Lab 2 : Python GUI Programming Report

學號：0410835 姓名：鄭余哲

1. 請簡述Python Tkinter GUI程式主體架構 (如何引入模組.宣告主視窗及物件…等等)。(30分)

一開始先以 Import Tkinter 引入Tkinter的模組，再以top=Tkinter.Tk()來創造一個top主視窗，宣告完主視窗之後可以創造其他widget物件(Frame、Label、Text…等等)，接者以pack、grid、place等function將widget放在parent物件上

最後以top.mainloop()將Tkinter物件放入等待迴圈，進入迴圈之後就會等待事件，並對各個事件做出event handle(自定義)

1. 請解釋Python計算機程式程式碼(將程式貼上並加上註解) 。(50分)

(請詳細標注能實現的功能，如 : ／０能顯示錯誤訊息，０／任何數字為０，第一次計算完的結果能繼續做運算，在上一次結果的畫面繼續輸入數字時不會接著原本的數字…等等)

import tkinter

from tkinter import font

top = tkinter.Tk()#創造主視窗

f1 = tkinter.Frame(top)#創造frame1

f2 = tkinter.Frame(top)#創造frame2

ft = font.Font(size = 30)#設定使用字體大小

var = tkinter.StringVar()#宣告Tkinter使用的字串變數

var.set("0")#初始字串變數為0

flag = 1#宣告flag，用來判斷狀態是否為剛初始還是計算中

def SetValue():

Screen = tkinter.Label(f1,textvariable = var, font = ft).grid(row = 0,column =1)#設定計算機螢幕顯示運算結果並放於frame1

def Click(x):#按件功能

global flag#要求使用global變數flag

if flag == 1 or var.get() == "ERROR":#如果為初始值或ERROR特判狀態

flag = 0#改狀態為計算中

if var.get()=="ERROR":#如果為ERROR直接更新內容

var.set(x)

return

if x == " x " or x == " / " or x == " + ":#運算符號狀態(讓運算結果可以繼續做運算)

var.set(var.get()+x)#串聯運算符號於字串變數

return

if var.get()!= "0" and x == " - ":運算符號狀態(讓運算結果可以繼續做運算)

var.set(var.get()+x)

return

var.set(x)#如果是輸入數字修改為新的數字

return

var.set(var.get()+x)#如果是初始狀態 直接改字串內容

def Clear():#清除螢幕功能

var.set("0")#清空數字內容

global flag#要求使用global變數flag

flag = 0#設定為初始狀態

def Calculate():#計算功能

tmp = var.get().split()#切開螢幕顯示內容

if tmp[0]=="x" or tmp[0]=="/":#如果第一位為乘除顯示ERROR

var.set("ERROR")

return

base = 0;#設定基準點(第一個數字位置為0)

if tmp[0] == "+":#如果第一位為加減 設定基準點為1(使負數可以加入計算)

base = 1

if tmp[0] == "-":

base = 1

tmp[1] = "-"+ tmp[1]#更新第一位數字為負的

ans = int(tmp[base])#將第一位數字先暫訂為答案

if (len(tmp) - base) % 2 == 0:#如果尾數停在運算符號(基數是數字偶數是運算符號)顯示ERROR

var.set("ERROR")

return

for l in range(base+1,len(tmp),2):#用for迴圈一一做運算

if tmp[l] == '+':

ans+=int(tmp[l+1])

if tmp[l] == '-':

ans-=int(tmp[l+1])

if tmp[l] == 'x':

ans\*=int(tmp[l+1])

if tmp[l] == '/':

if tmp[l+1] == '0':

var.set("ERROR")

return

ans/=int(tmp[l+1])

global flag

flag = 1

var.set(str(ans))#設定變數字串為運算結果

#依照規定設定widget的layout方式並設定button的handler function

SetValue()

Btn7 = tkinter.Button(f2, text = "7" , borderwidth=5, width = 4, font = ft, command = lambda: Click("7")).grid(row = 0, column = 0)

Btn8 = tkinter.Button(f2, text = "8" , borderwidth=5, width = 4, font = ft, command = lambda: Click("8")).grid(row = 0, column = 1)

Btn9 = tkinter.Button(f2, text = "9" , borderwidth=5, width = 4, font = ft, command = lambda: Click("9")).grid(row = 0, column = 2)

BtnAdd = tkinter.Button(f2, text = "+" , borderwidth=5, width = 4, font = ft, command = lambda: Click(" + ")).grid(row = 0, column = 3)

Btn4 = tkinter.Button(f2, text = "4" , borderwidth=5, width = 4, font = ft, command = lambda: Click("4")).grid(row = 1, column = 0)

Btn5 = tkinter.Button(f2, text = "5" , borderwidth=5, width = 4, font = ft, command = lambda: Click("5")).grid(row = 1, column = 1)

Btn6 = tkinter.Button(f2, text = "6" , borderwidth=5, width = 4, font = ft, command = lambda: Click("6")).grid(row = 1, column = 2)

Btnmin = tkinter.Button(f2, text = "-" , borderwidth=5, width = 4, font = ft, command = lambda: Click(" - ")).grid(row = 1, column = 3)

Btn1 = tkinter.Button(f2, text = "1" , borderwidth=5, width = 4, font = ft, command = lambda: Click("1")).grid(row = 2, column = 0)

Btn2 = tkinter.Button(f2, text = "2" , borderwidth=5, width = 4, font = ft, command = lambda: Click("2")).grid(row = 2, column = 1)

Btn3 = tkinter.Button(f2, text = "3" , borderwidth=5, width = 4, font = ft, command = lambda: Click("3")).grid(row = 2, column = 2)

Btnmul = tkinter.Button(f2, text = "x" , borderwidth=5, width = 4, font = ft, command = lambda: Click(" x ")).grid(row = 2, column = 3)

Btn0 = tkinter.Button(f2, text = "0" , borderwidth=5, width = 4, font = ft, command = lambda: Click("0")).grid(row = 3, column = 0)

BtnClear = tkinter.Button(f2, text = "C" , borderwidth=5, width = 4, font = ft, command = Clear).grid(row = 3, column =1)

BtnEqual = tkinter.Button(f2, text = "=" , borderwidth=5, width = 4, font = ft, command = Calculate).grid(row = 3, column = 2)

BtnDiv = tkinter.Button(f2, text = "/" , borderwidth=5, width = 4, font = ft, command = lambda: Click(" / ")).grid(row = 3, column = 3)

f1.pack()#將frame1放到主視窗上

f2.pack()#將frame2放到主視窗上

top.mainloop()

1. 心得與建議(100~150字) (20分)

計算機不能執行先乘除後加減，使用以來還是覺得很詭異，後來自己實驗結束後使用了eval，可以先乘除後加減之後才感覺比較像簡易計算機，不過由於次第一次用tkinter有點不習慣，不過跟android的widget感覺有些相似，所以在寫handler function比較快，而layout的部分，android可以直接以拖拉的方式設定，感覺比較方便。

助教人很好，會幫忙解決問題，也會講一下實驗內容，不過我覺得如果課前可以多稍微講一下可可能會更好，當然我也不排斥自己找方法實現而得到的成就感。

實驗內容，我覺得整體而言算蠻簡單的，也不太需要花時間，算是一個給剛接觸python很好的練習機會，由於不善常寫python的語法，還是會常常寫成C的語法，以至於常常想法對，可是卻會syntax error，剛開始，不習慣的時候，會有點挫折感，但在習慣之後 python感覺比c 好寫也比較方便，不過畢竟python是比較高階的語言，如果想要縮短程式執行時間，相同的function功能，在C上實作還是會比較快，所以儘管比較麻煩，C還是學好比較好。