**Python Program to solve Quadratic Equation**

**Purpose of program:**

This program can help people to understand:

1. Given the cosine or sine of an angle, finding the cosine or sine of the angle that is half as large involves solving a quadratic equation.

2. The process of simplifying expressions involving the square root of an expression involving the square root of another expression involves finding the two solutions of a quadratic equation.

**Target users:**

The mathematician can use this program to study the gold ratio is found as the solution of the quadratic equation *x²-x-1=0*. It is also possible to study the parabola for physicists, the specific expression of the equation is *y=ax²+bx+c.* Moreover, this program can help teachers in teaching. They have this program, it will double their working efficiency.

**How will they use the program?**

When user run this program, it will ask user to input a/b/c in ax²+bx+c=0 equation. Both of input number are integer. When user input wrong type of variable such as “@” and ”abc”, they’ll get an error message:” Please Enter Appropriate Values”. After typing these numbers, computer will output a message to users "Is your answer right?”. Hence, if users click the “Yes” button, computer will output a message randomly from "Great!", "Very good!", "Excellent!". In opposite, if users click the “No” button, computer will output a message to users "Keep tying.", "Check your calculation carefully.", "It must be wrong in your calculation somewhere."

**Which is the programming language?**

This program using python 3.5.0 to create. Sublime Text 3 and JetBrains PyCharm IDE to edit.

**Variables Table:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Scope** | **Description** |
| a | Float | quadratic | Get coefficient a from users’ input |
| b | Float | quadratic | Get coefficient b from users’ input |
| c | Float | quadratic | Get coefficient c from users’ input |
| delta | Float | quadratic | Calculate delta to judge root type |
| x1 | Complex | quadratic | Root 1 for the quadratic equation |
| x2 | Complex | quadratic | Root 2 for the quadratic equation |
| disc\_root | Complex | quadratic | Base number to calculate roots |
| correct\_word | Integer | correct | Random number chose from the list 0, 1, 2 |
| wrong\_word | Integer | wrong | Random number chose from the list 0, 1, 2 |

**Class Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | **Type** | **Scope** | **Description** |
| Application (Frame) | class | global | Application Class for the complete program |

**List Table**

|  |  |  |
| --- | --- | --- |
| **Type** | **Scope** | **Description** |
| correct\_wordlist | correct | Storing three list values in correct\_wordlist and displays when users get right result |
| wrong\_wordlist | wrong | Storing three list values in wrong\_wordlist and displays when users get wrong result |

**Testing Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** | **Test Result** |
| a=1  b=2  c=1 | Solve: x=(-1+0j) | -1+0j | Passed |
| a=2  b=2  c=1 | Solve:  x1=(-0.5+0.5j)  x2=(+0.5-0.5j) | -0.5+0.5j  -0.5-0.5j | Passed |
| a=5  b=2  c=1 | Solve:  x1=(-0.2+0.4j)  x2=(-0.2-0.4j) | -0.2+0.4j  -0.2-0.4j | Passed |

**Subroutine Table**

|  |  |
| --- | --- |
| **Subroutine Name** | **Description** |
| quadratic | Creates a solve quadratic equation subroutine |
| correct | Creates a subroutine that displays comments when users get the right answer |
| wrong | Creates a subroutine that displays comments when users get the wrong answer |

**Program Sketch**

**Solve Quadratic Equation**

**Solve**

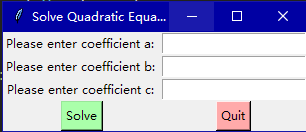
**Quit**

**Please enter coefficient c:**

**Please enter coefficient b:**

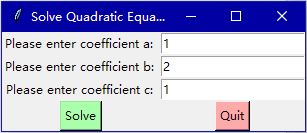
**Please enter coefficient a:**

**Program Code Screenshot – Explanation:**

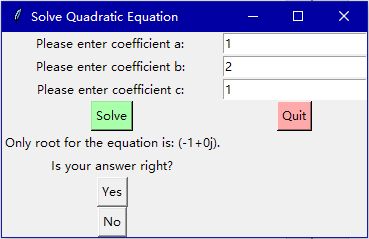


This is main GUI for the program. The progress of the program goes like the following.

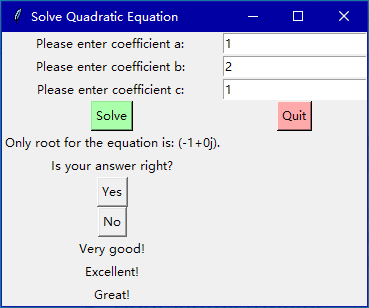
1. First, user input a/b/c (integer).



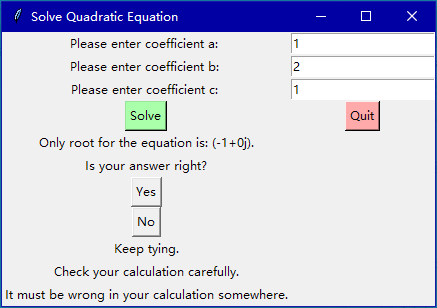
1. Click “Solve”, then the program will calculate the result of the equation and display down below.



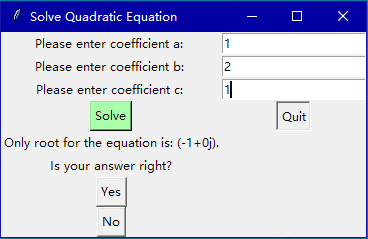
1. Programs among will give you a question “Is your answer right?” If you click button “Yes”, it will show the sentences randomly like below (Great!, Very good!, Excellent!).



Both of the programs will give users a question “Do you get the same answer?” If users click button “No”, it will show the sentences randomly like below (“Something wrong in your calculation”, “What a pity, it’s wrong.”, “Don’t give up! Do it again.”).

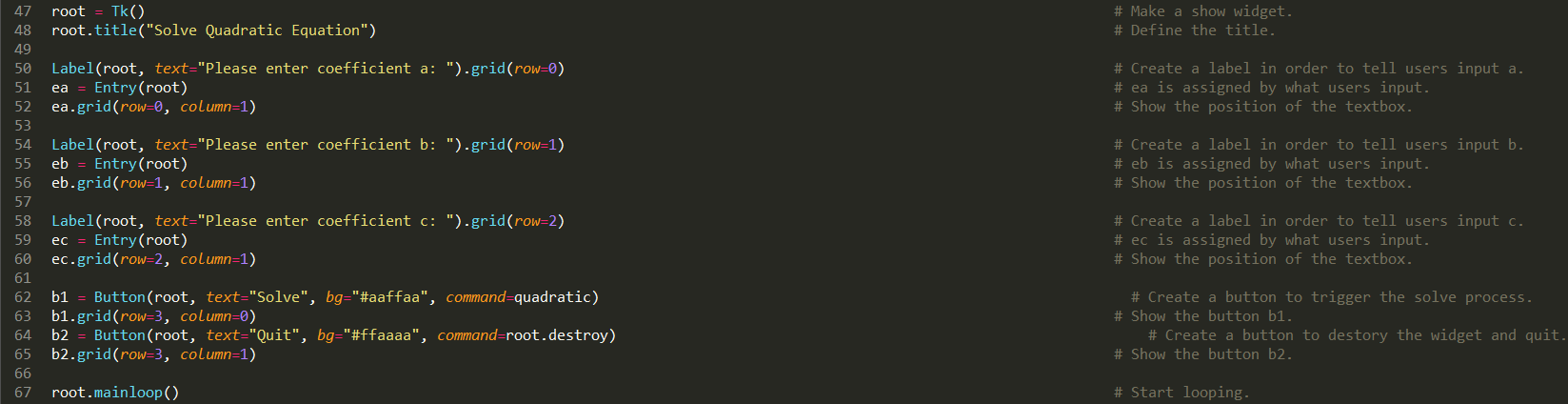


1. When users want to finish the program, they can click the button “Quit” to stop it



**Program Code – Screenshot:**





**Program Code:**

from tkinter import \* # Import tkinter for GUI.

import random # Import ramdom for wordlist control.

import cmath # Import cmath for calculate roots of the equation.

class Application(Frame): # Create a show widget.

globalnumber = 1 # Extend number scope to the whole program.

def quadratic(): # Quadratic subroutine to calculate the answer

try:

a = float(ea.get()) # Get a from ea.

b = float(eb.get()) # Get b from eb.

c = float(ec.get()) # Get c from ec.

delta = (b\*\*2) - (4\*a\*c) # Define the equation basic form.

if delta == 0: # When the equation has only one root.

disc\_root = cmath.sqrt(delta) # Use Complex Math to solve the equation.

x1 = (-b-disc\_root) / (2\*a) # Assign the value to x1.

Label(root, text="Only root for the equation is: "+str(x1)+".").grid() # Display the result.

Label(root, text="Is your answer right?").grid() # Create a label asks if you get the same answer.

Button(root, text="Yes", command=correct).grid() # Create a button "Yes" and output correct\_word list.

Button(root, text="No", command=wrong).grid() # Create a button "No" and outupt wrong\_word list.

else: # When the equation has two roots.

disc\_root = cmath.sqrt(delta) # Use Complex Math to solve the equation.

x1 = (-b-disc\_root) / (2\*a) # Assign the value to x1.

x2 = (-b+disc\_root) / (2\*a) # Assign the value to x2.

Label(root, text="Roots for the equation are: " + str(x1) + " and " + str(x2) + ".").grid() # Display the result.

Label(root, text="Is your answer right?").grid() # Create a label asks if you get the same answer.

Button(root, text="Yes", command=correct).grid() # Create a button "Yes" and output correct\_word list.

Button(root, text="No", command=wrong).grid() # Create a button "No" and outupt wrong\_word list.

except:

Label(root, text="Please Enter Appropriate Values").grid()

def correct(): # Create a correct word function.

correct\_word = int(random.randint(0, 2)) # Use random to choose a word in the wordlist.

correct\_wordlist = ["Great!", "Very good!", "Excellent!"] # Create a list to store three comments.

Label(root, text=correct\_wordlist[correct\_word]).grid() # Create a label to output the word that is randomly chose in the wordlist.

def wrong(): # Create a wrong word function.

wrong\_word=int(random.randint(0, 2)) # Use random to choose a word in the wordlist.

wrong\_wordlist=["Keep tying.", "Check your calculation carefully.", "It must be wrong in your calculation somewhere."] # Create a list to store three comments.

Label(root, text=wrong\_wordlist[wrong\_word]).grid() # Create a label to output the word that is randomly chose in the wordlist.

root = Tk() # Make a show widget.

root.title("Solve Quadratic Equation") # Define the title.

Label(root, text="Please enter coefficient a: ").grid(row=0) # Create a label in order to tell users input a.

ea = Entry(root) # ea is assigned by what users input.

ea.grid(row=0, column=1) # Show the position of the textbox.

Label(root, text="Please enter coefficient b: ").grid(row=1) # Create a label in order to tell users input b.

eb = Entry(root) # eb is assigned by what users input.

eb.grid(row=1, column=1) # Show the position of the textbox.

Label(root, text="Please enter coefficient c: ").grid(row=2) # Create a label in order to tell users input c.

ec = Entry(root) # ec is assigned by what users input.

ec.grid(row=2, column=1) # Show the position of the textbox.

b1 = Button(root, text="Solve", bg="#aaffaa", command=quadratic) # Create a button to trigger the solve process.

b1.grid(row=3, column=0) # Show the button b1.

b2 = Button(root, text="Quit", bg="#ffaaaa", command=root.destroy) # Create a button to destory the widget and quit.

b2.grid(row=3, column=1) # Show the button b2.

root.mainloop() # Start looping.

**Flow Chart:**

Start

Run the “Main” subroutine

Import the GUI and complex math functions

Create the Application class

Make three labels called enter a/b/c

Wrong

Correct

Solve Quadratic Equation

Name GUI window as “Root”

Main

Start loop until something happens

Solve Quadratic Equation

Get a/b/c from input as float number

Calculate Delta Δ=b^2-4ac

Yes

Calculate Delta Δ=b^2-4ac

If delta = 0

No

Use Complex Math Module to sqrt delta

Use Complex Math Module to sqrt delta

Calculate the only root x1

Calculate two roots x1/x2

Create a label to display result

Create a button “No”

Create a button “Yes”

Create a label display:” Is your answer right? ”

Return

Quit

End

Create a wordlist

Import random function

Wrong

Correct

Import random function

Create a wordlist

Create a label display correct\_wordlist[correct\_word]

Create a label display wrong\_wordlist[wrong\_word]

Return

Return