

NE336 Assignment 1 Due Sep 20th

Instructions

- All programming problems should be completed as separate .py files. The first line in each file should contain a comment with your complete name and student ID.
- Your py file should include : comments and test cases mentioned in the question.
- An online dropbox will be available for your submissions on learn. This dropbox will only be available until the 11:59 pm of the due date. Make sure your submission is received by this time since *no exceptions will be made*.

Questions

Question 1

Objective of the section is to understand how computers represent numbers and some consequences of this.

On February 25, 1991, a Patriot missile battery operating in Saudi Arabia failed to identify an incoming scud missile, resulting in the deaths of 28 American soldiers and injuries to around 100 others. This was due to a simple numerical error: the internal clock in the radar system counted in 1/10ths of a second using a 24-bit binary number representation.

- a) What is the error (expressed as a decimal number in seconds) associated with each “tick” of the system’s clock? Assume the first bit indicates units (2^0), the second bit halves (2^{-1}), etc. For example, a four-bit number might be 0.110 (decimal 0.75).
- b) What is the cumulative time error after 100 hrs (the time the missile battery had been operating)? How far will a Scud missile travel in this time (assume it is traveling at Mach 2)?
- c) What type of error is this? Suggest a simply mechanism by which the original programmers could have avoided it.

You **do not need to include a python file** for this question but you may find it useful to write a simple script for the required calculation.

Question 2

In a file called `summation.py`, write a function to evaluate the following sum (note: defined for $|x| < 1$ only).

$$f(x) = \sum_{n=0}^{\infty} x^{2n+1}$$

Use absolute approximate relative error and a tolerance of 10^{-5} . Your function should accept x as an input and check its validity:

- make sure x is an acceptable type.
- test whether $|x| < 1$ or otherwise print an error statement and return `None`.

Please show tests in your script to show what your function returns in these cases.

Once you have a working function, modify it so that it accepts the tolerance as a second optional input (defaulting to 10^{-5} otherwise).

Note: It can be shown that the summation evaluates to $\frac{x}{1-x^2}$. You can make use of this result in checking your algorithm, but you must not use it to test convergence.