

FUNCTIONS AND PASS BY DATA, ADDRESS AND REFERENCE

DT008 / 3 Software Design 2 Laboratory Exercise



October 16, 2019

Talha Tallat - D18124645

TU DUBLIN

# Program Code

/\*

Filename: FUNCTION AND PASS BY DATA, ADDRESS AND REFRENCE

Author: Talha Tallat (D18124645)

Date Started: 02/10/2019

Description:

\* This program allows the user to enter3 resistor values and calculate and displays the series and parallel resistance.

\* The program includes 3 separate functions that illustrate pass by data, pass by address and pass by reference.

\* Debugging, verifying and validating the application.

\*/

#include <stdio.h>

#include <string.h>

#include <iostream>

#include <math.h>

using namespace std;

void explain\_program(void);

void askUserTheSeriesResistance (float &res1, float &res2, float &res3);

float calculate\_series\_resistance\_by\_data (float res1, float res2, float res3);

float calculate\_Parallel\_resistance\_by\_data (float Res1, float Res2, float Res3);

float calculate\_series\_resistance\_by\_address (float \*res1, float \*res2, float \*res3);

float calculate\_Parallel\_resistance\_by\_address (float \*res1, float \*res2, float \*res3);

float calculate\_series\_resistance\_by\_Reference ( float &resistor1, float &resistor2, float &resistor3);

float calculate\_Parallel\_resistance\_by\_Reference ( float &resistor1, float &resistor2, float &resistor3);

void main ()

{

float resistor1;

float resistor2;

float resistor3;

float totalResistance;

explain\_program();

askUserTheSeriesResistance (resistor1 , resistor2 , resistor3);

totalResistance = calculate\_series\_resistance\_by\_data (resistor1, resistor2, resistor3);

printf("\n\nTotal series resistance by data => %f", totalResistance);

totalResistance = calculate\_Parallel\_resistance\_by\_data (resistor1, resistor2, resistor3);

printf("\nTotal Parallel resistance by data => %f", totalResistance);

calculate\_series\_resistance\_by\_address (&resistor1, &resistor2, &resistor3);

calculate\_Parallel\_resistance\_by\_address (&resistor1, &resistor2, &resistor3);

calculate\_series\_resistance\_by\_Reference (resistor1, resistor2, resistor3);

calculate\_Parallel\_resistance\_by\_Reference (resistor1, resistor2, resistor3);

}

/\*----------------------------Function Definition-------------------------\*/

void explain\_program()

{

printf ("\nThis program will ask you to enter three resistor values to calculate series and parallel resistance. ");

printf ("\nThe program includes 3 separate functions for series and parallel calculations that show, pass by data, pass by address and pass by reference.");

} // end of the explain\_program

/\*-----------------Get function asking user for the vales------------------\*/

void askUserTheSeriesResistance (float &resistance1, float &resistance2, float &resistance3)

{

float firstResistance;

float secoundResistance;

float thiredResistance;

printf ("\n\nInput the 1st resistor value in ohms => ");

scanf\_s ("%f", &firstResistance);

printf ("\nInput the 2nd resistor value in ohms => ");

scanf\_s ("%f", &secoundResistance);

printf ("\nInput the 3rd resistor value in ohms => ");

scanf\_s ("%f", &thiredResistance);

resistance1 = firstResistance;

resistance2 = secoundResistance;

resistance3 = thiredResistance;

}

/\*---------------------------Pass by Data---------------------------------\*/

//Series resistance

float calculate\_series\_resistance\_by\_data (float resistor1, float resistor2, float resistor3)

{

return (resistor1 + resistor2 + resistor3);

}

//Parallel resistance

float calculate\_Parallel\_resistance\_by\_data (float resistor1, float resistor2, float resistor3)

{

return (1/((1/resistor1) + (1/ resistor2) + (1/ resistor3)));

}

/\*--------------------------------Pass by Address--------------------------\*/

//Series resistance

float calculate\_series\_resistance\_by\_address (float \*resistor1, float \*resistor2, float \*resistor3)

{

float totalResistance = (\*resistor1) + (\*resistor2) + (\*resistor3);

printf("\n\nTotal series resistance by address => %f", totalResistance);

return totalResistance ;

}

//Parallel resistance

float calculate\_Parallel\_resistance\_by\_address (float \*resistor1, float \*resistor2, float \*resistor3)

{

float total\_Parallel\_Resistance = (1/((1/ \*resistor1) + (1/ \*resistor2) + (1/ \*resistor3)));

printf("\nTotal Parallel resistance by address => %f", total\_Parallel\_Resistance );

return total\_Parallel\_Resistance;

}

/\*-------------------------------Pass by Reference---------------------------\*/

//Series resistance

float calculate\_series\_resistance\_by\_Reference (float &resistor1, float &resistor2, float &resistor3)

{

float totalResistance1 = (resistor1) + (resistor2) + (resistor3);

printf ("\n\nTotal series resistance by Reference => %f", totalResistance1);

return totalResistance1 ;

}

//Parallel resistance

float calculate\_Parallel\_resistance\_by\_Reference (float &resistor1, float &resistor2, float &resistor3)

{

float total\_Parallel\_Resistance1 = (1/((1/ resistor1) + (1/ resistor2) + (1/ resistor3)));

printf ("\nTotal Parallel resistance by Reference => %f", total\_Parallel\_Resistance1 );

return total\_Parallel\_Resistance1;

}

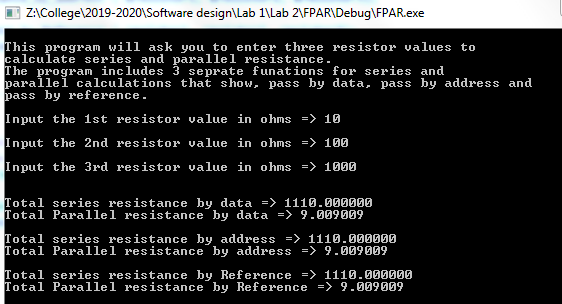


Figure - Debugging the program

The program is calculating the series and parallel resistance. To confirm that the values are right, the online series and parallel calculator was used to confirm the values as shown in fig .2.

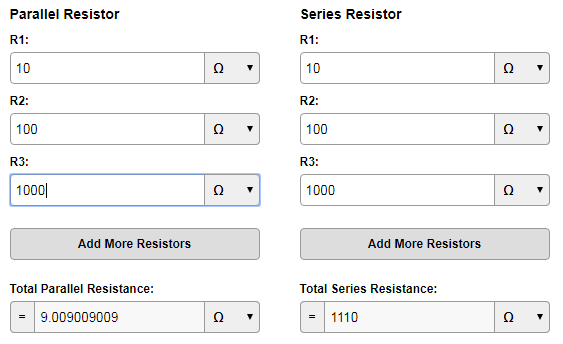


Figure - Total calculated Parallel and series resistance

# Reference

This reference that is shown below, was used to calculate the Parallel and series resistance.

Electronics, D. (2019). *Conversion Calculator Parallel and Series Resistor | DigiKey*. [online] Digikey.com. Available at: https://www.digikey.com/en/resources/conversion-calculators/conversion-calculator-parallel-and-series-resistor [Accessed 15 Oct. 2019].