CSE 1321L: Programming and Problem Solving I Lab

Assignment 1 Fall 2025

Module 1

What students will learn

- Problem Solving.
- Terminology.
- Basic Program Structure.
- Input and Output with the user.
- Basic calculations and calculations that require an intermediate solution.

Content

Overview:

For most of you, this will be the first time you have done any programming, which is exciting! The write-up of this first assignment will be a little longer than the others because we want you to understand how things are going to roll out the rest of the semester. Advice: Start early (certainly not the day the assignment is due), practice, and ask a lot of questions.

Unless calculations are trivial, you will almost always want to use an intermediate variable – where you store part of the solution. For example, you might remember the equation for gravity as:

$$F = G\frac{m_1 \times m_2}{R^2}$$

How would you write this as code? You might solve it in parts instead of one shot because it makes it easier to check. It would look like:

```
Temp1 = m1 * m2
Temp2 = r * r
Temp3 = temp1 / temp2
F1 = g * temp3
```

This is assuming the variables ***m1***, ***m2***, and ***g*** declared previously in the code.

Final note: *Do not cheat*

If your temptation is to look online, don't. Come see us instead and ask questions – we are here to help. Remember, you are going to have to write codes in your future job interviews, so learn it now to secure a high-paying job later.

Assignment1A: Customer Receipt

You've been asked to help develop a simple tool for a local bakery that prints **receipt messages** when a customer places an order. Your job is to write a small program that generates a short summary message based on the customer's input.

For this assignment:

- Prompt the user to enter their **name**.
- Prompt the user to enter the **item they purchased**.
- Prompt the user to enter the **quantity** of the item.
- Use **string concatenation** to build a sentence like this:

Thank you, [name]! You ordered [quantity] [item](s)

Hint:

- You are allowed to use F-strings

Below are two example runs. The user input is shown in **red and bold**.

Sample Output #1:

```
[Customer Receipt Generator]
Enter your name: Sarah
Enter the item you purchased: muffin
Enter the quantity: 5
Thank you, Sarah! You ordered 5 muffin(s).
```

Assignment1B: Electrical Energy Consumption Calculator

For this assignment, we will create a Python program that will calculate the **electrical energy consumption** of a device over time using the formula for energy:

$$E = P * t$$

Where:

- E: Energy consumed (in watt-hours, Wh)
- P: Power (in watts)
- t: Time (in hours)

However, to make this assignment more interesting, we won't ask the user for power directly. Instead, we will calculate it using this modified form of the power formula derived from Ohm's Law:

$$P = \frac{V^2}{R}$$

Where:

- V: Voltage (in volts)
- R: Resistance (in ohms)

By combining the two formulas, the final equation to calculate energy becomes:

$$E = \frac{V^2 * t}{R}$$

You will write a Python program that calculates **energy consumption in kilowatt-hours (kWh)** using the formula above.

For this assignment:

- Prompt the user to input the following values:
- Voltage (in volts)
- Resistance (in ohms)
- Time the device was used (in hours)
- Make sure all values are read as **floats**.
- Create a constant variable named CONVERSION_FACTOR = 1000 (since 1 kilowatt = 1000 watts) to convert watts to kilowatts.
- Use the formula:

$$E = \frac{V^2 * t}{R * CONVERSION\_FACTOR}$$

- Output the **energy consumed in kWh**, rounded to **2 decimal places**.

Example runs are shown below. The user input is shown in **red and bold**.

Sample Output #1:

```
[Electrical Energy Consumption Calculator]
Enter the voltage (in volts): 120
Enter the resistance (in ohms): 60
Enter the time the device was used (in hours): 5
The device consumed 1.2 kWh of energy.
```

Sample Output #2:

```
[Electrical Energy Consumption Calculator]
Enter the voltage (in volts): 110
Enter the resistance (in ohms): 30
Enter the time the device was used (in hours): 10
The device consumed 4.03 kWh of energy.
```

Assignment1C: Weight Conversion- Kilograms to Stones and Pounds

For this assignment, you will create a Python program that converts a weight given in **kilograms** into **stones and pounds**, a common measurement system used in the UK.

These are the conversions you should use in your solution:

$$1\ kilogram\ (kg) = 2.20462\ pounds\ (lb)$$

$$1\ stone = 14\ pounds\ (lb)$$

For this assignment:

- The program should **prompt and read** a weight value **in kilograms** from the user.
- Make sure to read the input as a **float**.
- Use the provided conversion rates to:
  - Convert the input kilograms to **pounds**
  - Then convert the pounds into **stones** and the **remaining pounds**
- You must perform the conversions **manually** using math operations such as:
  - int() for truncating decimal parts
  - // for floor division
  - % or subtraction to calculate remainders
- Round the **remaining pounds** to **2 decimal places**.

Notes:

- Do **not** simply print the total pounds and stones separately.
- You are expected to **break the total pounds** into full **stones** and the **remaining pounds**, using math techniques.

Example runs are shown below. The user input is shown in **red and bold**.

Sample Output #1:

```
[Weight Conversion- Kilograms to Stones and Pounds]
Enter weight in kilograms: 120
120.0 kilograms is approximately 18 stones and 12.55 pounds.
```

Sample Output 2:

```
[Weight Conversion- Kilograms to Stones and Pounds]
Enter weight in kilograms: 70
70.0 kilograms is approximately 11 stones and 0.32 pounds.
```

Submission Instructions:

- Programs must follow the output format provided. This includes each blank line, colons (:), and other symbols.
- Programs must be working correctly.
- Programs must be written in Python.
- Programs must be submitted with the correct **.py** format.
- Programs must be saved in files with the correct file name:
  - Assignment1A.py
  - Assignment1B.py
  - Assignment1C.py
- Programs (source code files) must be uploaded to Gradescope by the due date.