East West University

Department of Computer Science and Engineering
CSE360: Computer Architecture
Section- 3, FALL 2022

Project report

Multifunctional Calculator

Submitted To:

Dr. Md. Nawab Yousuf Ali

Professor, Department of CSE

Submitted By:

Group No-2

Name	ID No
Mysha Maliha Priyanka	2020-1-60-230
K. M. Safin Kamal	2020-1-60-235
Al Hossain	2020-1-60-016
Umme Hani Konok	2020-1-60-113

Date of Submission: 03/ 01 / 2023

Multifunctional Calculator

Al Hossain (2020-1-60-016), K. M. Safin Kamal (2020-1-60-235), Mysha Maliha Priyanka (2020-1-60-230), Umme Hani Konok (2020-1-60-113)

Department of Computer Science and Engineering, East West University, Dhaka 1212, Bangladesh

Abstract: **Our main objective is to develop a multifunction calculator using python. These four-functions are addition, subtraction, multiplication and division. However, we will also have some extra functions so that we could give our calculator a more scientific feature to make our project a little challenging and have a good outcome. For this, here we will also calculate sort, root, square root, power functions (individually having square and cubic power button for ease), logarithm (log_{10}), natural logarithm (ln), exponential value (e), factorial, degree, radian, percentage, trigonometric functions(sin, cos, tan), pie value calculations, number system converter(Binary, Decimal, Octal, Hexa) etc.**

Keywords: Tkinter, GUI, features, calculator, input, output, interface.

I. Introduction

A Calculator is a small, portable, often inexpensive electronic device that is used to perform both basic complex operation and arithmetic.

Our calculator can do such basic calculations as well as have scientific calculating features. It can help people of every phase or life. Educational along with office purpose and so on.

- Language: Python
- Software: PyCharm, VS code

II. Related works

The first tool created specifically for use in mathematical computations was the abacus, likely invented in Sumeria around 2500 B.C. In 1617, Scottish mathematician John Napier published Rabdology, or "calculation with rods" The user determines their sum by changing the vertical alignment of the rods, and horizontally reading the multiplication tables. In 1642, the first true "calculator" was invented: one that performed calculations through a clockwork-type mechanism. In 1945, Curt Herzstark invented the first handheld, mechanic calculator, from a design he had created in 1938. The calculators we know today were not invented until the 1970s, and the use of smartphones as calculators did not begin until at least the late nineties. Modern pocket calculators were developed after the invention of the smartphone in 1995.

III. Proposed work

Our proposed project is about multifunctional calculator. Which have many functionalities like a scientific calculator but more than a normal calculator. Our calculator is mainly for educational and office both purpose. As we make the calculator user friendly where users can easily go through the features perfectly, user will going to enjoy the necessary calculations.

IV. Experimental result

Our code runs perfectly. All the functions work without any error. We have tested our code several times and run the project completely.
Results analysis:

No.	Input	instruction	Output
1	7+4	addition	11
2	7-4	substract	3
3	7*4	multiply	28
4	7/4	divide	1.75
5	7!	factorial	50402``
6	7%	percentage	0.07
7	7^2	square	49
8	7^3	cube	343
9	7^4	power	2401
10	7^-1	inverse	0.14285714285714285
11	root7	root	2.6457513110645907
12	log7	Log	0.8450980400142568
13	345231	sort	1 2 3 3 4 5
14	Ln1	ln	0
15	sin30	sin	0.5
16	cos30	cos	0.8660254037844387
17	tan30	tan	0.5773502691896257
18	30	degree	1718.8733853924696
19	tan-1(30)	tan^-1	88.09084756700362
20	sin^-1(0.1)	sin^-1	5.739170477266787
21	cos^-1(0.1)	cos^-1	84.26082952273322
22	100	radian	1.7453292519943295
23	12	D-bin	1100
24	12	D-hex	c
25	12	D-oct	14
26	1100	B-dec	12
27	1100	b-hex	c
28	1100	b-oct	14
29	c	h-dec	12
30	c	h-bin	1100
28	c	h-oct	14
29	14	o-dec	12
30	14	o-bin	1100
31	14	o-hex	c

V. Conclusion and future enhancements

This initiative has been really trustworthy and educational. It has forced us to acquire and comprehend the numerous pointless ideas of the Python language. Since we've used Python Tkinter as our GUI, it offers a variety of controls, like buttons, labels, and text boxes, to help us create a user-friendly program. The projected project has a bright future, as seen by the rapidly expanding use of the internet. Finally, it has provided us with a priceless lesson on the benefits of working with others and socializing in a group.
Future enhancements of this project, which we can think of now:

- Make this calculator more interactive by using input as voice command.
- We can use deep learning and Artificial Intelligence
- Implement graphs and charts

Reference:

[1]Cherry, L., & Morris, R. (1978). BC—An Arbitrary Precision Desk-Calculator Language. *Bell Laboratories*.

[2] Klarmann, S., & Wackerfuß, J. (2021). Possibilities and drawbacks using arbitrary precision numbers for structural analysis. *PAMM*, *20*(1), e202000079.

[3] Dzhambov, V., & Drangajov, S. (2011). Computing of special functions with arbitrary precision in the environment of .NET framework. *Cybernetics and Information Technologies*, *11*(2), 32-45.

[4] Morris, R. (1997). BC− An Arbitrary Precision Desk-Calculator Language Lorinda Cherry.

Appendix:

Code:

```
from tkinter import *
import math

def click(value):
    ex = field.get()
    result = ' '

    try:

        if value == 'DEL':
            ex = ex[0:len(ex)-1]        #using slicing to delete last character
            field.delete(0,END)
            field.insert(0,ex)
            return

        elif value == 'CE':       #delete everything
            field.delete(0,END)

        elif value == '√':
            result = math.sqrt(eval(ex))    #evaluate int float everything

        elif value == 'π':
            result = math.pi

        elif value == 'cosθ':
            result = math.cos(math.radians(eval(ex)))

        elif value == 'tanθ':
            result = math.tan(math.radians(eval(ex)))

        elif value == 'sinθ':
            result = math.sin(math.radians(eval(ex)))

        elif value == 'Sort':
            result = [int(x) for x in str(eval(ex))]
            result.sort()

        elif value == 'sin-1':
            result = math.asin(eval(ex))
            result = math.degrees(result)

        elif value == 'cos-1':
            result = math.acos(eval(ex))
            result = math.degrees(result)

        elif value == 'tan-1':
            result = math.atan(eval(ex))
            result = math.degrees(result)

        elif value == chr(8731):
            result = eval(ex) ** (1 / 3)

        elif value == 'x\u02b8':
            field.insert(END, '**')
            return

        elif value == 'x\u00B3':
            result = eval(ex) ** 3
```

```
elif value == 'x\u00B2':
    result = eval(ex) ** 2

elif value == 'ln':
    result = math.log2(eval(ex))

elif value == 'Deg':
    result = math.degrees(eval(ex))

elif value == "Rad":
    result = math.radians(eval(ex))

elif value == '%':
    result = eval(ex)/100

elif value == 'Exp':
    result = math.e

elif value == 'log₁₀':
    result = math.log10(eval(ex))

elif value == 'x!':
    result = math.factorial(eval(ex))

elif value == chr(247):
    field.insert(END, "/")
    return

elif value == '=':
    result = eval(ex)

elif value == 'D-Bin':
    result = bin(eval(ex)).replace("0b","")

elif value == 'D-Hex':
    result = hex(eval(ex)).replace("0x","")

elif value == 'D-Oct':
    result = oct(eval(ex)).replace("0o","")

elif value == 'B-Dec':
    result = int(ex, 2)

elif value == 'B-Hex':
    decimal = int(ex, 2)
    result = hex(decimal).replace("0x","")

elif value == 'B-Oct':
    decimal = int(ex, 2)
    result = oct(decimal).replace("0o", "")

elif value == 'H-Dec':
    result = int(ex, 16)

elif value == 'H-Bin':
    decimal = int(ex, 16)
    result = bin(decimal).replace("0b","")

elif value == 'H-Oct':
    decimal = int(ex, 16)
    result = oct(decimal).replace("0o", "")
```

```
        elif value == 'O-Dec':
            result = int(ex, 8)

        elif value == 'O-Bin':
            decimal = int(ex, 8)
            result = bin(decimal).replace("0b","")

        elif value == 'O-Hex':
            decimal = int(ex, 8)
            result = hex(decimal).replace("0x", "")

        elif value == 'x-1':
            result = 1/eval(ex)

        elif value == 'Abs':
            result = math.fabs(eval(ex))

        else:
            field.insert(END, value)
            return

        field.delete(0,END)
        field.insert(0,result)

    except SyntaxError:
        pass

root = Tk()     #Tk class object is root
root.title('Arbitrary Precision Calculator')
root.config(bg='white')
root.geometry('420x713+700+200')

field=Entry(root, font=('arial',18,'bold'), bg='black', fg='white', bd=15,
relief=SUNKEN, width=25)        #fg means font color & Entry class's object is field
field.grid(row=0,column=0,columnspan=6)

button_text_list = ["D-Bin", "D-Hex", "D-Oct", "sinθ", "cosθ", "tanθ",
                    "B-Dec", "B-Hex", "B-Oct", "sin-1", "cos-1", "tan-1",
                    "H-Dec", "H-Bin", "H-Oct", "ln", "Deg", "Rad",
                    "O-Dec", "O-Bin", "O-Hex", "Sort", "(", ")",
                    "A", "B", "C", "%", "DEL", "CE",
                    "D", "E", "F",chr(247), "x-1", "x!",
                    "7", "8", "9", "*", "x\u02b8", "log₁₀",
                    "4", "5", "6", "+", "x\u00B2", "√",
                    "1", "2", "3", "-", "x\u00B3", chr(8731),
                    "0", ".", "π", "=", "Exp", "Abs"]

row_no = 1
column_no = 0
for i in button_text_list:
    button = Button(root, width=5, height=2, bd=2, relief=SUNKEN, text=i, bg='black',
fg='white',
                    font=('arial', 14, 'bold'), activebackground='white',
                    command=lambda button=i:click(button))       #Button class object
is button

    button.grid(row=row_no,column=column_no,pady=2)
    column_no = column_no + 1
    if column_no > 5:
        row_no = row_no + 1
        column_no = 0

root.mainloop()
```