
Remember to indent the bodies of your IF-THEN statements and looping structures.

- 1) (3 pts) The geometric average of the n positive values

$$x_1, x_2, \dots, x_n$$

is defined as

$$x_G = \sqrt[n]{x_1 \cdot x_2 \cdots x_n}.$$

Modify the program from Section 2.8 of the looping notes to compute the geometric average of a set of user-input positive values. Test your program using the set of numbers

4, 2, 8, 9, 11, 6, 13

Note that this program is almost identical to Question 2 of Homework 13 (you should only have to change three lines).

- 2) (7 pts) This problem is an example of a Monte-Carlo method. In a Monte-Carlo method, random numbers are used to compute solutions to problems that are too complex for a traditional mathematical analysis (though for this problem, we know what the answer should be).

Write a program that simulates the flipping of a coin. Here are some guidelines:

- a) The user should input the number of coin flips, n .
- b) To flip the coin, use the `rand` function.

```
flip = rand;
```

will generate a uniformly-distributed random value in the interval $(0, 1]$.

- c) Use two counters to store the number of heads and number of tails.
- d) Assume a fair coin; this means that if the flip value is ≤ 0.5 , then the result is a head, otherwise it is a tails.
- e) Print out the total percentage of heads and tails obtained.
- f) Run your code for $n = 500$ several times. Are the results what you expect?
- g) Run your code for $n = 10000$ several times. How do these results compare with the previous value of n ?
- h) Run your code for $n = 1000000$ several times. How do these results compare with the previous value of n ?

- 3) (3 pts) Download the file `xdat.dat` from the web page and copy this into your working MATLAB directory. This file contains a long vector. Load the file into MATLAB using the syntax

```
load xdat.dat
```

This will create a variable named `xdat` in your MATLAB script.

Write a script that will locate the position of the first element of `xdat` that is greater than 100 and the position of the first element that is less than -100. It is easy to do this using two separate loops, but you can get 1 point extra credit if you can do it using only one loop.