CS0011 Introduction to Computing for Scientists
Project 1
Lab 3

Lab 3

Part 1: Utilizing functions

1. First, you should move some of your code from Lab 2 into a function named `print_header`. This function should take 1 keyword argument named `csv` with a default value of `True`. It should print out a CSV header if `csv` is `True`, e.g.:

   ```
   Fuel,Velocity,Duration,Fuel cost,Losses,Value lost
   ```

   It should print out a human-readable header if `csv` is `False`, e.g.:

   ```
         Fuel    Velocity    Duration       Fuel cost            Losses       Value lost
   ```

2. Next, you should move more of your code from Lab 2 into a function named `print_row`. This function should take 1 positional argument and 1 keyword argument. The positional argument should be named `fuel` and should represent the amount of fuel to be analyzed and displayed on the current row. The keyword argument should be named `csv` with a default value of `True`.

   `print_row` should print out a CSV row if `csv` is `True`, e.g.:

   ```
   10000,953.1017980432493,910.768983268842,10000000,109292277.99226104,119292277.99226104
   ```

   It should print out a human-readable header if `CSV` is `False`, e.g.:

   ```
     10,000 kg  953.10 m/s  910.8 days  10,000,000 USD  109,292,278 USD  119,292,278 USD
   ```

3. Finaly, move the rest of your code from Lab 2 into a function named `main` and modify it to call `print_header` and `print_row`. `main` should not take any arguments. Call `main` at the end of your program.

Part 2: Input validation

Now add an additional function named `get_choice`. This function should not take any arguments. When your `main` function asks the user for their preferred output format (human-readable or CSV) it should call `get_choice`. If the user does not enter a valid choice, they should be continually prompted until a valid choice is entered. Your `get_choice` function should implement this functionality using recursion.

An example run of the desired progam would be:

```
How would you like to format the output?
1: CSV
2: Human-readable
Enter 1 or 2:  5
Invalid choice! Please try again
Enter 1 or 2:  -2
Invalid choice! Please try again
Enter 1 or 2:  no
Invalid choice! Please try again
Enter 1 or 2:  1
Fuel,Velocity,Duration,Fuel cost,Losses,Value lost
10000,953.1017980432493,910.768983268842,10000000,109292277.99226104,119292277.99226104
```

Part 3: Trying multiple amounts of fuel

Modify your program to try multiple values for the amount of fuel, rather than asking the user for the amount. Try values of fuel from very little (cheap, but risks greater damages) to the maximum (fast, but expensive).

Some initial questions to consider:

- What is the smallest amount of fuel you will try?
- What is the largest amount of fuel you will try?
- How many different amounts of fuel will you try in this interval? (You will need to try at least 10)

You can set these values as magic numbers within your program to help define the base and recursive cases for your recursion. You could have your program check 10 fuel levels from 10000 (inclusive) to 20000 (exclusive) (i.e., 10000, 11000, 12000, 13000, 14000, 15000, 16000, 17000, 18000, and 19000), by setting `MIN_FUEL = 10000`, `MAX_FUEL = 20000`, `CHANGE_FUEL = 1000`. In this case, your program would start at 10000 fuel, and keep adding 1000 more fuel on until it reached (exclusive) 20000.

Once you have come up with the set of fuel values to try, your program should print all the same values you printed in Lab 2 (velocity, duration, fuel cost, losses from damage, and total value lost) for each fuel value. If you were checking only fuel values of `10000` and `11000`, a run of the expected program would be:

```
How would you like to format the output?
1: CSV
2: Human-readable
Enter 1 or 2:  3
Invalid choice! Please try again
Enter 1 or 2:  2
       Fuel     Velocity    Duration       Fuel cost           Losses       Value lost
  10,000 kg   953.10 m/s  910.8 days  10,000,000 USD  109,292,278 USD  119,292,278 USD
  11,000 kg 1,043.60 m/s  831.8 days  11,000,000 USD   99,814,729 USD  110,814,729 USD
```

Try several different intervals until you have one that you think well-demonstrates what the best choice would be (e.g., lowest value lost).

Once you are happy with the interval of fuel values you are checking, show your work to the lab instructor and submit your finished project solution for grading.