**Чепоков Е, Караваев С**

**(bloodiesmail@gmail.com)**

**ДОКУМЕНТАЦИЯ НА ПРОГРАММУ, ИЗВЛЕКАЮЩУЮ КОРЕНЬ**

**(ДЛЯ ПОЛЬЗОВАТЕЛЯ)**

Пермь

2017

**СОДЕРЖАНИЕ**

1. Введение;
2. Наименование, область применения, назначение разработки;
3. Основание для разработки;
4. Технические требования к программному изделию;
5. Стадии и этапы разработки, тесты и ошибки;
6. Персонал и оплата труда;
7. Связь с нами.

# 1. ВВЕДЕНИЕ

Мы хотим подготовить вас морально к этому программному продукту, надеюсь, он хоть как-то оправдает ваши ожидания.

Программа разработана на основе кода, данного нам на лекции.

Рассматриваемый программный продукт разработан на основе выданного Заказчиком технического задания, которое также приведено ниже с незначительными изменениями, и состоит из программы извлечения корня.

# 2. НАИМЕНОВАНИЕ, ОБЛАСТЬ ПРИМЕНЕНИЯ, НАЗНАЧЕНИЕ РАЗРАБОТКИ

Наша программа «Извлекатель» применяется школьниками и студентами для извлечения корня в сфере образования, так же может быть применима ленивым пользователям, которые уже закончили обучение, но жизненная ситуация заставила их извлечь корень.

# 3. ОСНОВАНИЕ ДЛЯ РАЗРАБОТКИ

Основанием для разработки стало домашнее задание, выданное на первом курсе НИУ ВШЭ.

# 4. ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ К ПРОГРАММНОМУ ИЗДЕЛИЮ

1. Функциональность: корни не только арифметические, из нуля, комплексные, длинные числа, заданная точность, аналитические (Часть требований к нашему сожалению не выполнены в связи с незнанием).
2. Ошибкоустойчивость
3. GUI (с учетом управления точностью и полиязычности)
4. Полиязычность
5. Кроссплатформенность

# 5. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ, ТЕСТЫ И ОШИБКИ

Наши алгоритмы будут использоваться для программы, реализованной в среде разработки PyCharm на языке среднего уровня Python.

Код программы:

1. from tkinter import \*
2. import tkinter
3. import tkinter.ttk
4. import math
5. def create\_widgets\_in\_first\_frame():
6. # Create the label for the frame
7. first\_window\_label = tkinter.ttk.Label(first\_frame,

text='Choose Language')

1. first\_window\_label.grid(column=2,

row=0,

pady=10,

padx=10,

sticky=(tkinter.N))

1. # Create the button for the frame
2. create\_widgets\_in\_first\_frame.add\_img = tkinter.PhotoImage(file="1.png")
3. create\_widgets\_in\_second\_frame.add\_img = tkinter.PhotoImage(file="2.png")
4. create\_widgets\_in\_third\_frame.add\_img = tkinter.PhotoImage(file="3.png")
5. create\_widgets\_in\_fourth\_frame.add\_img = tkinter.PhotoImage(file="4.png")
6. first\_window\_next\_button = tkinter.Button(first\_frame,

text="Русский",

1. image=create\_widgets\_in\_first\_frame.add\_img,
2. command=call\_second\_frame\_on\_top)
3. first\_window\_next\_button.grid(column=1,

row=1,

pady=10,

padx=10)

1. first\_window\_next\_button = tkinter.Button(first\_frame,

text="Английский",

1. image=create\_widgets\_in\_second\_frame.add\_img,
2. command=call\_third\_frame\_on\_top)
3. first\_window\_next\_button.grid(column=1,

row=2,

pady=10,

padx=10)

1. first\_window\_next\_button = tkinter.Button(first\_frame,

text="Китайский",

1. image=create\_widgets\_in\_third\_frame.add\_img,
2. command=call\_fourth\_frame\_on\_top)
3. first\_window\_next\_button.grid(column=3,

row=1,

pady=10,

padx=10)

1. first\_window\_next\_button = tkinter.Button(first\_frame,
2. text="Французский",
3. image=create\_widgets\_in\_fourth\_frame.add\_img,
4. command=call\_fifth\_frame\_on\_top)
5. first\_window\_next\_button.grid(column=3,

row=2,

pady=10,

padx=10)

1. first\_window\_quit\_button = tkinter.Button(first\_frame,

text="Exit",

1. command=quit\_program)
2. first\_window\_quit\_button.grid(column=4,

row=3,

pady=10,

padx=10)

1. def create\_widgets\_in\_second\_frame():
2. # Create the label for the frame
3. second\_window\_label = tkinter.ttk.Label(second\_frame,

text='Введите число:')

1. second\_window\_label.grid(column=1,

row=0,

pady=10,

padx=10,

sticky=(tkinter.N))

1. message = StringVar()
2. entry1 = tkinter.Entry(second\_frame, text='', textvariable=message, width=50,)
3. entry1.grid(column=2,

row=0,

pady=10,

padx=10,

sticky=(tkinter.N))

1. second\_window\_label = tkinter.ttk.Label(second\_frame,

text='Точность:')

1. second\_window\_label.grid(column=1,

row=1,

pady=10,

padx=10,

sticky=(tkinter.N))

1. many = StringVar()
2. message\_entry = Entry(second\_frame, text='', textvariable=many, width=20, )
3. message\_entry.grid(column=2,

row=1,

pady=10,

padx=10,

sticky=(tkinter.N))

1. def func1():
   * 1. try:
     2. x = float(entry1.get())
     3. a = int(message\_entry.get())
     4. second\_window\_label1.config(
        + 1. text=("{0:." + str(a) + "f}").format(

math.sqrt(-x) if x < 0 else math.sqrt(x))

* + - * 1. + ("i" if x < 0 else ""))
    1. except ValueError:
    2. second\_window\_label1.config(text="Ошибка введите цифры")

1. second\_window\_label1 = tkinter.Label(second\_frame,

text="Ответ:")

1. second\_window\_label1.grid(column=2,

row=3,

pady=10,

padx=10,

sticky=(tkinter.N))

1. second\_window\_label = tkinter.ttk.Label(second\_frame,

text="Ответ:")

1. second\_window\_label.grid(column=1,

row=3,

pady=10,

padx=10,

sticky=(tkinter.N))

1. # Create the button for the frame
2. second\_window\_enter\_button = tkinter.Button(second\_frame, text='Решить', command=func1)
3. second\_window\_enter\_button.grid(column=2,

row=2,

pady=10,

padx=10)

1. second\_window\_back\_button = tkinter.Button(second\_frame,

text="Назад",

1. command=call\_first\_frame\_on\_top)
2. second\_window\_back\_button.grid(column=0,

row=5,

pady=10,

padx=10)

1. second\_window\_next\_button = tkinter.Button(second\_frame,

text="Выход",

1. command=quit\_program)
2. second\_window\_next\_button.grid(column=5,

row=5,

pady=10,

padx=10)

1. def create\_widgets\_in\_third\_frame():
2. # Create the label for the frame
3. third\_window\_label = tkinter.ttk.Label(third\_frame,

text='输入一个数字:')

1. third\_window\_label.grid(column=1,

row=0,

pady=10,

padx=10,

sticky=(tkinter.N))

1. message = StringVar()
2. entry1 = tkinter.Entry(third\_frame, text='', textvariable=message, width=50, )
3. entry1.grid(column=2,
   * + - 1. row=0,
         2. pady=10,
         3. padx=10,
         4. sticky=(tkinter.N))
4. third\_window\_label = tkinter.ttk.Label(third\_frame,

text='准确度:')

1. third\_window\_label.grid(column=1,

row=1,

pady=10,

padx=10,

sticky=(tkinter.N))

1. many = StringVar()
2. message\_entry = Entry(third\_frame, text='', textvariable=many, width=20, )
3. message\_entry.grid(column=2,

row=1,

pady=10,

padx=10,

sticky=(tkinter.N))

1. def func1():
   * 1. try:
     2. x = float(entry1.get())
     3. a = int(message\_entry.get())
     4. third\_window\_label1.config(
        + 1. text=("{0:." + str(a) + "f}").format(

math.sqrt(-x) if x < 0 else math.sqrt(x))

* + - * 1. + ("i" if x < 0 else ""))
    1. except ValueError:
    2. third\_window\_label1.config(text="错误输入数字")

1. third\_window\_label1 = tkinter.Label(third\_frame,

text="回答:")

1. third\_window\_label1.grid(column=2,

row=3,

pady=10,

padx=10,

sticky=(tkinter.N))

1. third\_window\_label = tkinter.ttk.Label(third\_frame,

text="回答:")

1. third\_window\_label.grid(column=1,

row=3,

pady=10,

padx=10,

sticky=(tkinter.N))

1. # Create the button for the frame
2. third\_window\_enter\_button = tkinter.Button(third\_frame, text='解决', command=func1)
3. third\_window\_enter\_button.grid(column=2,

row=2,

pady=10,

padx=10)

1. # Create the button for the frame
2. third\_window\_back\_button = tkinter.Button(third\_frame,

text="向后",

1. command=call\_first\_frame\_on\_top)
2. third\_window\_back\_button.grid(column=0,

row=5,

pady=10,

padx=10,

sticky=(tkinter.N))

1. third\_window\_quit\_button = tkinter.Button(third\_frame,

text="输出",

command = quit\_program)

1. third\_window\_quit\_button.grid(column=5,

row=5,

pady=10,

padx=10,

sticky=(tkinter.N))

1. def create\_widgets\_in\_fourth\_frame():
2. # Create the label for the frame
3. fourth\_window\_label = tkinter.ttk.Label(fourth\_frame,

text='Enter a number:')

1. fourth\_window\_label.grid(column=1,

row=0,

pady=10,

padx=10,

sticky=(tkinter.N))

1. message = StringVar()
2. entry1 = tkinter.Entry(fourth\_frame, text='', textvariable=message, width=50, )
3. entry1.grid(column=2,
   * + - 1. row=0,
         2. pady=10,
         3. padx=10,
         4. sticky=(tkinter.N))
4. fourth\_window\_label = tkinter.ttk.Label(fourth\_frame,

text='Accuracy:')

1. fourth\_window\_label.grid(column=1,

row=1,

pady=10,

padx=10,

sticky=(tkinter.N))

1. many = StringVar()
2. message\_entry = Entry(fourth\_frame, text='', textvariable=many, width=20, )
3. message\_entry.grid(column=2,

row=1,

pady=10,

padx=10,

sticky=(tkinter.N))

1. def func1():
   * 1. try:
     2. x = float(entry1.get())
     3. a = int(message\_entry.get())
     4. fourth\_window\_label1.config(
        + 1. text=("{0:." + str(a) + "f}").format(

math.sqrt(-x) if x < 0 else math.sqrt(x))

* + - * 1. + ("i" if x < 0 else ""))
    1. except ValueError:
    2. fourth\_window\_label1.config(text="Error enter numbers")

1. fourth\_window\_label1 = tkinter.Label(fourth\_frame,

text="Answer:")

1. fourth\_window\_label1.grid(column=2,

row=3,

pady=10,

padx=10,

sticky=(tkinter.N))

1. fourth\_window\_label = tkinter.ttk.Label(fourth\_frame,

text="Answer:")

1. fourth\_window\_label.grid(column=1,

row=3,

pady=10,

padx=10,

sticky=(tkinter.N))

1. # Create the button for the frame
2. fourth\_window\_enter\_button = tkinter.Button(fourth\_frame, text='Solve', command=func1)
3. fourth\_window\_enter\_button.grid(column=2,

row=2,

pady=10,

padx=10)

1. # Create the button for the frame
2. fourth\_window\_back\_button = tkinter.Button(fourth\_frame,

text="Back",

1. command=call\_first\_frame\_on\_top)
2. fourth\_window\_back\_button.grid(column=0,

row=5,

pady=10,

padx=10,

sticky=(tkinter.N))

1. fourth\_window\_quit\_button = tkinter.Button(fourth\_frame,

text="Exit",

1. command=quit\_program)
2. fourth\_window\_quit\_button.grid(column=5,

row=5,

pady=10,

padx=10,

sticky=(tkinter.N))

1. def create\_widgets\_in\_fifth\_frame():
2. # Create the label for the frame
3. fifth\_window\_label = tkinter.ttk.Label(fifth\_frame,

text='Entrez un nombre:')

1. fifth\_window\_label.grid(column=1,

row=0,

pady=10,

padx=10,

sticky=(tkinter.N))

1. message = StringVar()
2. entry1 = tkinter.Entry(fifth\_frame, text='', textvariable=message, width=50, )
3. entry1.grid(column=2,
   * + - 1. row=0,
         2. pady=10,
         3. padx=10,
         4. sticky=(tkinter.N))
4. fifth\_window\_label = tkinter.ttk.Label(fifth\_frame,

text='Précision:')

1. fifth\_window\_label.grid(column=1,

row=1,

pady=10,

padx=10,

sticky=(tkinter.N))

1. many = StringVar()
2. message\_entry = Entry(fifth\_frame, text='', textvariable=many, width=20, )
3. message\_entry.grid(column=2,

row=1,

pady=10,

padx=10,

sticky=(tkinter.N))

1. def func1():
   * 1. try:
     2. x = float(entry1.get())
     3. a = int(message\_entry.get())
     4. fifth\_window\_label1.config(
        + 1. text=("{0:." + str(a) + "f}").format(

math.sqrt(-x) if x < 0 else math.sqrt(x))

* + - * 1. + ("i" if x < 0 else ""))
    1. except ValueError:
    2. fifth\_window\_label1.config(text="Erreur entrez les chiffres")

1. fifth\_window\_label1 = tkinter.Label(fifth\_frame,

text="Réponse:")

1. fifth\_window\_label1.grid(column=2,

row=3,

pady=10,

padx=10,

sticky=(tkinter.N))

1. fifth\_window\_label = tkinter.ttk.Label(fifth\_frame,

text="Réponse:")

1. fifth\_window\_label.grid(column=1,

row=3,

pady=10,

padx=10,

sticky=(tkinter.N))

1. # Create the button for the frame
2. fifth\_window\_enter\_button = tkinter.Button(fifth\_frame, text='Résoudre', command=func1)
3. fifth\_window\_enter\_button.grid(column=2,

row=2,

pady=10,

padx=10)

1. # Create the button for the frame
2. fifth\_window\_back\_button = tkinter.Button(fifth\_frame,

text="Retourner",

1. command=call\_first\_frame\_on\_top)
2. fifth\_window\_back\_button.grid(column=0,

row=5,

pady=10,

padx=10,

sticky=(tkinter.N))

1. fifth\_window\_quit\_button = tkinter.Button(fifth\_frame,

text="Sortir",

1. command=quit\_program)
2. fifth\_window\_quit\_button.grid(column=5,

row=5,

pady=10,

padx=10,

sticky=(tkinter.N))

1. def call\_first\_frame\_on\_top():
2. # This function can be called only from the second window.
3. # Hide the second window and show the first window.
4. second\_frame.place\_forget()
5. third\_frame.place\_forget()
6. fourth\_frame.place\_forget()
7. fifth\_frame.place\_forget()
8. first\_frame.place(relx=0.1, rely=0.1)
9. def call\_second\_frame\_on\_top():
10. # This function can be called from the first and third windows.
11. # Hide the first and third windows and show the second window.
12. first\_frame.place\_forget()
13. second\_frame.place(relx=0.1, rely=0.1)
14. def call\_third\_frame\_on\_top():
15. # This function can only be called from the second window.
16. # Hide the second window and show the third window.
17. first\_frame.place\_forget()
18. third\_frame.place(relx=0.1, rely=0.1)
19. def call\_fourth\_frame\_on\_top():
20. # This function can only be called from the second window.
21. # Hide the second window and show the third window.
22. first\_frame.place\_forget()
23. fourth\_frame.place(relx=0.1, rely=0.1)
24. def call\_fifth\_frame\_on\_top():
25. # This function can only be called from the second window.
26. # Hide the second window and show the third window.
27. first\_frame.place\_forget()
28. fifth\_frame.place(relx=0.1, rely=0.1)
29. def quit\_program():
30. root\_window.destroy()
31. ###############################
32. # Main program starts here :) #
33. ###############################
34. # Create the root GUI window.
35. root\_window = tkinter.Tk()
36. root\_window.title("Калькулятор квадратов")
37. root\_window.geometry("700x400")
38. root\_window.resizable(False, False)
39. # Create frames inside the root window to hold other GUI elements. All frames must be created in the main program, otherwise they are not accessible in functions.
40. first\_frame = tkinter.ttk.Frame(root\_window, width=750, height=450+300+200)
41. first\_frame.place(relx=0.1, rely=0.1)
42. second\_frame = tkinter.ttk.Frame(root\_window, width=750, height=450+300+200)
43. second\_frame.place(relx=0.1, rely=0.1)
44. third\_frame = tkinter.ttk.Frame(root\_window, width=750, height=450+300+200)
45. third\_frame.place(relx=0.1, rely=0.1)
46. fourth\_frame = tkinter.ttk.Frame(root\_window, width=750, height=450+300+200)
47. fourth\_frame.place(relx=0.1, rely=0.1)
48. fifth\_frame = tkinter.ttk.Frame(root\_window, width=750, height=450+300+200)
49. fifth\_frame.place(relx=0.1, rely=0.1)
50. # Create all widgets to all frames
51. create\_widgets\_in\_first\_frame()
52. create\_widgets\_in\_second\_frame()
53. create\_widgets\_in\_third\_frame()
54. create\_widgets\_in\_fourth\_frame()
55. create\_widgets\_in\_fifth\_frame()
56. # Hide all frames in reverse order, but leave first frame visible (unhidden).
57. second\_frame.place\_forget()
58. third\_frame.place\_forget()
59. fourth\_frame.place\_forget()
60. fifth\_frame.place\_forget()
61. # Start tkinter event - loop
62. root\_window.mainloop()
63. Application.EnableVisualStyles()
64. Application.SetCompatibleTextRenderingDefault(False)
65. form = MyForm()
66. Application.Run(form)

# ПЕРСОНАЛ И ОПЛАТА ТРУДА

1. Чепоков Елизар программист – 90ч на разработку программы, 6ч на разработку сайта, 50% оплаты
2. Караваев Александр – 5ч на разработку сайта, 3 часа на разработку документации, 50% оплаты

# 7. СВЯЗЬ С НАМИ

Наш сайт: [http:// bloodiesproject.gq](http://bloodiesproject.gq/index.html)

Горячая линия: **+79638739767**