Carleton University
Department of Systems and Computer Engineering
ECOR 1041 - Computation and Programming

Lab 5 – Function Design Recipe (FDR)

Objectives

To apply *Practical Programming*'s function design recipe (FDR) to develop simple functions

Overview

In this lab, you will write code "all by yourself". The exercises are short and there are only two of them. Yet, you may find yourself lost for a while because you must first interpret the problem and then make the decisions on your own regarding how to proceed. Be patient with yourself. Before you throw your computer out the window and drop out of engineering – review previous examples –in the textbook, lectures, and previous labs – to help yourself. In teacher-talk, we are asking you to "apply" your knowledge.

Hopefully, you will come to appreciate the power of the FDR as a methodical way of solving programming problems, not simply a documentation chore.

All the code for all the following exercises should be placed in the same file using the same layout described previously (import, functions, main script).

For this lab (and all subsequent work in this course), function headers must contain a docstring and type annotations, as described in the lectures on the Function Design Recipe.

Begin by creating a new file within Wing 101. Save it as lab5.py

Exercise 1

The volume of a spherical segment can be calculated by the following function:

$$f : h, R_1, R_2 \rightarrow f(h, R_1, R_2); f(h, R_1, R_2) = (1/6)\pi h(3R_1^2 + 3R_2^2 + h^2)$$

where $\boldsymbol{h}$ is the height of the spherical segment, $\boldsymbol{R_1}$ is the radius of the base circle of the segment, and $\boldsymbol{R_2}$ is the radius of the top circle of the segment.

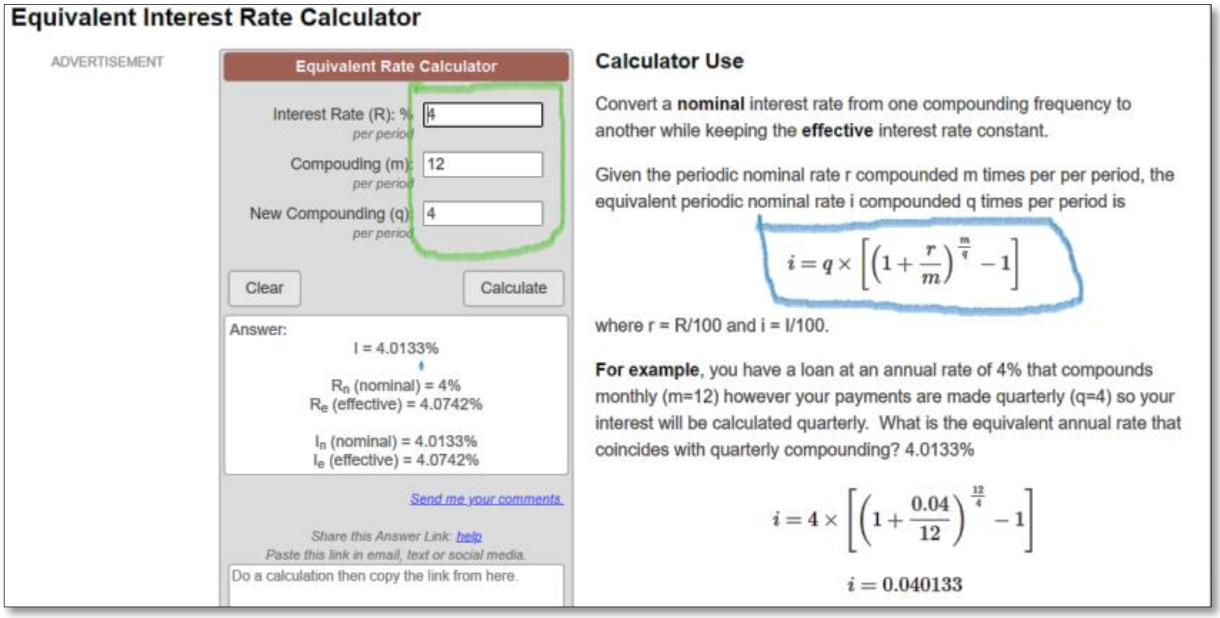
Using the FDR, design and implement a function to calculate the volume of a spherical segment

- The only limits that you have to follow are those made to help marking easier
  - The name of your function must be: `volume_spherical_segment`
  - Function takes three integers parameters which are height ($\boldsymbol{h}$), the radius of the base circle of the segment ($\boldsymbol{R_1}$), and the radius of the top circle of the segment ($\boldsymbol{R_2}$) (i.e. height($\boldsymbol{h}$) is 1, $\boldsymbol{R_1}$ is 2, and $\boldsymbol{R_2}$ is 4).
  - Function returns the volume of a spherical segment

Exercise 2

Here's a really hard exercise that is real-life experience. Don't worry – the exam questions are never at this level of difficulty. In the lab though, it is good to see what programming all is about. You do have enough knowledge about programming syntax at this stage to do this problem, but you must also be able to understand the problem first. Problem-solving -- That's the hard part!

Explore and study the **Equivalent Interest Rate Calculator**. Here's a few pointers to guide you:



- The blue line circles the key formula: Equivalent Interest Rate Calculator.
- The green line shows the menu box where you can select which values do you use within the formula to run calculations.
- Scroll to the bottom where there is an example.
- Play with the calculator by plugging in the example numbers to make sure you understand all the parameters.

Using the FDR, design and implement a function to calculate the **equivalent interest rate** using Equivalent Interest Rate Equation.

- The only limits that you have to follow are those made to help marking
  - The name of your function must be: `equivalent_interest_rate`
  - We will assume that **Equivalent Interest Rate** is a float and greater or equal than zero.
  - The units for all other parameters must match the online descriptions.
- Hint: Check your programming results against those computed by the website.
- Hint: If you are having trouble getting the math correct, try using easy numbers, like 1.

Wrap Up

- Make sure that you included your name and student number
- Check proper use constants (UPPER_CASE) and variables (lower_case) (There is a 10/100 deduction for misuse of UPPPER & lower case)

- Check the indents of the function bodies. (There is a 10/100 deduction for misuse of indentation)
- Check file organization: (1) imports, (2) **all** function definitions; (2) Main Script (There is a 10/100 deduction for not organizing the file according to the instructions)
- Confirm that your filename **lab5.py** matches exactly.
- Confirm that your .py script runs properly, otherwise the TA will also assign a zero.
- Submit the file on Brightspace.

You are required to keep a backup copy of (all) your work for the duration of the term.

Last edited: January 17, 2022