CDS-230-001 - FALL 2023

PROBLEM SET 2

9/7/23

Instructions: Use the PS template provided in class to enter your name and your answers. When you are done run your script and make sure it executes without syntax errors. Note: You may need the **math** module to access math functions. Make sure you import it once at the top of your script.

Important: if you use concepts or techniques that have not been *yet* been covered in class, then I may ask you to explain the technique to me.

Exercise 2.1: Loops (20 points)

In this exercise you will write Python code **snippets** that use **loops and conditionals**. Note: Do not use any type of Python function.

Write code to

a) print the first 10 odd numbers (1 pt)

b) print the first 10 even numbers in reverse (1 pt)

c) print the first 10 non-zero multiples of 13 (5 pts)

d) display the sum of the digits of a number. Test with 5, 123, and 980 (5 pts)

e) display the product of the digits of a number. Test with 5, 123, and 980 (5 pts)

f) display all the numbers between 100 and 500 which are divisible by 11 but not by 2 (3 pts)

Note 1: for parts (d) and (e) you can avoid repeating the tests by wrapping your solution with an additional for-loop that selects the test cases as follows:

```
for num in [5, 123, 980]:
    # your solution here
```

Note 2: for some print statements you may want to suppress printing a new line after every call to print() by adding the argument , end=", " to print(), e.g.

```
print(i, end=", ")
```

Exercise 2.2: Random numbers (12 points)

In this exercise you will create a list of random integers using the random module as follows:

```
# import Python's import module
import random
random.seed(123)
# Use randint() to generate a random integer between 1 and 10
rand_int = random.randint(1,10)
```

- Use the above code to generate 1000 random integers and store them in a **list** called *rand_list*. Hint: use a **for loop**. (4 pts.)
- Given your newly created list, use **slicing** to
  - print the first 10 numbers (1 pt.)
  - convert the list to a tuple, call it *rand_tuple*. How many elements are there in *rand_tuple*? Print the number. (3 pts.)
  - Use a **for loop** to count and print the number of odd *numbers* in the tuple. (4 pts)

Exercise 2.3: Perfect numbers (6 points) A perfect number is a positive integer which is equal to the sum of its divisors. For example the divisors of 6 are 1,2,3 and the sum of the divisors is also 6, so 6 is a perfect number.

Write code that determines and prints all the perfect numbers between 1 and 1000.

Hint: use a double-loop.

Note: do **not** solve this problem using Python functions, whether user-defined or intrinsic.

Exercise 2.4: Projectile trajectory (12 points)

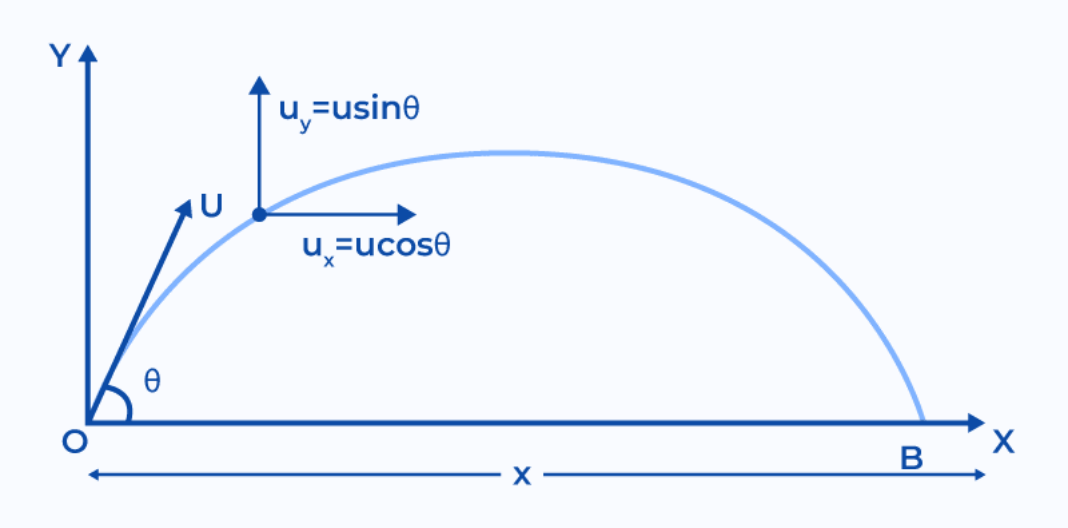
In math the equation of a parabola is $f(x) = ax^2 + bx + c$. Similarly, in basic Physics one can show that the *trajectory*, i.e. the path through space, of an *ideal* projectile is also a parabola. Such an equation can be derived by combining the linear and vertical motion equations (by eliminating t) to get

$$f(x) = xtan(\theta) + \frac{gx^2}{2u^2cos^2(\theta)} + y_0 \tag{1}$$

where θ is the projectile launch angle, u is the initial speed, y_0 is the initial height and g is the acceleration due to gravity.

In this problem your task is to compute $f(x) = y$ for $x \in [0, 10]$ (i.e. 11 values) where you will use the following model parameters: $g = -9.81\,\mathrm{m\,s^{-2}}$, $u = 15\,\mathrm{m/s}$, $y_0 = 0\,\mathrm{m}$, $\theta = 60^\circ$

In your solution



a) Define the variables as given. Note: trig functions in Python require angles measured in radians (3 pts)

b) Evaluate f(x) for the required x values. Note: Do not use functions.

- You will need a loop to compute the 11 values of f(x) for each x. (3 pts)
- Use a *list* to store the $f(x)$ values. (3 pts)

c) Print, using 2 decimal values of precision, the values of x and f(x) in a "table" as follows (3 pts):

```
x0, f(0)
...
x10, f(10)
```