MATH 210 Assignment 4

NumPy and Matplotlib

INSTRUCTIONS

- ◇ Create a new Jupyter notebook, set the kernel to Python 3, present your solutions in the notebook and clearly label the solutions
- ◇ Your solutions should include clear explanations (including proper use of markdown language and LaTeX) and your functions should include comments and documentaiton strings
- ◇ There are 25 total points: each question is worth 4 points and 1 point will be awarded for the overall presentation of your notebook
- ◇ Each question is graded out of 4 points according to the rubric:
  - 4 - Solution is correct and written clearly (including comments if needed)
  - 3 - Solution is correct but is unclear
  - 2 - Solution is partly correct
  - 1 - Solution needs improvement
- ◇ Submit the `.ipynb` file to Connect by **11pm Tuesday February 9**
- ◇ You may work on these problems with others but you must write your solutions on your own

QUESTIONS

1. (a) Define a function called `parametric_curve` which takes an input `k` and plots the parametric curve:
   $$x = \cos(t) - \cos(kt) \quad \text{and} \quad y = \sin(t) - \sin(kt)$$
   for $t \in [0, 2\pi]$. (Use the matplotlib.pyplot command `plt.axis('equal')` to display the plot with equal units along each axis.)

   (b) Call the function `parametric_curve` from part (a) for `k` equal to 2, 5, 10, and 50.

2. Define a function called `plot_fun` which takes a vectorized function `f`, a Python list `interval` of length 2 and a string `title` (in that order: `plot_fun(f,interval,title)`) and plots the function `f` over the interval given by `interval` and with the title given by `title`. For example, `plot_fun(np.cos,[0,2*np.pi],'Graph of Cosine')` will plot $y = \cos(x)$ for $x \in [0, 2\pi]$.

3. Define a function called `dice_game` which takes two positive integers `num_dice` and `winner` (in that order: `dice_game(num_dice,winner)`) and performs the following tasks:
   - ◇ Simulate a roll of `num_dice` many dice. This means that if `num_dice` is 3, a roll is the sum of 3 random numbers chosen from {1,2,3,4,5,6}.

- ⋄ If the winning number `winner` is rolled, then print a winning message and end the game (by calling `return None`). For example, if `winner` equals 5, then print "You rolled a 5! You win!"
- ⋄ If the winning number is not rolled, then print a losing message and simulate another roll of the dice. For example, if 3 is rolled and 5 is the winner, print "You rolled a 3. Try again!"
- ⋄ The function should continue rolling the dice until there is a winning roll.
- ⋄ If the function is called such that `winner` is an impossible outcome (in other words, if `winner` < `num_dice` or `winner` > `6*num_dice`), then the function should print the message "Impossible game." and then end the game.

4. (a) Write LaTeX code in a markdown cell to display the following function:

$$f_N(x) = \frac{4}{\pi} \sum_{n=0}^{N} \frac{\sin(2\pi(2n+1)x)}{(2n+1)}$$

(b) Write a function called `square_wave` which takes a positive integer `N` and a Python list `interval` of length 2 (in that order: `square_wave(N,interval)`) and plots the function $f_N(x)$ (defined above in part (a)) over the interval given by the list `interval`. For example, `square_wave(2,[0,3])` plots the function

$$\frac{4}{\pi} \left(\sin(2\pi x) + \frac{1}{3} \sin(6\pi x) + \frac{1}{5} \sin(10\pi x) \right)$$

over the interval $[0, 3]$. (Note that your function should plot the function smoothly over *any* interval.)

5. (a) Write LaTeX code in a markdown cell to display the following:

The Taylor series of $\ln(x)$ centered at $x = 1$ is given by:

$$\ln(x) = \sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{n} (x-1)^n \ , \quad x \in (0, 2) \, .$$

(b) Write a function called `ln_taylor` which takes a float `x` in the closed interval $(0, 2)$ and a positive integer `N` (in that order: `ln_taylor(x,N)`) and returns the Nth partial sum of the Taylor series of ln (defined above in part(a)) evaluated at x:

$$\sum_{n=1}^{N} \frac{(-1)^{n+1}}{n} (x-1)^n$$

If the input value `x` is outside the interval of convergence $(0, 2)$ of the Taylor series, then your function should print a warning message and return the Python value `None`.

6. (a) Write LaTeX code in a markdown cell to display the Prime Number Theorem:

Let $\pi(x)$ be the prime-counting function: for any real number x, the value $\pi(x)$ is the number of primes p less than or equal to x. Then

$$\lim_{x \to \infty} \frac{\pi(x)}{x / \ln(x)} = 1$$

(b) Write a function called `prime_number_theorem` which takes a number N and plots both functions $y = \pi(x)$ and $y = x/\ln(x)$ over the interval $[2, N]$. The plot should include the title "Prime Number Theorem".