

Содержание

ВВЕДЕНИЕ.....	5
1.ПОСТАНОВКА ЗАДАЧИ.....	6
2.ТЕОРИТИЧЕСКИЕ ОСНОВЫ.....	7
3.БЛОК-СХЕМА.....	8
4.ЛИСТИНГ ПРОГРАММЫ.....	9
5.РЕЗУЛЬТАТ РАБОТЫ ПРОГРАММЫ.....	12
ЗАКЛЮЧЕНИЕ.....	13
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ.....	14

ВВЕДЕНИЕ

Python — высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. В то же время стандартная библиотека включает большой объём полезных функций.

Python поддерживает несколько парадигм программирования, в том числе структурное, объектно-ориентированное, функциональное, императивное и аспектно-ориентированное. Основные архитектурные черты — динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений и удобные высокоуровневые структуры данных. Код в Питоне организовывается в функции и классы, которые могут объединяться в модули (которые в свою очередь могут быть объединены в пакеты).

Эталонной реализацией Python является интерпретатор CPython, поддерживающий большинство активно используемых платформ. Он распространяется под свободной лицензией Python Software Foundation License, позволяющей использовать его без ограничений в любых приложениях, включая проприетарные. Есть реализации интерпретаторов для JVM (с возможностью компиляции), MSIL (с возможностью компиляции), LLVM и других. Проект PyPy предлагает реализацию Питона на самом Питоне, что уменьшает затраты на изменения языка и постановку экспериментов над новыми возможностями.

Python — активно развивающийся язык программирования, новые версии (с добавлением/изменением языковых свойств) выходят примерно раз в два с половиной года. Вследствие этого и некоторых других причин на Python отсутствуют стандарт ANSI, ISO или другие официальные стандарты, их роль выполняет CPython.

1. ПОСТАНОВКА ЗАДАЧИ

Создать процедуру для решения уравнения $ax^2+bx+c=0$ и построения его графика. Входные параметры: уравнение, а, b, с. Выходные параметры будут отображаться графическим интерфейсом в виде графика.

2. ТЕОРИТИЧЕСКИЕ ОСНОВЫ

Квадратное уравнение — алгебраическое уравнение общего вида

$$ax^2+bx+c=0$$

где x — свободная переменная, a , b , c — коэффициенты.

Выражение ax^2+bx+c называют квадратным трёхчленом.

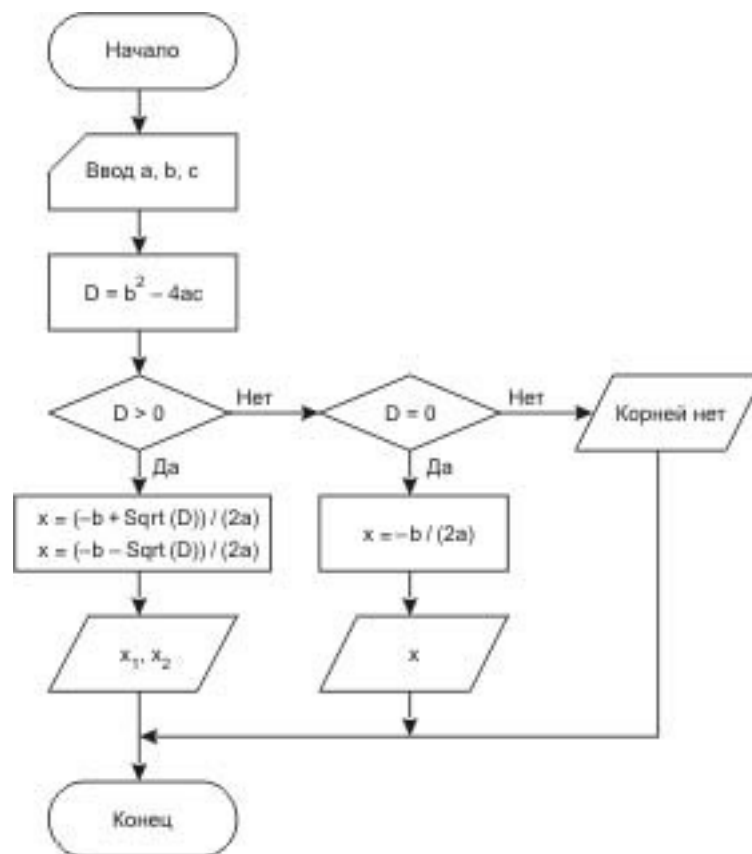
Корень такого уравнения (корень квадратного трёхчлена) — это значение переменной x , обращающее квадратный трёхчлен в ноль, то есть значение, обращающее квадратное уравнение в тождество.

Коэффициенты квадратного уравнения имеют собственные названия: коэффициент a называют первым или старшим, коэффициент b называют вторым или коэффициентом при x , c называется свободным членом этого уравнения.

Полным квадратным уравнением называют такое, все коэффициенты которого отличны от нуля.

Неполным квадратным уравнением называется такое, в котором хотя бы один из коэффициентов кроме старшего (либо второй коэффициент, либо свободный член) равен нулю.

3.БЛОК-СХЕМА



Блок схема 1.Алгоритм работы программы.

4.ЛИСТИНГ ПРОГРАММЫ.

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
import pylab
import matplotlib
from math import *
import numpy as np
import matplotlib.pyplot as plt
# Используем графическую библиотеку Tkinter
import Tkinter
import ttk
# Создаем форму (окно)
tk=Tkinter.Tk()
tk.title("Кур. работа")
# Создаем GUI элементы ввода данных
lbla=Tkinter.Label(tk)
lbla["text"]="A= "
lbla.pack()
# Создаем поле ввода
a=Tkinter.Entry(tk)
a.pack()
lblb=Tkinter.Label(tk)
lblb["text"]="B= "
lblb.pack()
# Создаем поле ввода
b=Tkinter.Entry(tk,width=20,bd=3)
b.pack()
lblc=Tkinter.Label(tk)
lblc["text"]="C= "
lblc.pack()
# Создаем поле ввода
c=Tkinter.Entry(tk,width=20,bd=3)
c.pack()
# Определяем функцию-обработчик события нажатия на
конпку
def solve():
    a1=float(a.get())
    b1=float(b.get())
    c1=float(c.get())
    d=b1**2-4*a1*c1

    if d<0:
# Создание объект Label (Надпись)
    lbl0=Tkinter.Label(tk)
```

```

        lbl0["text"]="Корней нет, D меньше нуля"
        lbl0.pack()
    if d>0:
        x1=(-1*b1+d**0.5)/2*a1
        x2=(-1*b1-d**0.5)/2*a1

# Создание объект Label(Надпись)
        lbl1=Tkinter.Label(tk)
        lbl1["text"]="X1= ", x1
        lbl1.pack()
# Создание объект Label(Надпись)
        lbl2=Tkinter.Label(tk)
        lbl2["text"]="X2= ", x2
        lbl2.pack()
        matplotlib.rcParams["axes.grid"] = True
        xmin=-100
        xmax=100

        x = pylab.arange (xmin, xmax, 1)
        y=a1*(x-x1)*(x-x2)
        pylab.clf()
        plt.plot(x,y,x1,x2)
        plt.show()
    if d==0:
        x=(-1*b1)/(2*a1)
# Создание объект Label(Надпись)
        lbl=Tkinter.Label(tk)
        lbl["text"]="X1=X2= ", x
        lbl.pack()
        matplotlib.rcParams["axes.grid"] = True
        x = pylab.arange (-20.0, 20.1, 0.1)
        y=a1*x**2+b1*x+c1
        pylab.clf()
        plt.plot(x,y)
        plt.show()
# Создаем кнопку График и результат
btn=Tkinter.Button(tk)
btn["text"]="График и результат"
# Привязываем функцию-обработчик к событию нажатия
btn["command"]=solve
btn.pack()

# Создаем кнопку выхода из приложения
button=Tkinter.Button(tk)

```

```
button["text"]="Закрыть"  
button["command"]=tk.quit  
button.pack()
```

```
# Запуск  
tk.mainloop()
```

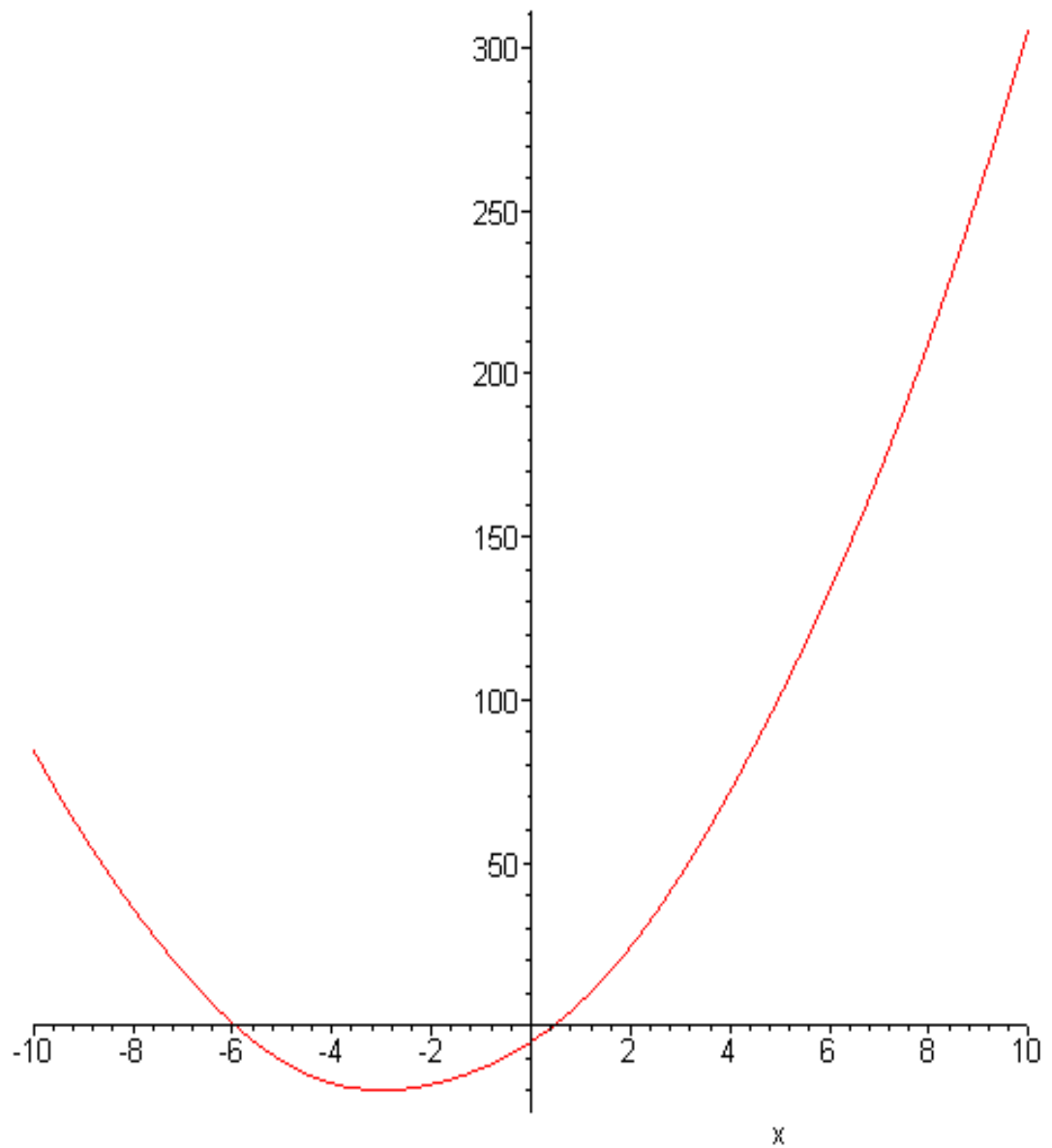


Рис.1. График функции.

5.РЕЗУЛЬТАТ РАБОТЫ ПРОГРАММЫ.

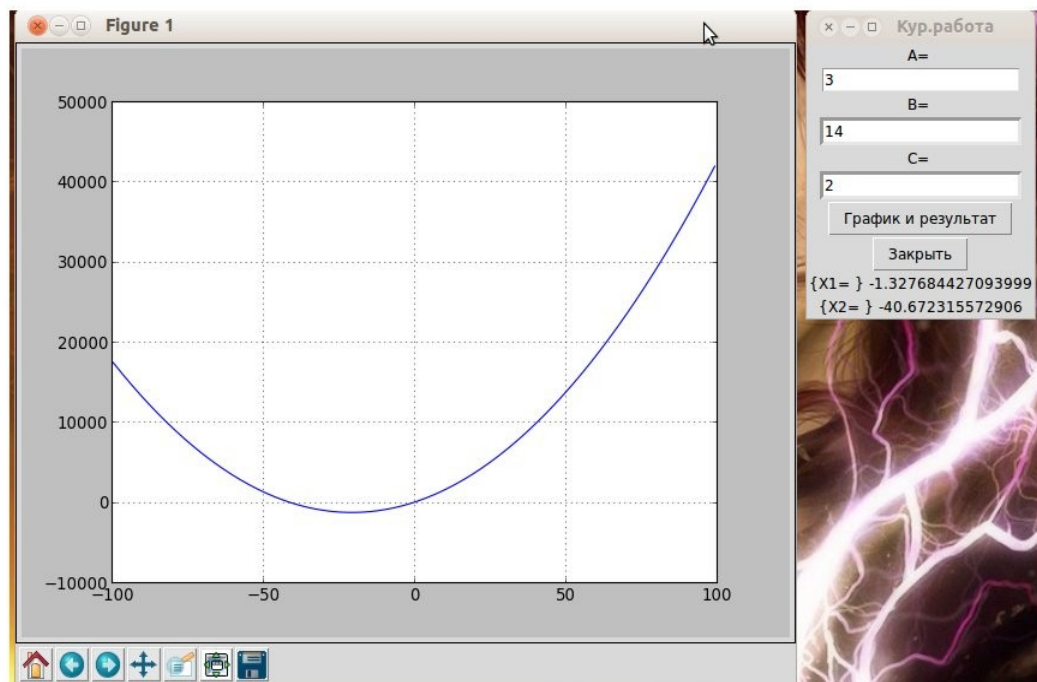


Рис. Результат работы программы.

На терминале представлено рабочее окно выполненной программы. В окнах ввода задаются коэффициенты a, b, c . При введении данных программа рассчитывает заданное уравнение с исходными данными и выводится график решения данного уравнения.

ЗАКЛЮЧЕНИЕ

В ходе данной курсовой работы был изучен метод написания кода программы для решения квадратного уравнения и построения графика при помощи языка программирования Python. Для удобства решения коэффициенты вводились через клавиатуру.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Россум Г., Дрейк Ф.Л.Дж. Язык программирования Python.
 2. Сузи Р.А. Язык программирования Python.
 3. Gift N., Jones J. M. Python for Unix and Linux System Administration.
- Vaingast S. Beginning Python Visualization - Crafting Visual Transformation Scripts.