| **Laboratory Activity No. 10** | |
| --- | --- |
| **Polymorphism** | |
| **Course Code:** CPE009 | **Program:** BSCPE |
| **Course Title:** Object-Oriented Programming | **Date Performed:4/29/2024** |
| **Section: CPE12S6** | **Date Submitted:4/29/2024** |
| **Name: Julongbayan, Danielle   Quillan, Precious** | **Instructor:Ma’am Sayo** |
| **1. Objective(s):** | |
| This activity aims to familiarize to convert a TUI program to GUI program in Object-Oriented Programming | |
| **2. Intended Learning Outcomes (ILOs):** | |
| The students should be able to:  2.1 Identify the use of Polymorphism in Object-Oriented Programming  2.2 Implement an Object-Oriented Program that converts text-user interface to a graphical-user interface. | |
| **3. Discussion:** | |
| In general, programs consist of three components—input, processing, and output. In TUI programs, input is usually obtained from an input statement or by importing data from a file. Output is usually given by a print statement or stored in a file. When we convert a TUI program to a GUI program, we replace input and print statements with Label/Entry pairs. Processing data and inputting and outputting data to files works much the same in both types of programs. The primary difference is that the processing in GUI programs is usually triggered by an event. | |
| **4. Materials and Equipment:** | |
| Windows Operating System  Google Colab  Pycharm Community version/Anaconda Navigator 2020 or higher version | |
| **5. Procedure:** | |
| **Method 1**  1. Type these codes in Pycharm,  #TUI Form  def main():  # Find the largest number among three numbers  L = []  num1 = eval(input("Enter the first number:"))  L.append(num1)  num2 = eval(input("Enter the second number:"))  L.append(num2)  num3 = eval(input("Enter the third number:"))  L.append(num3)  print("The largest number among the three is:",str(max(L)))  main() | |

**Method 2**

2. Type these codes in Pycharm,

from tkinter import \*

window = Tk()

window.title("Find the largest number")

window.geometry("400x300+20+10")

def findLargest():

L = []

L.append(eval(conOfent2.get()))

L.append(eval(conOfent3.get()))

L.append(eval(conOfent4.get()))

conOfLargest.set(max(L))

lbl1 = Label(window, text = "The Program that Finds the Largest Number") lbl1.grid(row=0, column=1, columnspan=2,sticky=EW)

lbl2 = Label(window,text = "Enter the first number:")

lbl2.grid(row=1, column = 0,sticky=W)

conOfent2 = StringVar()

ent2 = Entry(window,bd=3,textvariable=conOfent2)

ent2.grid(row=1, column = 1)

lbl3 = Label(window,text = "Enter the second number:")

lbl3.grid(row=2, column=0)

conOfent3=StringVar()

ent3 = Entry(window,bd=3,textvariable=conOfent3)

ent3.grid(row=2,column=1)

lbl4 = Label(window,text="Enter the third number:")

lbl4.grid(row=3,column =0, sticky=W)

conOfent4 = StringVar()

ent4 = Entry(window,bd=3,textvariable=conOfent4)

ent4.grid(row=3, column=1)

btn1 = Button(window,text = "Find the largest no.",command=findLargest) btn1.grid(row=4, column = 1)

lbl5 = Label(window,text="The largest number:")

lbl5.grid(row=5,column=0,sticky=W)

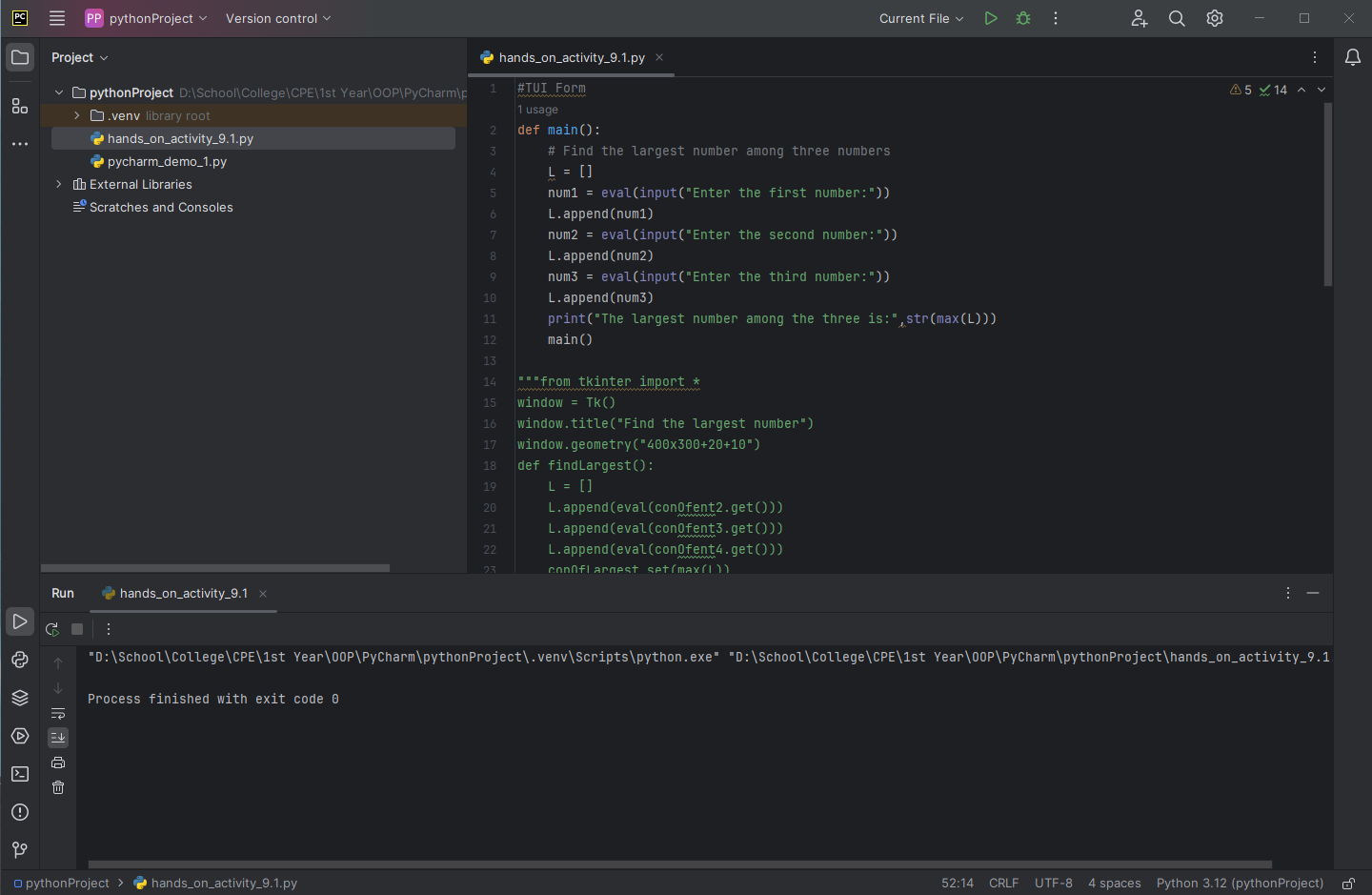
conOfLargest = StringVar()

ent5 = Entry(window,bd=3,state="readonly",textvariable=conOfLargest) ent5.grid(row=5,column=1)

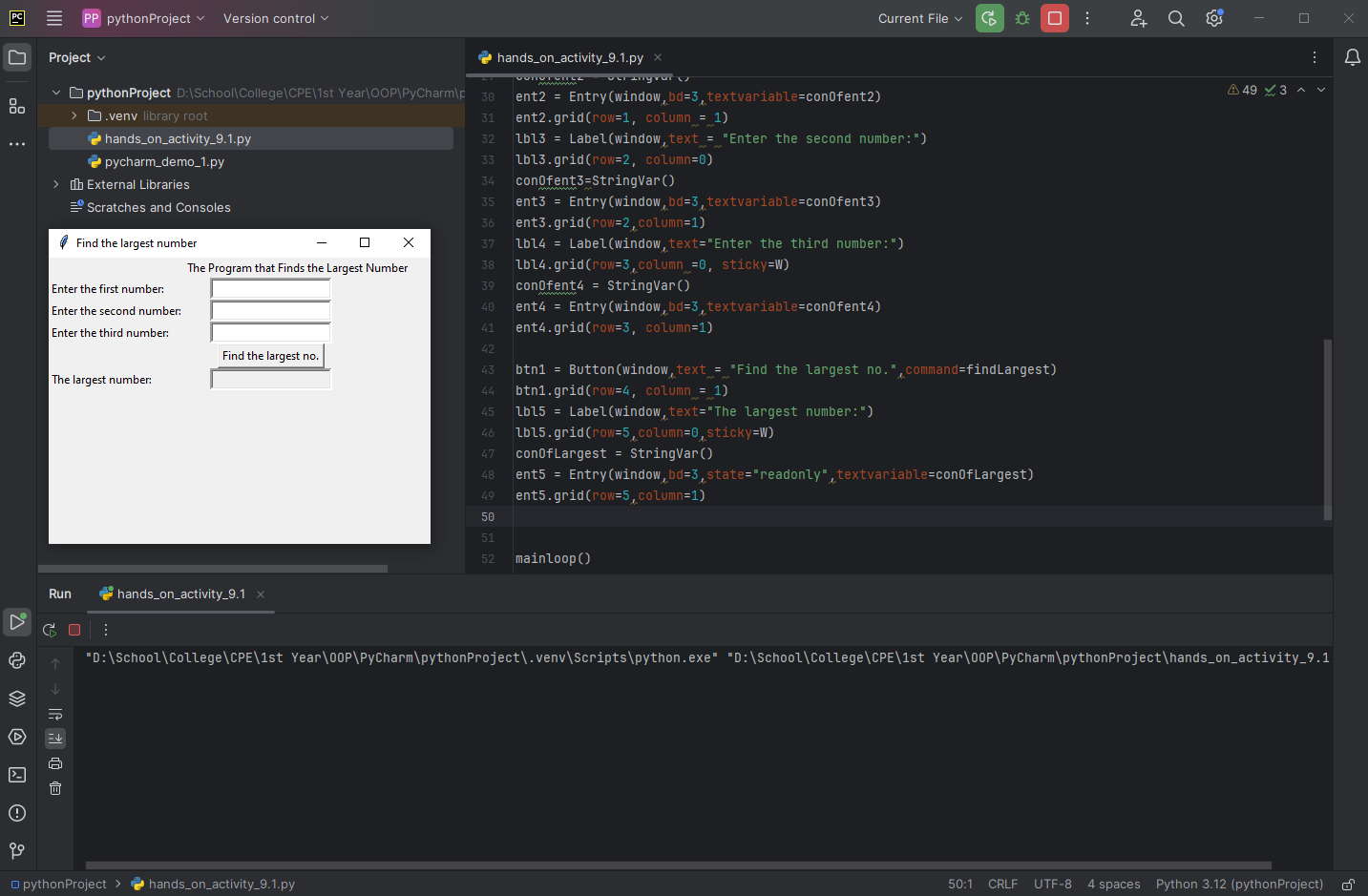
mainloop()

3. Screenshot of Method 1 and Method 2 output program:

**METHOD 1:**

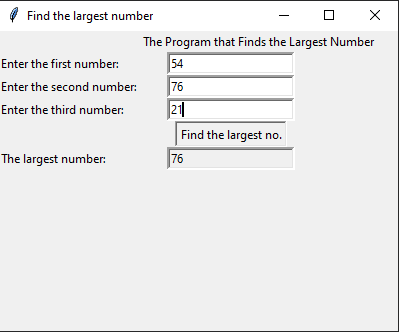


**METHOD 2:**



4. Run the program and observe the output.

5. Observation:



The program ran by launching a GUI and being able to input numbers and after pressing the button, It searched for the largest number and displayed the Largest number at the bottom text box.

|  |
| --- |
| **6. Supplementary Activity:** |
| Create a program that will compute the average of three grades (Grade1, Grade2, Grade3) in both TUI and GUI forms. Run the programs and take screenshots of each output. Observe and explain each solution (e.g. advantages and/or disadvantages)  **TUI:**      **GUI:**    I was able to compute the average using the block of code done below. The program will be included in the drive folder. The format for Method 2 was already enough to find the answer for GUI. As I just had to rename and retype minor stuff and changed max(L) to sum(L)/3 to compute the average of the elements inside of Array L.  from tkinter import \*  window = Tk()  window.title("Find the largest number")  window.geometry("400x300+20+10")  def findLargest():  L = []  L.append(eval(conOfent2.get()))  L.append(eval(conOfent3.get()))  L.append(eval(conOfent4.get()))  conOfLargest.set(sum(L)/3)  lbl1 = Label(window, text = "The Program that Averages 3 grades")  lbl1.grid(row=0, column=1, columnspan=2,sticky=EW)  lbl2 = Label(window,text = "Enter the first grade:")  lbl2.grid(row=1, column = 0,sticky=W)  conOfent2 = StringVar()  ent2 = Entry(window,bd=3,textvariable=conOfent2)  ent2.grid(row=1, column = 1)  lbl3 = Label(window,text = "Enter the second grade:")  lbl3.grid(row=2, column=0)  conOfent3=StringVar()  ent3 = Entry(window,bd=3,textvariable=conOfent3)  ent3.grid(row=2,column=1)  lbl4 = Label(window,text="Enter the third grade:")  lbl4.grid(row=3,column =0, sticky=W)  conOfent4 = StringVar()  ent4 = Entry(window,bd=3,textvariable=conOfent4)  ent4.grid(row=3, column=1)  btn1 = Button(window,text = "Compute the Average",command=findLargest)  btn1.grid(row=4, column = 1)  lbl5 = Label(window,text="Average:")  lbl5.grid(row=5,column=0,sticky=W)  conOfLargest = StringVar()  ent5 = Entry(window,bd=3,state="readonly",textvariable=conOfLargest)  ent5.grid(row=5,column=1)  mainloop()   |
|  |

| **Questions**  **1. Explain the advantages and disadvantages of using applying GUI in an Object-Oriented Program.**  The advantages is that the GUI is very user or consumer-friendly. It’s very straightforward and easy for the user to use. The users are presented with a Graphical Interface which explains what they have to input and they can just click on a button to do the function they are expected to do.  **2. Cite examples of actual systems that use GUI?**  Examples of systems using GUI are actually in abundance. Windows, Apple, Linux, the operating systems didn’t always have a GUI. They used to be text-based computers and people would have to perform text-based commands for the computer to do work. With the popularity and ease of access with GUI, it overshadowed TUI and became the one people most prefer.  **3. Why do you think GUI is used in an actual system that we use today?**  Because it is simple and easy to use. Unlike TUI, GUI uses visual elements instead of text to run a program. Inputting information becomes easy with GUI as it allows users to simply click, or drag and drop items instead of typing a specific command. It is also visually appealing. GUI’s visual elements can be arranged to form an organized display, unlike TUI’s block of text. |
| --- |
| **7. Conclusion:** |
| While GUI is easy to use, it is hard to program. |
| **8. Assessment Rubric:** |