**#Program 3: Python script for GUI**

#!/usr/bin/python

from gi.repository import Gtk

import os

import serial\_graphics

**#Program to give GUI for the Project ADDER Authors: Musaddiq Faraz, Sameer Shaik, Shashi Kiran Reddy, Sujay**

class GUI(Gtk.Window):

**‘’’Provides the main window for the GUI’’’**

def toggle\_graphics(self,check):

if (self.show1 == 1):

self.show1 = 0

self.entry\_3.hide()

self.entry\_4.hide()

self.button2.hide()

self.label\_3.hide()

else:

self.show1=1

self.entry\_3.show()

self.entry\_4.show()

self.label\_3.show()

self.button2.show()

def \_\_init\_\_(self):

**#the constructor of the GUI Class**

Gtk.Window.\_\_init\_\_(self, title="ADDER")

self.set\_position(Gtk.WindowPosition.CENTER)

self.set\_border\_width(15)

self.entry\_text\_1 = "null"

self.entry\_text\_2 = "null"

table = Gtk.Table(10, 10, False)

self.add(table)

self.label = Gtk.Label("Answer : " + answer)

button = Gtk.Button("ADD")

button.connect("clicked", self.button\_pressed)

label\_app = Gtk.Label("Application Adder ")

label\_1 = Gtk.Label("Enter number1: ")

entry\_1 = Gtk.Entry()

entry\_1.connect("changed", self.enter\_callback\_1, entry\_1)

label\_2 = Gtk.Label("Enter number2: ")

entry\_2 = Gtk.Entry()

entry\_2.connect("changed", self.enter\_callback\_2, entry\_2)

self.label\_3 = Gtk.Label("Enter 4-bit Number ")

self.entry\_3 = Gtk.Entry()

self.entry\_4 = Gtk.Entry()

self.entry\_3.set\_max\_length(4)

self.entry\_4.set\_max\_length(4)

self.show1 = 1

self.button2 = Gtk.Button("Simulate")

self.button2.connect("clicked", self.button2\_pressed)

check = Gtk.CheckButton("4-bit Graphical Adder")

check.set\_active(True)

check.connect("toggled", self.toggle\_graphics)

check.show()

about = Gtk.Button("About")

about.connect("clicked", self.aboutdig)

table.attach(label\_app,1,10,1,2)

table.attach(label\_1,1,2,2,3)

table.attach(entry\_1,2,10,2,3)

table.attach(label\_2,1,2,3,4)

table.attach(entry\_2,2,10,3,4)

table.attach(self.label, 1, 10, 5, 6)

table.attach(button, 2, 10, 4, 5)

table.attach(about, 1, 2, 4, 5)

table.attach(check,1,10,6,7)

table.attach(self.label\_3,1,3,7,8)

table.attach(self.entry\_3,3,5,7,8)

table.attach(self.entry\_4,5,7,7,8)

table.attach(self.button2,7,10,7,8)

def aboutdig(self, widget):

**#defines the about dailog box which opens on clicking the about button**

message="Program to simulate adder...\nAuthors: Musaddiq Faraz, Sameer Shaik, Shashi Kiran Reddy, Sujay"

dialog = Gtk.MessageDialog(self, 0, Gtk.MessageType.INFO,Gtk.ButtonsType.OK, "About the program")

dialog.format\_secondary\_markup(message)

dialog.run()

dialog.destroy()

def button2\_pressed(self, button):

**#executes on clicking the simulate button**

x = self.entry\_3.get\_text()

y = self.entry\_4.get\_text()

c = ""

temp1 = int(x)

temp2 = int(y)

s = str(temp1 + temp2)

while temp1!=0 and temp2!=0 :

if (temp1%10+temp2%10 > 9):

c = c + '1'

else:

c = c + '0'

temp1 = temp1/10

temp2 = temp2/10

x = correct(x[::-1])

y = correct(y[::-1])

c = correct(c[::-1])

s = correct(s[::-1])

c = '0' + c

serial\_graphics.main(x,y,c,s)

def button\_pressed(self, button):

**#executes on clicking the a button**

os.system("gcc main.c -o add")

if (os.name == "posix"):

command="./add "+self.entry\_text\_1+" "+self.entry\_text\_2+" >ans"

else:

command="add "+self.entry\_text\_1+" "+self.entry\_text\_2+" >ans"

os.system(command)

fobj = open("ans","r")

answer = fobj.readline()

answer = answer[:len(answer)-1]

fobj.close()

os.system("rm add")

self.label.set\_text("Sum = "+answer)

def enter\_callback\_1(self, widget, entry\_1):

**#called when the entry box1 is edited**

self.entry\_text\_1 = entry\_1.get\_text()

return

def enter\_callback\_2(self, widget, entry\_2):

**#called when the entry box1 is edited**

self.entry\_text\_2 = entry\_2.get\_text()

return

def correct(str1):

**#Correcting the number of digits entered to 4, for simulation**

len1=len(str1)

while len1<4:

str1 = str1 + '0'

len1+=1

return str1

entry\_text\_1="null"

answer="null"

win = GUI()

win.connect("delete-event", Gtk.main\_quit)

win.show\_all()

Gtk.main()