Fundamentals and basic coding environment: Chapman 2

1. a) You are assigned to develop a circuit analysis package for high school students to get familiar with calculating voltage and current at any point in a virtual circuit network.

In the lecture, we had a program plan for the sub-task of calculating equivalent resistance given 3 resistors connected in series, with their mean resistance values, and their Gaussian noise variances.

Your assignment is to make a program plan for the sub-task of calculating equivalent resistance given 3 resistors connected in parallel, with their mean resistance values and their Gaussian noise variances.

b) Please implement what you had planned into a .f95 fortran code. Please compile and test your code, and submit your code by email to ytian@olemiss.edu by Fri 9/12/08 5pm. Please note that your code needs to follow the coding style including required comment lines, correct naming of program, variable and constance, without syntax errors or run-time errors for full credits.

Hint: You need to inject the following statements into your program for random number generation

\begin{verbatim}
! Declare randomNumber
real :: randomNumber
! Initialize the pseudorandom number generator
call random_seed()
! Obtain next random number
call random_number(randomNumber)
\end{verbatim}
Solution 1.a) Program planning for resistorsInParallel.f95

- Problem statement
Calculate the equivalent resistance $R_{eq}$ of three resistors $R_1$, $R_2$, and $R_3$ connected in parallel. Use Gaussian noise for each resistor.

- Input data
  Resistance for $R_1$, noise variance $\sigma_1$
  Resistance for $R_2$, noise variance $\sigma_2$
  Resistance for $R_3$, noise variance $\sigma_3$

- Output data
  The equivalent resistance $R_{eq}$

- Algorithm
  - Prompt the user for $R_1$, $\sigma_1$
  - Prompt the user for $R_2$, $\sigma_2$
  - Prompt the user for $R_3$, $\sigma_3$
  - Select a random number to simulate resistance noise $\Delta$
  - Calculate $R_{eq}$ as follows:
    \[
    R_{1\text{ noisy}} = R_1 + \exp\left(-\frac{\Delta^2}{\sigma_1^2}\right)
    \]
    \[
    R_{2\text{ noisy}} = R_2 + \exp\left(-\frac{\Delta^2}{\sigma_2^2}\right)
    \]
    \[
    R_{3\text{ noisy}} = R_3 + \exp\left(-\frac{\Delta^2}{\sigma_3^2}\right)
    \]
    \[
    R_{eq} = \frac{1}{\frac{1}{R_{1\text{ noisy}}} + \frac{1}{R_{2\text{ noisy}}} + \frac{1}{R_{3\text{ noisy}}}}
    \]
  - Print out user input data and the result

- Testing plan
  $R_1$ $\sigma_1$ $R_2$ $\sigma_2$ $R_3$ $\sigma_3$ $\Delta$
  10. 1. 10. 1. 10. 1. 1.

$R_{eq} = 3.455960$
Calculate equivalent resistance for resistors in parallel
Each resistance is simulated with additive Gaussian noise
User input of the resistance values and the Gaussian variance values

program resistorsInParallel
  real, parameter :: E = 2.718281828

  real :: randomNoise
  real :: resistance1, resistance2, resistance3
  real :: sigma1, sigma2, sigma3
  real :: r1n, r2n, r3n
  real :: equivalentResistance

  ! prompt and receive user input
  write (*,*) "Please enter parameters for resistor #1 (resistance, noise variance)"
  read (*,*) resistance1, sigma1
  write (*,*) "Please enter parameters for resistor #2 (resistance, noise variance)"
  read (*,*) resistance2, sigma2
  write (*,*) "Please enter parameters for resistor #3 (resistance, noise variance)"
  read (*,*) resistance3, sigma3

  ! generate a random number to simulate noise
  call random_seed()
  call random_number(randomNoise)
  write (*,*) randomNoise

  ! generate noisy resistance values
  r1n = resistance1 + E ** (-randomNoise**2 / sigma1**2)
  r2n = resistance2 + E ** (-randomNoise**2 / sigma2**2)
  r3n = resistance3 + E ** (-randomNoise**2 / sigma3**2)

  equivalentResistance = r1n + r2n + r3n

  write (*,*) "The equivalent resistance for resistors"
  write (*,*) "#1: ", r1n, ", #2: ", r2n, ", #3: ", r3n
  write (*,*) " connected in parallel is ", equivalentResistance
end