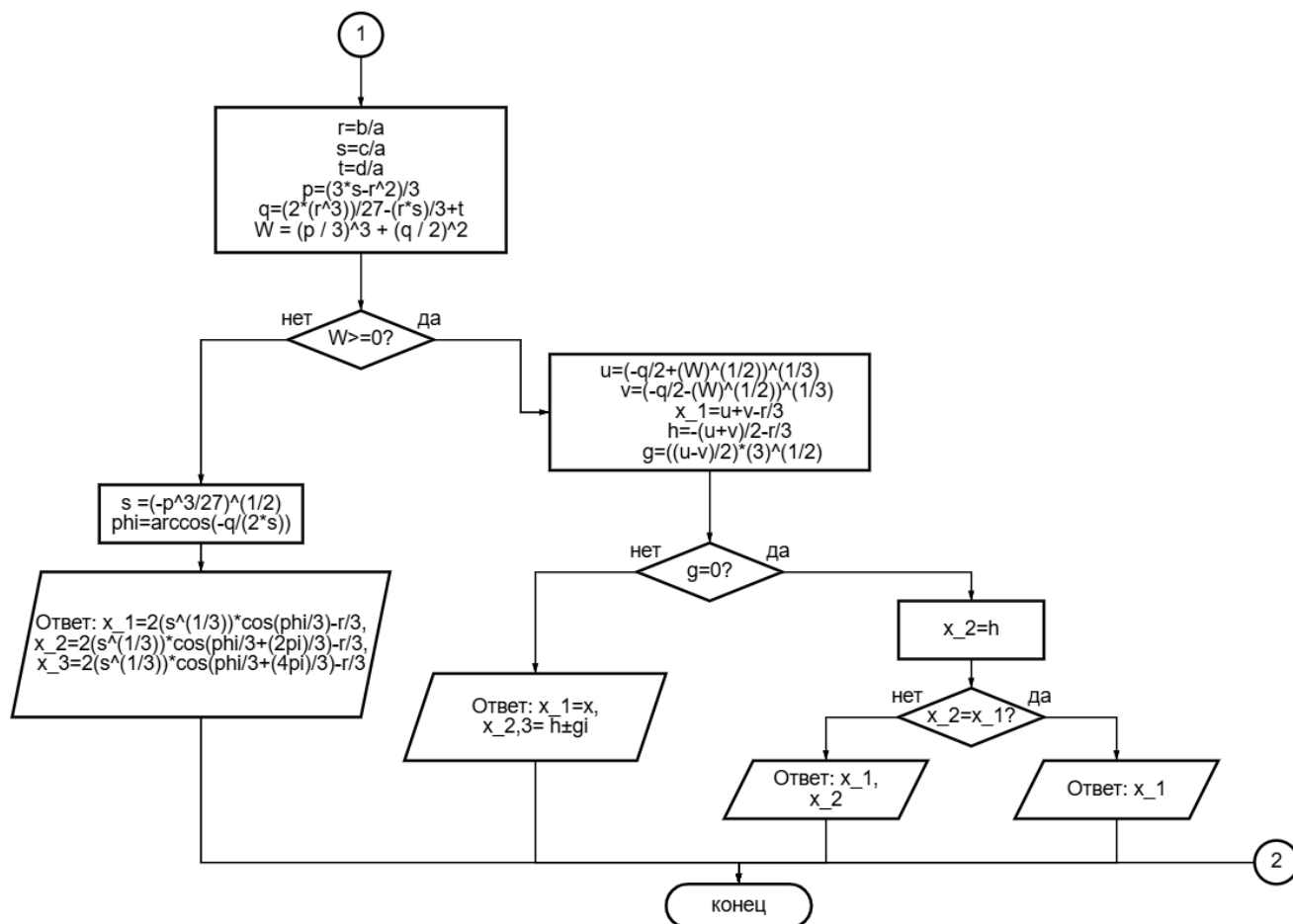


Рисунок 2 – Вторая часть блок-схемы

КОД

Ниже приведен код, написанный для решения кубического уравнения по указанной выше блок-схеме. Программа реализована на языке Python.

```
import sys
import math

#Ввод вещественных коэффициентов a, b, c, d
try:
    a, b, c, d = map(float, input("Введите через пробел 4 числовых коэффици-
ента кубического уравнения\n").split())
#Проверка на корректность входных данных
except:
    print("Ошибка: неверный формат данных")
    sys.exit()

#Решения случаев при нулевом коэффициенте при x^3
if a==0:
    if b==0:
        if c==0:
            if d==0:
                print("Решение уравнения – любое число")
            else:
                print("Нет решений уравнения")
        else:
            print(f"Ответ: x = {0-d / c}")
    #Решение обыкновенного квадратного уравнения через дискриминант
    else:
        dis=c**2-4*b*d #Дискриминант квадратного уравнения
        #Вывод двух вещественных решений уравнения при положительном дискри-
минанте
        if dis>0:
            print(f"Ответ:\nx_1 = {(0-c-math.sqrt(dis))/(2*b)}, \nx_2 = {(0-
c+math.sqrt(dis))/(2*b)}")
            #Вывод двух комплексных решений уравнения при отрицательном дискрими-
нанте
            elif dis<0:
                x_1=complex(0-c, -math.sqrt(-dis)) / (2 * b)
                x_2=complex(0-c, math.sqrt(-dis)) / (2 * b)
                print(f"Ответ:\nx_1 = {x_1 if x_1.imag!=0 else x_1.real}, \nx_2 =
{x_2 if x_2.imag!=0 else x_2.real}")
                #Вывод одного вещественного решения уравнения при нулевом дискрими-
нанте
            else:
                print(f"Ответ: x_1 = x_2 = {0-c / 2*b}")
#Метод Кардано при a!=0
else:
    #Инициализация и расчет вспомогательных переменных для удобной замены
    r=b/a
    s=c/a
    t=d/a
    p=(3*s-r*r)/3
    q=(2*(r**3))/27-r*s/3+t
    #Вычисление дискриминанта
    W = (p / 3)**3 + (q / 2)**2
    #Расчет и вывод вещественного и двух комплексных решений уравнения
    if round(W,8)>=0:
        #расчет вспомогательных и искомых переменных
        u=math.cbrt(-q/2+math.sqrt(W))
        v=math.cbrt(-q/2-math.sqrt(W))
```

```

x_1=round((u+v-r/3),8)
h=round(-(u+v)/2-r/3,8)
g=round(((u-v)/2)*math.sqrt(3),8)
#Вывод при нулевой комплексной части одного из решений уравнения
if g==0:
    x_2=h #Нулевая комплексная часть обращает решение уравнения в ве-
щественное число
    #Вывод трехкратного вещественного решения уравнения
    if x_2==x_1:
        print(f"Ответ:\nx_1 = x_2 = x_3 = {x_1}")
    # Вывод однократного и двукратного вещественного решения уравне-
ния
    else:
        print(f"Ответ:\nx_1 = {x_1},\nx_2 = x_3 = {x_2}")
else: #Расчет двух комплексных решений уравнения
    x_2=complex(h,g)
    x_3=complex(h,-g)
    #Вывод вещественного решения уравнения и двух комплексных
    print(f"Ответ:\nx_1 = {x_1},\nx_2 = {x_2}\nx_3 = {x_3}")

# Расчет и вывод трех вещественных решений уравнения
else:
    s = math.sqrt(-p**3/27)
    phi=math.acos(-q/(2*s))
    x_1=round(2*math.cbrt(s)*math.cos(phi/3)-b/a/3,8)
    x_2=round(2*math.cbrt(s)*math.cos(phi/3+2*math.pi/3)-b/(a*3),8)
    x_3=round(2*math.cbrt(s)*math.cos(phi/3+4*math.pi/3)-b/(a*3),8)
    print(f"Ответ:\nx_1 = {x_1},\nx_2 = {x_2},\nx_3 = {x_3}")

```

ТЕСТИРОВАНИЕ

Проведено тестирование программы с помощью различных входных данных. Был использован интерпретатор Python 3.11.0 и среда разработки PyCharm 2022.3.2

1. Все коэффициенты нулевые

Введите через пробел 4 числовых коэффициента кубического уравнения

0 0 0 0

Решение уравнения – любое число

Рисунок 3 – Тест №1

2. Только свободный коэффициент ненулевой

Введите через пробел 4 числовых коэффициента кубического уравнения

0 0 0 1

Нет решений уравнения

Рисунок 4 – Тест №2

3. Линейное уравнение

Введите через пробел 4 числовых коэффициента кубического уравнения

0 0 2 3

Ответ: $x = -1.5$

Рисунок 5 – Тест №3

4. Квадратное уравнение с нулевым дискриминантом

Введите через пробел 4 числовых коэффициента кубического уравнения

0 1 4 4

Ответ: $x_1 = x_2 = -2.0$

Рисунок 6 – Тест №4

5. Квадратное уравнение с отрицательным дискриминантом

Введите через пробел 4 числовых коэффициента кубического уравнения

0 1 4 5

Ответ:

$x_1 = (-2-1j),$

$x_2 = (-2+1j)$

Рисунок 7 – Тест №5

6. Квадратное уравнение с положительным дискриминантом

Введите через пробел 4 числовых коэффициента кубического уравнения

0 1 4 3

Ответ:

$x_1 = -3.0,$

$x_2 = -1.0$

Рисунок 8 – Тест №6

7. Кубическое уравнение с отрицательным дискриминантом

Введите через пробел 4 числовых коэффициента кубического уравнения

1 -6 11 -6

Ответ:

$x_1 = 3.0,$

$x_2 = 1.0,$

$x_3 = 2.0$

Рисунок 9 – Тест №7

8. Кубическое уравнение с положительным дискриминантом и нулевыми коэффициентами при x^2, x

Введите через пробел 4 числовых коэффициента кубического уравнения

1 0 0 -1

Ответ:

$x_1 = 1.0,$

$x_2 = (-0.5+0.8660254j)$

$x_3 = (-0.5-0.8660254j)$

Рисунок 10 – Тест №8

9. Кубическое уравнение с нулевым дискриминантом

Введите через пробел 4 числовых коэффициента кубического уравнения

1 -1 -1 1

Ответ:

$x_1 = -1.0,$

$x_2 = x_3 = 1.0$

Рисунок 11 – Тест №9

10. Кубическое уравнение с трехкратным вещественным решением

Введите через пробел 4 числовых коэффициента кубического уравнения

1 -3 3 -1

Ответ:

x_1 = x_2 = x_3 = 1.0

Рисунок 12 – Тест №10

11. Кубическое уравнение с положительным дискриминантом и ненулевыми коэффициентами

Введите через пробел 4 числовых коэффициента кубического уравнения

1 -4 6 -4

Ответ:

x_1 = 2.0,

x_2 = (1+1j)

x_3 = (1-1j)

Рисунок 13 – Тест №11

12. Только коэффициент при x^3 ненулевой

Введите через пробел 4 числовых коэффициента кубического уравнения

1 0 0 0

Ответ:

x_1 = x_2 = x_3 = 0.0

Рисунок 14 – Тест №12

13. Уравнение с большими коэффициентами

Введите через пробел 4 числовых коэффициента кубического уравнения

1 1000 1000000 1000000000

Ответ:

x_1 = -1000.0,

x_2 = 1000j

x_3 = -1000j

Рисунок 15 – Тест №13

14. Уравнение с маленькими коэффициентами

Введите через пробел 4 числовых коэффициента кубического уравнения
1 1e-6 1e-12 1e-18

Ответ:

$$x_1 = -1e-06,$$

$$x_2 = 1e-06j$$

$$x_3 = -1e-06j$$

Рисунок 16 – Тест №14

15. Кубическое уравнение с одинаковыми положительными коэффициентами

Введите через пробел 4 числовых коэффициента кубического уравнения
1 1 1 1

Ответ:

$$x_1 = -1.0,$$

$$x_2 = 1j$$

$$x_3 = -1j$$

Рисунок 17 – Тест №15

16. Кубическое уравнение с отрицательными коэффициентами

Введите через пробел 4 числовых коэффициента кубического уравнения
-2 -3 -1 -4

Ответ:

$$x_1 = -1.82600518,$$

$$x_2 = (0.16300259+1.03378784j)$$

$$x_3 = (0.16300259-1.03378784j)$$

Рисунок 18 – Тест №16

17. Некорректные входные данные

Введите через пробел 4 числовых коэффициента кубического уравнения
k

Ошибка: неверный формат данных

Рисунок 19 – Тест №17

ЗАКЛЮЧЕНИЕ

Были проанализированы способы решения кубических уравнений и написана блок-схема алгоритма программы, которая находит корни уравнения через дискриминант и метод Кардано. На языке Python написан код программы с выводом только неповторяющихся корней, учетом вычислительной погрешности языка программирования и интерфейсом для пользователя.

Программа успешно протестирована на различных коэффициентах

- линейных уравнений;
- квадратных уравнений с различными дискриминантами;
- кубических уравнений с тремя вещественными решениями;
- кубических уравнений с одним вещественным и двумя комплексными решениями.

Все задачи выполнены и цель достигнута.