Calculator: Shared C libraries in Python

Hemanth Kumar Desineedi and G V V Sharma*

Contents

Abstract—**This manual shows how to build a calculator using Python and shared C libraries. Through this, even beginners can learn how to build some simple software applications with graphical user interfaces (GUIs).**

1 Python Calculator

Problem 1. Download the python code from [1] and execute it.

2 Shared Libraries in GCC

Problem 2. Write a C function to multiply two given numbers. Save it in the file titled as **mul.c**

Solution:

```
//function to multiply two numbers

float mul(float num1, float num2)
    {
     return num1*num2; //function
        returns multiplication of
        num1 and num2
    }

//Run the following commnad for
    generating the .so file
//cc -fPIC -shared -o mul.so mul.c
```

*The author is with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadepall@iith.ac.in.

Problem 3. Open the Terminal and go to the directory where the **mul.c** file is saved.

Problem 4. Type the following command in the Terminal.

Solution:

```
cc -fPIC -shared -o mul.so mul.c
```

Problem 5. Type the following program in **main.c**

Solution:

```
#include <stdio.h>

float mul(float,float);

int main(void)
{
printf("%f\n",mul(4,5));
return 0;
}
//gcc main.c mul.so -Wl,-rpath=$(
    pwd)
```

Problem 6. Run the above program

Solution:

```
gcc main.c mul.so -Wl,-rpath=$(pwd
    )
./a.out
```

The advantange of using **mul.so** is that the multiplication function needs to be compiled only once. It can then be used in any C program.

Problem 7. Repeat the above exercises for adding two numbers.

Problem 8. Write all the required C routines for the calculator in Problem 1 and generate the shared libraries.

3 Shared libraries in Python

Problem 9. Write a Python script to multiply two numbers using C function.

Solution:

```
#Calling C function in Python
from ctypes import *

#load the shared object file
multip = CDLL('./mul.so')

a=2.0
b=8.0

#Find multiplication of floats

mul = multip.mul
mul.restype = c_float

print (a,"x",b,"=", mul(c_float(a)
    , c_float(b)))
```

Problem 10. Call the function written above in the Python GUI calculator to perform multipication.

Solution: Download **calc_mul_root.py** file from the [2] and save it in directory where **mul.c** is saved. Execute **calc_mul.py**.

Problem 11. Use C routines in **calc_mul_root.py** for all arithmetic operations in the calculator.

References

[1] A. Deep. (2018) Python Calculator. [Online]. Available: https://github.com/gadepall/EE1083/blob/master/calculator/codes/solution/pythonprogs/tkcalc.py

[2] H. Kumar. (2018) Python Calculator using Shared Libraries. [Online]. Available: https://github.com/gadepall/EE1083/blob/master/calculator/codes/solution/pythonprogs/calc_mul_root.py