**Name:** Eshan Uniyal **UID:** 205-172-354

# Obstacles

1. **Syntax** – As a Python programmer, I kept making minor syntax mistakes (missing semicolons, no brackets in *if* conditions, et cetera). I started getting the hang of it after a bit of debugging, though.
2. **Testing** – Since the program has many moving parts (and good scope for edge cases with the variable rate system), making sure it works in all cases was tedious
3. **Warning C26451** – Visual Studio was underlining (*meter* – *tier\_cutoff*) in line 52 with a warning C26451, saying **“Arithmetic overflow: Using operator '-' on a 4 byte value and then casting the result to a 8 byte value. Cast the value to the wider type before calling operator '-' to avoid overflow (io.2).” I cast *meter* and *tier\_cutoff* as doubles instead of integers, and that seemed to fix the issue.  
   Conversely, I had to cast *initial\_reading* and *final\_reading* as doubles instead of integers, because the same warning was showing up in *meter*’s assignment statement**

# Test Data

**Testing error catching**

1. (-1, 100, Eshan, 7) – correct output: “The initial meter reading must be nonnegative.”
2. (100, 99, Eshan, 7) – correct output: “The final meter reading must be at least as large as the initial reading.”
3. (100, 100, Eshan, 7) – correct output in boundary case (100, 100): “The bill for Eshan is $0.00”
4. (100, 101, <empty string>, 7) – correct output: “You must enter a customer name.”
5. (100, 101, Eshan, 0) – correct output: “The month number must be in the range 1 through 12.”
6. (100, 101, Eshan, 1) – correct output in boundary case (1): “The bill for Eshan is $2.71”
7. (100, 101, Eshan, 12) – correct output in boundary case (12): “The bill for Eshan is $2.71”
8. (100, 101, Eshan, 13) – correct output: “The month number must be in the range 1 through 12.”

**Testing String Input**

1. (100, 100, Eshan Uniyal, 7) – correct output: The bill for Eshan Uniyal is $0.00”

**Testing order in which errors are caught**

1. (-1, -2, <empty string>, 13) – correct output: “The initial meter reading must be nonnegative.”
2. (1, -2, <empty string>, 13) – correct output: “The final meter reading must be at least as large as the initial reading.”
3. (1, 2, <empty string>, 13) – correct output: “You must enter a customer name.”
4. (1, 2, Eshan, 13) –correct output: “The month number must be in the range 1 through 12.

**Testing evaluation of total amount**

1. (100, 200, Eshan, 3) – correct output: “The bill for Eshan is $283.36”
2. (100, 200, Eshan, 4) – correct output (boundary case): “The bill for Eshan is $309.76”
3. (100, 200, Eshan, 10) – correct output (boundary case): “The bill for Eshan is $309.76”
4. (100, 200, Eshan, 11) – correct output: ”The bill for Eshan is $282.36”