МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. Н.Э.Баумана

Отчет по лабораторной работе №1

по курсу «Разработка интернет-приложений»

Основы языка Python

Подготовил

Ионов С.А.

ИУ5-52Б

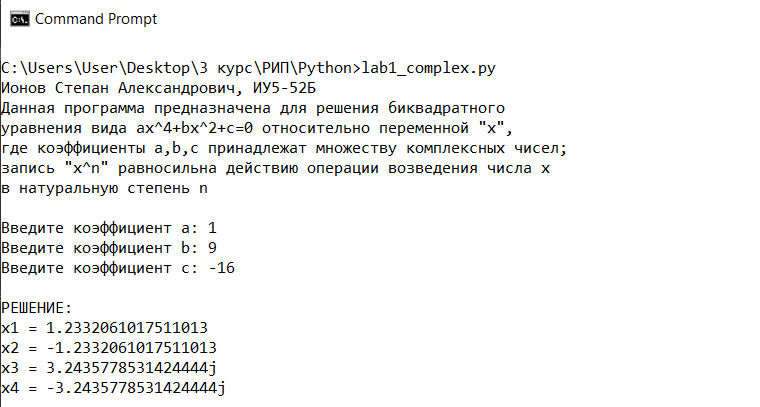
1. Описание задания

Разработать программу для решения биквадратного уравнения.

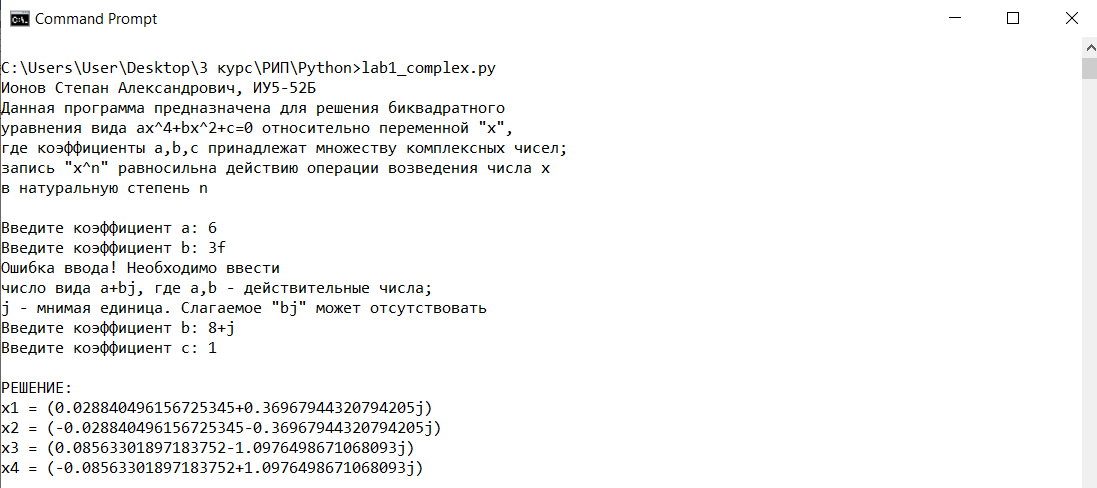
1. Программа должна быть разработана в виде консольного приложения на языке Python.
2. Программа осуществляет ввод с клавиатуры коэффициентов А, В, С, вычисляет дискриминант и корни уравнения (в зависимости от дискриминанта).
3. Если коэффициент А, В, С введен некорректно, то необходимо проигнорировать некорректное значение и ввести коэффициент повторно.
4. Первой строкой программа выводит ФИО разработчика и номер группы.
5. ДОПОЛНИТЕЛЬНОЕ ТРЕБОВАНИЕ. Коэффициенты А, В, С задаются в виде параметров командной строки. Если они не указаны, то вводятся с клавиатуры в соответствии с пунктом 2. Проверка из пункта 3 в этом случае производится для параметров командной строки без повторного ввода с клавиатуры.
6. Текст программы

**from** cmath **import** sqrt  
**import** argparse  
**import** sys  
  
  
**def** create\_parser(): *# получение параметров командной строки* parser = argparse.ArgumentParser()  
 parser.add\_argument(**"a"**, nargs=**"?"**)  
 parser.add\_argument(**"b"**, nargs=**"?"**)  
 parser.add\_argument(**"c"**, nargs=**"?"**)  
 **return** parser.parse\_args()  
  
  
**def** get\_from\_param(val, instructions): *# проверка параметров* **try**:  
 **return** complex(val)  
 **except** ValueError:  
 print(instructions[**'param\_value\_error'**])  
 print(instructions[**'instr\_for\_parameters'**])  
 sys.exit()  
 **except** TypeError:  
 print(instructions[**'param\_type\_error'**])  
 print(instructions[**'instr\_for\_parameters'**])  
 sys.exit()  
  
  
**def** get\_from\_input(input\_request, error\_msg): *# проверка ввода* **while True**:  
 **try**:  
 coefficient = input(input\_request)  
 **return** complex(coefficient)  
 **except** ValueError:  
 print(error\_msg)  
 **except** EOFError:  
 print(error\_msg)  
  
  
**def** main\_solution(root, str1, str2): *# вывод решения при 'a', не равном нулю,* **if** root.real == 0: *# и не нулевом дискриминанте* print(str1, **" = "**, root.imag, **"j"**, sep=**""**)  
 print(str2, **" = "**, -root.imag, **"j"**, sep=**""**)  
 **elif** root.imag == 0:  
 print(str1, **" = "**, root.real, sep=**""**)  
 print(str2, **" = "**, -root.real, sep=**""**)  
 **else**:  
 print(str1, **" = "**, root, sep=**""**)  
 print(str2, **" = "**, -root, sep=**""**)  
  
  
instr = dict(intro=**"""Ионов Степан Александрович, ИУ5-52Б  
Данная программа предназначена для решения биквадратного  
уравнения вида ax^4+bx^2+c=0 относительно переменной "x",  
где коэффициенты a,b,c принадлежат множеству комплексных чисел;  
запись "x^n" равносильна действию операции возведения числа x  
в натуральную степень n\n"""**,  
 request\_for\_first\_coeff=**"Введите коэффициент a: "**,  
 request\_for\_second\_coeff=**"Введите коэффициент b: "**,  
 request\_for\_third\_coeff=**"Введите коэффициент c: "**,  
 param\_value\_error=**"Ошибка типа данных параметров командной строки!"**,  
 param\_type\_error=**"Не хватает параметров командной строки!"**,  
 instr\_for\_parameters=**"Введите коэффициенты a,b,c через пробел"**,  
 input\_value\_error=**"""Ошибка ввода! Необходимо ввести  
число вида a+bj, где a,b - действительные числа;  
j - мнимая единица. Слагаемое "bj" может отсутствовать"""**)   
print(instr[**'intro'**])  
**if** len(sys.argv) > 1: *# коэффициенты как параметры командной строки* namespace = create\_parser()  
 a = get\_from\_param(namespace.a, instr)  
 b = get\_from\_param(namespace.b, instr)  
 c = get\_from\_param(namespace.c, instr)  
**else**: *# коэффициенты как вводимые данные по запросу* a = get\_from\_input(instr[**'request\_for\_first\_coeff'**],  
 instr[**'input\_value\_error'**]) *# старший коэффициент уравнения* b = get\_from\_input(instr[**'request\_for\_second\_coeff'**],  
 instr[**'input\_value\_error'**]) *# средний коэффициент уравнения* c = get\_from\_input(instr[**'request\_for\_third\_coeff'**],  
 instr[**'input\_value\_error'**]) *# свободный член уравнения*print(**"\nРЕШЕНИЕ:"**)  
**if** a == 0: *# анализ решений и выдача ответа* **if** b != 0:  
 **if** c != 0:  
 z = sqrt(-c/b)  
 **if** z.real == 0:  
 print(**"x1 = "**, z.imag, **"j"**, sep=**""**)  
 print(**"x2 = "**, -z.imag, **"j"**, sep=**""**)  
 **else**:  
 print(**"x1 = "**, z.real, sep=**""**)  
 print(**"x2 = "**, -z.real, sep=**""**)  
 **else**:  
 z = 0.0  
 print(**"x = "**, z, sep=**""**)  
 **elif** b == 0 **and** c == 0:  
 print(**"x - любой из множества комплексных чисел"**)  
 **else**:  
 print(**"Решений нет"**)  
**elif** b != 0 **and** c == 0:  
 z = sqrt(-b/a)  
 print(**"x1 = x2 = "**, 0, sep=**""**)  
 **if** z.real == 0:  
 print(**"x3 = "**, z.imag, **"j"**, sep=**""**)  
 print(**"x4 = "**, -z.imag, **"j"**, sep=**""**)  
 **else**:  
 print(**"x3 = "**, z.real, sep=**""**)  
 print(**"x4 = "**, -z.real, sep=**""**)  
**else**:  
 d = b\*\*2 - 4\*a\*c *# дискриминант уравнения относительно второй степени* **if** d == 0:  
 t = sqrt(-b/(2\*a))  
 **if** t == 0:  
 print(**"x = "**, 0, sep=**""**)  
 **else**:  
 **if** t.real == 0:  
 print(**"x1 = x2 = "**, t.imag, **"j"**, sep=**""**)  
 print(**"x3 = x4 = "**, -t.imag, **"j"**, sep=**""**)  
 **else**:  
 print(**"x1 = x2 = "**, t.real, sep=**""**)  
 print(**"x3 = x4 = "**, -t.real, sep=**""**)  
 **else**:  
 t1 = sqrt((-b+sqrt(d))/(2\*a))  
 t2 = sqrt((-b-sqrt(d))/(2\*a))  
 main\_solution(t1, **"x1"**, **"x2"**)  
 main\_solution(t2, **"x3"**, **"x4"**)

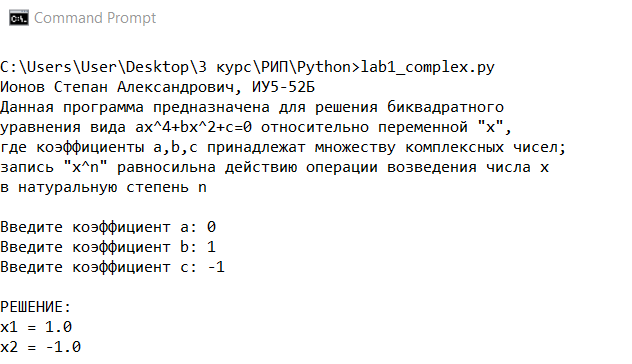
1. Экранные формы с примерами выполнения программы
2. Ввод без параметров для решения уравнения, где a,b,c - ненулевые



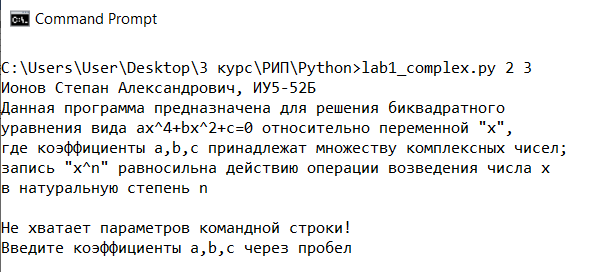
1. Ввод без параметров для решения уравнения, где в b – ошибка



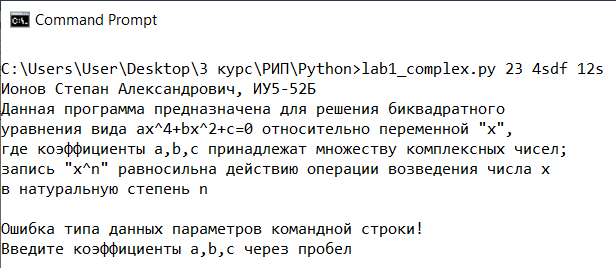
1. Ввод без параметров для решения уравнения, где a=0



1. Ввод с параметрами, где их неполное количество



1. Ввод с параметрами, где они записаны неверно



1. Ввод с параметрами, где c = 0

