Problem Set for 2/28/2024

Engineering 104 - Fundamentals of Engineering Computing

Formatting, Organization & Code Comments - Complete the following problems in `Python` and include as part of the submission of the appropriate assignment. Your assignment file should include a proper heading, comments and show clear organizational structure with each problem clearly printed, separated and with each result variable clearly displayed. All problems worked should have a formatted/structured print-out. Print a string denoting each problem, with the solution to the problem clearly printed as a formatted string below the denoted problem. Separate each problem using a blank line in both the code and the printed results. Code comments should be completed throughout the file on every line of code by default. If this assignment requires you to write and submit additional auxiliary script, or any other files in the submission, please append your initials capitalized to the end of the file name.

Python Lecture #18 Problems - Control Flow I (10 Points)

Problem 18.1 (4 Points) - The normal price of attendance for an event is $35 but costs $0 if you are less that 4 years old, $25 if you are less that 18, $5 if you are older that 65. Use an `if-elif-else` *to set the price of attendance based on an age variable. Print a statement that gives you the price of attendance based on an age you choose.*

Problem 18.2 (3 Points) - Define a function (recursion concept test) to calculate the factorial of any given integer using an `if` *statement. Run you function for the value of 5. A factorial is the given number multiplied by every integer less than that number to 1. For example 3-factorial(3!) and 4-factorial(4!):*

$$3! = 3 \times 2 \times 1 = 6 \qquad 4! = 4 \times 3 \times 2 \times 1 = 24$$

Problem 18.3 (3 Points) - Complete the `if-elif-elseExample.py` *script example in text book section 5.1. Once this is functioning re-order the conditional* `if` *statements so that they are in a different order and check that the function still gives results as expected. As a print statement state how your function was re-ordered.*