DSCI 503 – HW 01 Instructions

General Instructions

Create a new notebook named **`HW_01_YourLastName.ipynb`** and complete problems 1 – 8 described below.

Any set of instructions you see in this document with an orange bar to the left will indicate a place where you should create a markdown cell. For each new problem, create a markdown cell that indicates the title of that problem as a level 2 header.

Any set of instructions you see with a blue bar to the left will provide instructions for creating a single code cell.

Read the instructions for each problem carefully. Each problem is worth 6 points. An additional 2 points are allocated for formatting and following general instructions.

Assignment Header

Create a markdown cell with a level 1 header that reads: "DSCI 503 - Homework 01". Add your name below that as a level 3 header.

Problem 1: Arithmetic and Geometric Means

Consider four numbers denoted by, x_0, x_1, x_2, and x_3.

- The **arithmetic mean** of these numbers is given by the formula: $\frac{x_0 + x_1 + x_2 + x_3}{4}$
- The **geometric mean** of these numbers is given by the formula: $(x_0 \cdot x_1 \cdot x_2 \cdot x_3)^{1/4}$

In this problem, you will calculate the arithmetic and geometric mean of four numbers.

Perform the following steps in a single code cell.

1. Create variables named **`x0`**, **`x1`**, **`x2`**, and **`x3`** to store the values 11, 12, 17, and 19 (in that order).
2. Use the addition operator **+** to calculate the sum of the variables created in Part 1. Store the result in a variable named **`total`**. (Note that **`sum`** is reserved for a built-in function, and should not be used as a variable name.)
3. Use the multiplication operator * to find the product of the variables created in Part 1. Store the result in a variable named **`product`**.
4. Use the variable **`total`** to calculate the arithmetic mean of the variables created in Part 1. Store the results in a variable named **`arith_mean`**.
5. Use the variable **`product`** to calculate the geometric mean of the variables created in Part 1. Store the results in a variable named **`geom_mean`**.
6. Print the results with descriptive text output as shown below. Replace the **xxxx** symbols with the appropriate values. Round the arithmetic and geometric means to 2 decimal places **when displaying** these values. (Note that only the values displayed should be rounded, not the values stored within the variables.)
7. Match the formatting shown below exactly. Use additional spaces within your print statements to ensure that the numeric values displayed are left-aligned.

```
Sum:              xxxx
Product:          xxxx
Arithmetic Mean:  xxxx
Geometric Mean:   xxxx
```

Problem 2: Calculating Bill

In this problem, you will calculate the total cost of an order at a restaurant. Assume that the order consists of 2 side salads, 2 hamburgers, and 2 diet sodas. Assume also that the side salads cost $3.95 each, the hamburgers cost $8.95 each, and the sodas cost $2.50 each. Assume that the sales tax rate is 4.75% and the customers left a 15% tip.

Perform the following steps in a single code cell.

1. Use the ***** operator to calculate the total cost of the side salads. Store the result in a variable named **`ss_cost`**.
2. Use the ***** operator to calculate the total cost of the hamburgers. Store the result in a variable named **`hb_cost`**.
3. Use the ***** operator to calculate the total cost of the diet sodas. Store the result in a variable named **`ds_cost`**.
4. Add the variables created in Steps 1 – 3, storing the results in a variable named **`subtotal`**.
5. Calculate the tax charge by multiplying the subtotal by 0.0475. Round this result to 2 decimal places, and store the rounded value in a variable named **`tax`**.
6. Calculate the tip amount by multiplying the subtotal by 0.15. Round this result to 2 decimal places, and store the rounded value in a variable named **`tip`**.
7. Calculate the total charge by adding the subtotal, the tax charge, and the tip amount. Store the result in a variable named **`total_charge`**.
8. Print the results with descriptive text output as shown below. Replace the **xxxx** symbols with the appropriate values. Match the formatting shown below exactly, including spacing. The dashed lines should contain 22 dashes. There should be no spaces immediately after the dollar signs. The prices should be right-aligned with the dashed-lines.

```
2 Side Salads:   $xxxx
2 Hamburgers:   $xxxxx
2 Diet Sodas:    $xxxx
----------------------
Subtotal:       $xxxxx
Tax (4.75%):     $xxxx
Tip (15%):       $xxxx
----------------------
Total Charge:   $xxxxx
```

Problem 3: Volume of a Sphere

The volume of a sphere with radius r is given by the formula $V = \frac{4}{3} \cdot \pi \cdot r^3$. In this problem you will calculate the volume of spheres with various radii.

Write code to calculate the amount required to repay this loan by performing the following steps:

1. Create a variable named **`pi`** to store the value 3.14159.
2. Create three variables named **`r1`**, **`r2`**, and **`r3`** to store the values 4.6, 7.2 and 9.7. Each of these values is intended to represent the radius of a sphere.
3. Use the variables created in Parts 1 and 2 to calculate the volumes of the spheres with the given radii. Store the results in variables named **`v1`**, **`v2`**, and **`v3`**. Do not round the values stored in these variables.
4. Print the results using three message in format shown below, with the **xxxx** symbols replaced with the appropriate values. Round the displayed values for the volumes to 3 decimal places.

```
The volume of a sphere with radius xxxx is equal to xxxx.
```

Problem 4: Simple Interest

If a loan of P dollars collects simple interest at an annual rate of `i` and is repaid after `t` years, then the amount repaid, denoted by A, is given by the following formula: $A = P \cdot (1 + i \cdot t)$

Suppose that a loan of $210 collects simple interest at an annual rate of 0.09 and is repaid after 10 months (i.e. 10/12 of a year).

Write code to calculate the amount required to repay this loan by performing the following steps:

1. Create variables **`P`**, **`i`**, and **`t`** to store the relevant values for this loan.
2. Use the variables created in Step 1 above to calculate the repayment amount, storing the result in a variable **`A`**.
3. Print the result in a message using the format shown below, with the **xxxx** symbols replaced with the appropriate value.
4. Pay careful attention to the formatting in your output. In particular, there should be a single space between the colon and dollar sign, and there should be no spaces between the dollar sign and the amount.

```
Amount repaid: $xxxx
```

Problem 5: Compound Interest

If loan of P dollars collects compound interest at an annual effective rate of `i` and is repaid after `t` years, then the amount repaid, denoted by A, is given by the formula: $A = P \cdot (1 + i)^t$

Suppose that a loan of $400 collects compound interest at an annual rate of 0.05 and is repaid after 3 years.

Write code to calculate the amount required to repay this loan by performing the following steps:

1. Create variables **`P`**, **`i`**, and **`t`** to store the relevant values for this loan.
2. Use the variables created in Step 1 above to calculate the repayment amount, storing the result in a variable **`A`**. Round your result to 2 decimal places.
3. Print the result in a message using the format shown below, with the xxxx symbols replaced with the appropriate value.
4. Pay careful attention to the formatting in your output. In particular, there should be a single space between the colon and dollar sign, and there should be no spaces between the dollar sign and the amount.

```
Amount repaid: $xxxx
```

Problem 6: Annuity

An annuity is a scheduled sequence of payments or deposits. Assume that deposits in the amount of PMT are made at the end of each year. If the account receiving the deposits collects interest at an annual effective interest rate of `i`, then the balance of the account after n years is given by: $A = PMT \cdot \frac{(1+i)^n - 1}{i}$

Suppose a couple establishes a college fund for their newborn child by depositing $2000 into an account collecting interest at an annual effective rate of 0.04 for 18 years.

Write code to calculate the balance of this account at the end of 18 years. Following these steps below.

1. Create variables **`PMT`**, **`i`**, and **`n`** to store the relevant values for this loan.
2. Use the variables created in Step 1 above to calculate the repayment amount, storing the result in a variable **`A`**. Round your result to 2 decimal places.
3. Print the result in a message using the following format shown below, with the `xxxx` symbols replaced with the appropriate value.
4. Pay careful attention to the formatting in your output. In particular, there should be a single space between the colon and dollar sign, and there should be no spaces between the dollar sign and the amount.

```
Balance after 18 years: $xxxx
```

Problem 7: Probability

Assume that A and B refer to two specific events. Let `P(A)` denote the probability of Event A occurring, and let `P(B)` denote the probability of event B occurring. Let `P(A or B)` denote the probability of at least one of the two events occurring, and let `P(A and B)` denote the probability of both events occurring.

Two events are said to be independent if the knowledge that one of the events has occurred has no effect on the probability that the other event has occurred.

Probability theory tells us the following:

- `P(A or B)` and `P(A and B)` are related by the formula: `P(A or B) = P(A) + P(B) - P(A and B)`.
- If A and B are independent, then `P(A and B) = P(A) * P(B)`.

Assume that Allen and Beth are meeting for dinner. The probability that Allen is late is equal to 0.6. The probability that Beth is late is equal to 0.3. Whether or not one person is late is independent of whether or not the other person is late.

Write code to calculate the probability that at least one person is late to dinner, and the probability that both people are late to dinner. You can be accomplished as follows:

1. Create a variable named **`prob_A`** to store the probability that Allen is late. Create a variable named **`prob_B`** to store the probability that Beth is late.
2. Use the formulas provided above to calculate the probability that both people are late, as well as the probability that at least one of the two people is late. Store the results in variables named **`prob_A_and_B`** and **`prob_A_or_B`**. Use the variables you created in Step 1 for your calculations.
3. Print the following messages, with the xxxx symbols replaced with the appropriate values. Try to match the format and spacing in these messages exactly. In particular, make sure that the numerical values display are aligned on the left.

```
Probability that both people are late:        xxxx
Probability that at least one person is late: xxxx
```

Problem 8: Constructing an Output Message

In this problem, you will be asked to define a single string variable **`msg`** to store a multiline message.

Create a code cell to perform the following steps:

1. Create a variable named **`first`** to store a string containing your first name.
2. Create a variable named **`last`** to store a string containing your last name.
3. Create a variable named **`sid`** to store an integer containing your student ID number. Ignore any leading zeros in your ID number.
4. Use the variables created above to define a variable named **`msg`**. This variable should contain a string, that when printed, will display as shown below with the xxxx symbols replaced with the appropriate values. This will involve the use of the escape character **`\n`**.
5. Print **`msg`**.

Match the formatting shown below exactly, including the spacing. The message should be printed on three different lines, but the **`print()`** function should be used only once. The values used to replace the xxxx symbols should be aligned with each other on the left. The output does NOT need to be indented.

```
First Name: xxxx
Last Name:  xxxx
Student ID: xxxx
```

Submission Instructions

When you are done, click **Kernel > Restart and Run All**. If any cell produces an error, then manually run every cell after that one, in order. Save your notebook, and then export the notebook as an HTML file. Upload the HTML file to Canvas and upload the IPYNB file to CoCalc, placing it in the **Homework/HW 01** folder.