# Lab 3 Algorithm

## Van Robbins 9/17/2024

### Problem Description

Design and create a python program that given three numbers outputs the numbers in order and says when they were inputed. It will have at least 3 user defined functions, at least one function will be called within another function, at least one function will take in at least two parameters, at least one function will return a value and at least one global variable is used and accessed inside a function.

### Algorithm

1. Comment Block File Header that details the name of the file, purpose, special requirements, variables, and version of the file
2. Create Variables So That All Values Are Stored In Variables Before They Are Used
   1. Use input function to create three variables defined by the person running the program
      1. num1 will be the first number inputed
      2. num2 will be the second number inputed
      3. num3 will be the third and final number inputed
   2. Initialize an empty array titled correctOrder that will later store the results of the program
3. Functions
   1. Create a function titled compare that will take in two integers that will be assigned the names first and second; It will find which of these numbers is larger
      1. Within the function use an if statement to see if first is smaller than second
         1. Since we know that the first number is smaller we can store this information by the way we send the variables compareAll to find the order them in relation to the third number
         2. call the compareAll function with the sending the first number then the second number, and call the global variable num3 to send the third number
         3. Our return will be the results of the compareAll function
      2. Otherwise the first number is greater than the second
         1. Since we know that the second number is smaller we can store this information by the way we send the variables to compareAll to find the order of them in relation to the third number
         2. call the compareAll function with the sending the second number then the first number, and call the global variable num3 to send the third number
         3. Our return will be the results of the compareAll function
   2. Create a function titled compareAll that will take in three integers that will be assigned small, big, unknown; It will compare the all the numbers given that we know whether the first or second number was larger than one another
      1. Unknown is always equal to the number we have not yet compared to anything ie num3
      2. Initialize a local variable mid to 0, it will be used later when we find the order
      3. Within the function use an if statement to see if big is less than unknown
         1. Since we know that big is larger than small and unknown is larger than big, big is the middle number
         2. Set mid to equal big first so we do not lose the value of big
         3. Set big to equal unknown
      4. If not use an if else statement to see if small is less than unknown
         1. We know that big is larger than small and since we used an if else statement big is also larger than unkown
         2. We also now know if this if else statement is called that small is less than unknown
         3. This means that small is in fact the smallest number, and big is in fact the biggest number
         4. This leaves only unknown to be the middle number so we set mid to equal unknown
      5. If neither of these if/if else statements are true than unknown must be the smallest number and small is the middle number
         1. Set mid to equal small
         2. Set small to equal unknown
      6. We can then return (small, mid, big) which will be an array with the numbers in order from least to largest of the three numbers
   3. Create a function titled order that will take in one integer and will be given the name findEntry; It will find when the numbers were entered
      1. Use an if statement to see if findEntry is equal to num1 the global variable of the first number entered
         1. if true return “1st number entered”
      2. Use an if else statement to see if findEntry is equal to num2 the global variable of the second number entered
         1. if true return “2nd number entered”
      3. Otherwise findEntry must be equal to num3 the third number entered
         1. return “3rd number entered”
4. We are ready to run our functions and print the results
   1. First ensuring that each number entered is unique
      1. Use an if statement that ensures no number is none of the entered numbers are equal to one another
      2. If they are print out “Try again! Enter three different integers”
   2. We now know our functions will run without error since they are three unique numbers
      1. We can set correctOrder to equal compare(num1, num2) this will call the compare function and then hence forth the compareAll function
         1. correctOrder will equal the return value of compareAll which is the small, mid, big
         2. small is at the index of 0 in the correctOrder array, mid is at the index of 1, and big is at the index of 2
      2. We are finally ready to print, we will use three print statements one for each number
         1. We will with the print(f) for all three print statements to allow for ease of inserting variables
         2. First print statement “The Smallest number is {correctOrder[0]} it was the {order(correctOrder[0])}”
            1. Calls the function order to find when the smallest number was entered at the beginning
         3. Second print statement “The Middle number is {correctOrder[1]} it was the {order(correctOrder[1])}”
            1. Calls the function order to find when the middle number was entered at the beginning
         4. Third print statement “The Largest number is {correctOrder[2]} it was the {order(correctOrder[2])}”
            1. Calls the function order to find when the largest number was entered at the beginning