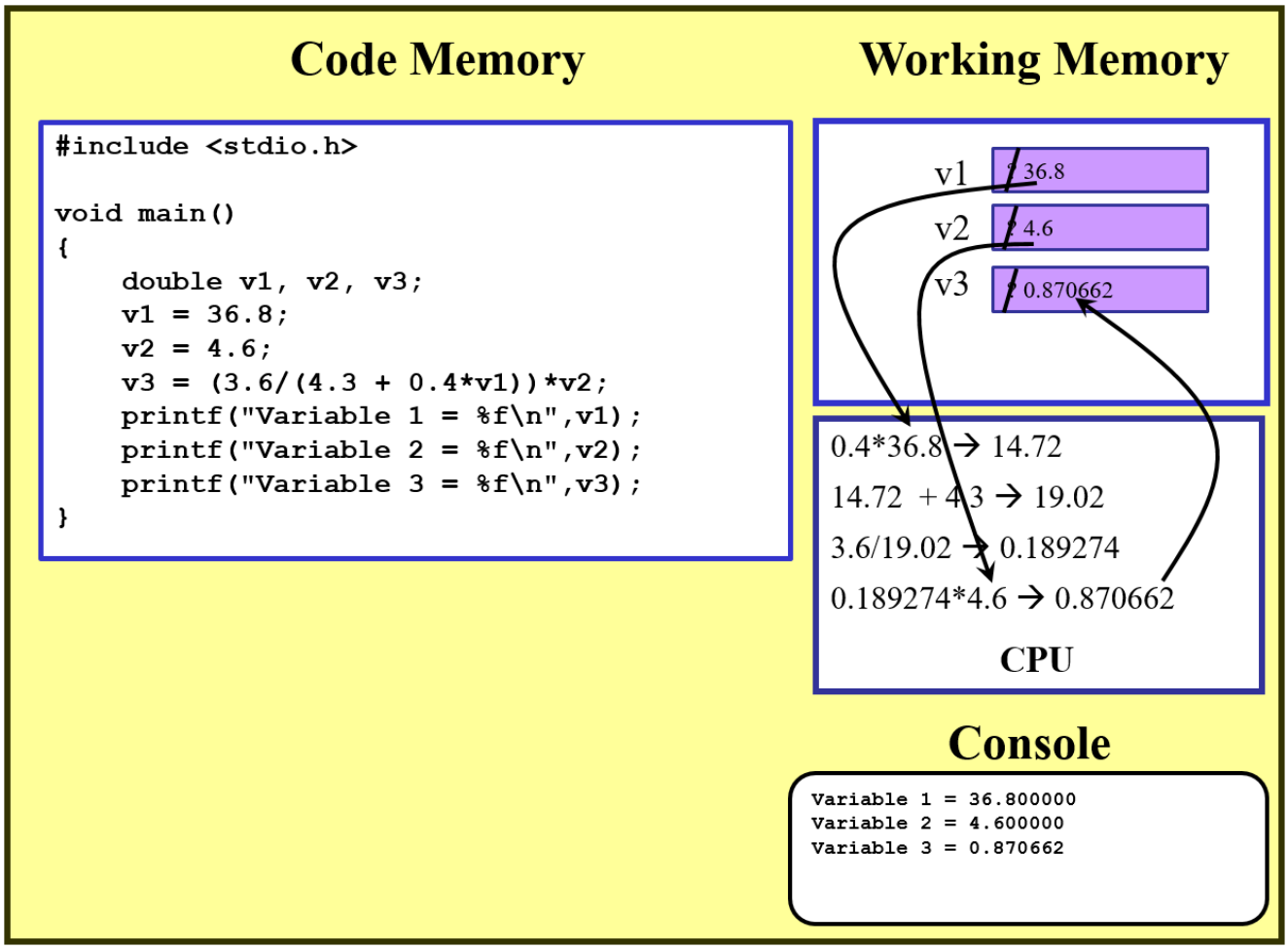
GNG1106 Fall 2018 - Assignment 1 - Solution

Question 1 (10 marks)

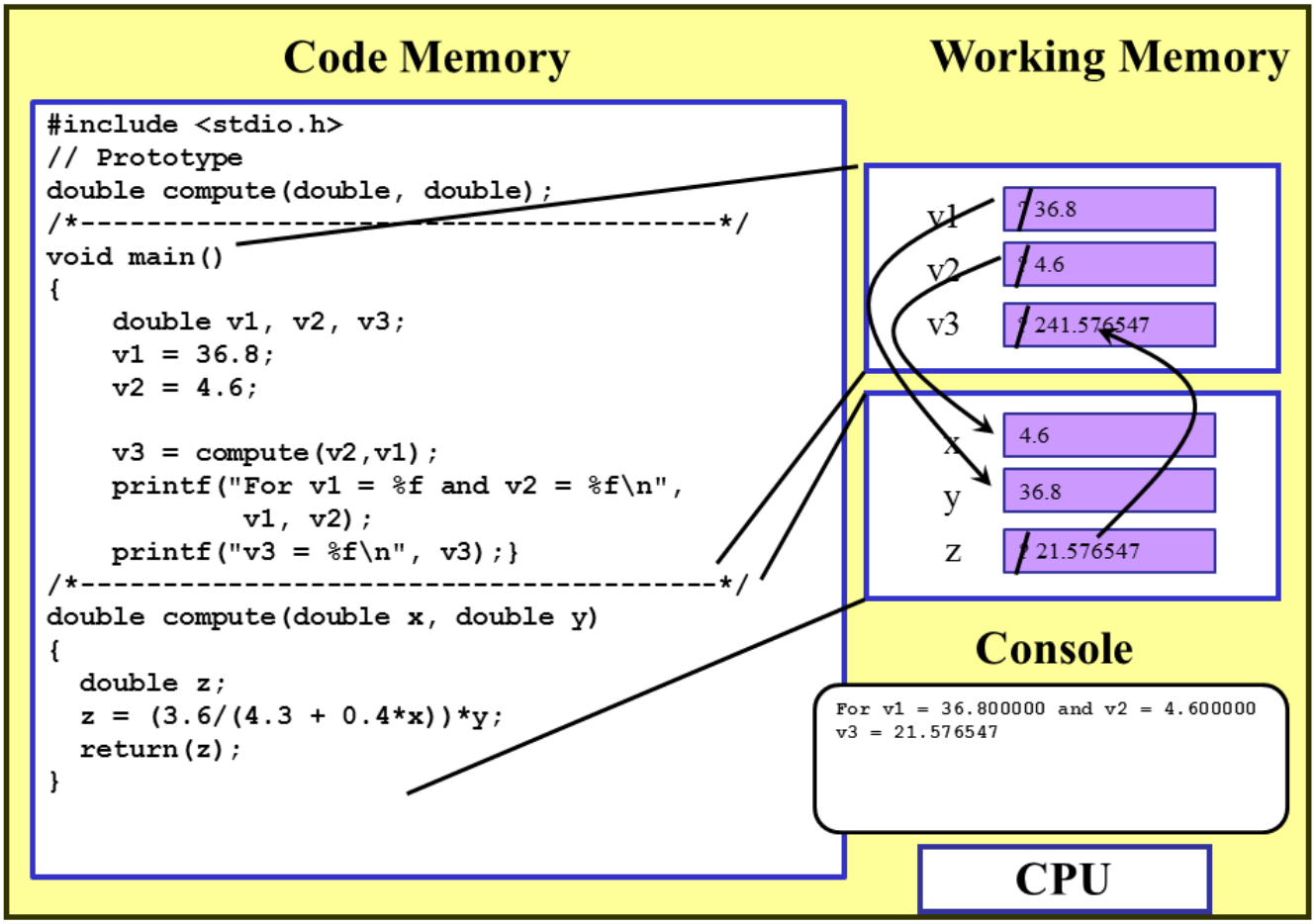
(a) (5 marks)



Marking Scheme:

Drawing the three variables in memory	1 mark (1/3 per variable)
Assigning values to the variables	1 mark (give 0.5 for ?)
Results for each operation	1 mark (0.25 for each operation)
Showing flow from working memory to CPU	1 mark (1/3 for each arrow)
Output in Console	1 mark (1/3 line 1, 0.5 line 2)
Total	5 marks

(b) (5 marks)



Marking Scheme:

Variables in working memory for main 1 marks
- 0.25 for drawing variables
- 0.25 for ?
- 0.5 for values

Variables in working memory for function compute 2.0 marks
- 0.5 for drawing parameters and their values (? should not be present deduct 0.25)
- 0.5 for drawing variable z
- 1 for values in z (0.5 for ? and 0.5 for updated value)

Exchange of values between working memory 1 marks
- 1/3 for each arrow

Output on the console 1 mark (0.5 line 1, 0.5 line 2)

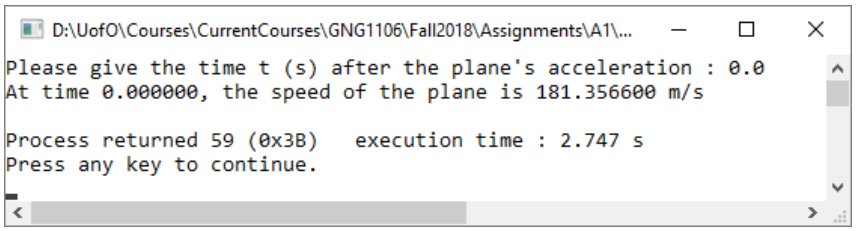
Total 5 marks

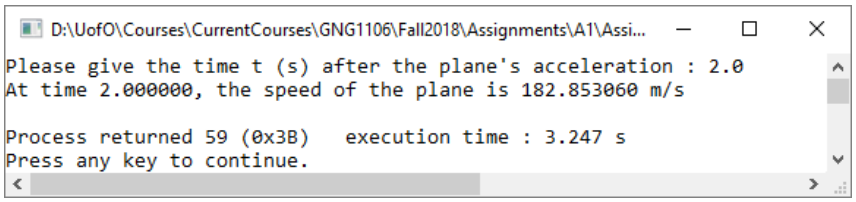
Question 2 (10 marks)

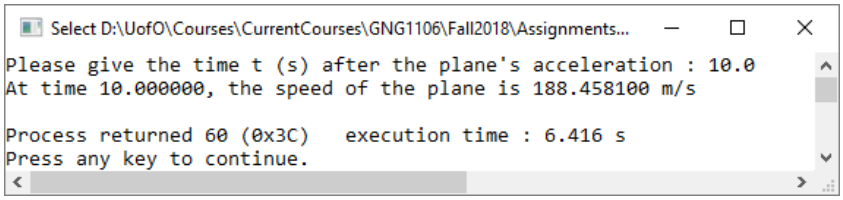
C Source Code

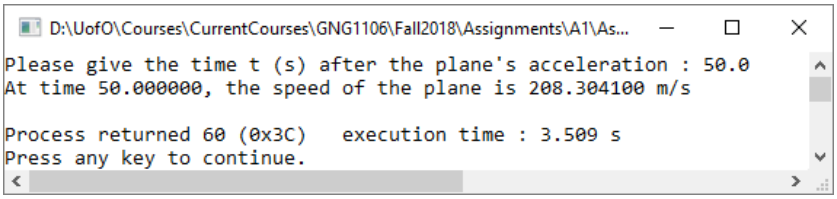
```
/*-------------------------------------------------------
File: A1_question2.c
Author: Gilbert Arbez
Description: The program calculates the speed of a
             plane after its acceleration.
-------------------------------------------------------*/

#include <stdio.h>
#include <math.h>
// Function prototype
double calculateSpeed(double);

/*-------------------------------------------------------
Function: main
Description: Asks the user to give the time t after the
             plane starts its acceleration. Calls
             calculateSpeed to calculate the plane's speed
             at that time and displays results.
-------------------------------------------------------*/
void main()
{
     //Declaration of variables
    double t;  // time
    double v;  // speed (velocity)

    //Initialization of variables
    printf("Please give the time t (s) after the plane's acceleration : ");
    scanf("%lf",&t);

    // Call function to calculate volume of water.
    v = calculateSpeed(t);

    // Display results
    printf("At time %f, the speed of the plane is %f m/s\n",t, v);
}

/*-------------------------------------------------------------
Function: calculeSpeed
Parameters:
    t - time after acceleration
Returns: The speed of the plane at time t.
Description: Calculates the speed of the plane at time t (time after
             the plane starts its acceleration).
-------------------------------------------------------------*/
double calculateSpeed(double t)
{
     //Déclaration of variables
    double speed;  // plane speed
    // Calculate
    speed = 0.00001*t*t*t - 0.00488*t*t + 0.75795*t + 181.3566;
    return(speed);
}
```

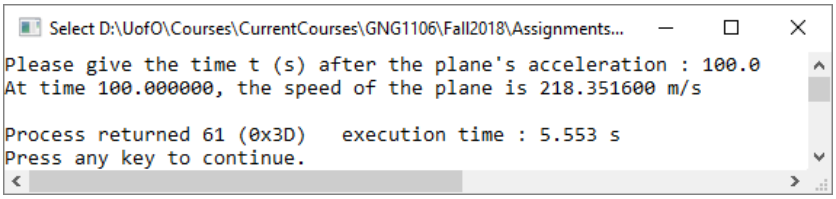
Output

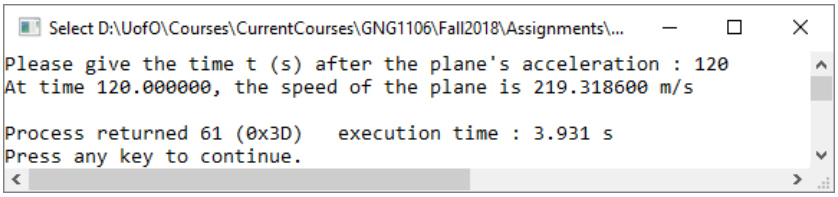

D:\UofO\Courses\CurrentCourses\GNG1106\Fall2018\Assignments\A1\...
Please give the time t (s) after the plane's acceleration : 0.0
At time 0.000000, the speed of the plane is 181.356600 m/s
Process returned 59 (0x3B) execution time : 2.747 s
Press any key to continue.


D:\UofO\Courses\CurrentCourses\GNG1106\Fall2018\Assignments\A1\Assi...
Please give the time t (s) after the plane's acceleration : 2.0
At time 2.000000, the speed of the plane is 182.853060 m/s
Process returned 59 (0x3B) execution time : 3.247 s
Press any key to continue.


Select D:\UofO\Courses\CurrentCourses\GNG1106\Fall2018\Assignments...
Please give the time t (s) after the plane's acceleration : 10.0
At time 10.000000, the speed of the plane is 188.458100 m/s
Process returned 60 (0x3C) execution time : 6.416 s
Press any key to continue.


D:\UofO\Courses\CurrentCourses\GNG1106\Fall2018\Assignments\A1\As...
Please give the time t (s) after the plane's acceleration : 50.0
At time 50.000000, the speed of the plane is 208.304100 m/s
Process returned 60 (0x3C) execution time : 3.509 s
Press any key to continue.


Select D:\UofO\Courses\CurrentCourses\GNG1106\Fall2018\Assignments...
Please give the time t (s) after the plane's acceleration : 100.0
At time 100.000000, the speed of the plane is 218.351600 m/s
Process returned 61 (0x3D) execution time : 5.553 s
Press any key to continue.


Select D:\UofO\Courses\CurrentCourses\GNG1106\Fall2018\Assignments\...
Please give the time t (s) after the plane's acceleration : 120
At time 120.000000, the speed of the plane is 219.318600 m/s
Process returned 61 (0x3D) execution time : 3.931 s
Press any key to continue.

Marking Scheme:

C Program	
Main function	
Comments (header)	1 mark
Variable Declaration	0.5 mark
Input from user	1 mark
Call to **`calculateVolume`**	0.5 mark
Display results	0.5 mark
Function **`calculateVolume`**	
Comments (header)	1 mark
Function header/prototype	1 mark
Variable declaration	0.5 mark
Calculation	0.5 mark
Return instruction	0.5 mark
Output	3.0 marks (0.5 per output)
Total	10 marks

Question 3 (15 marks)

Test Cases (examples)

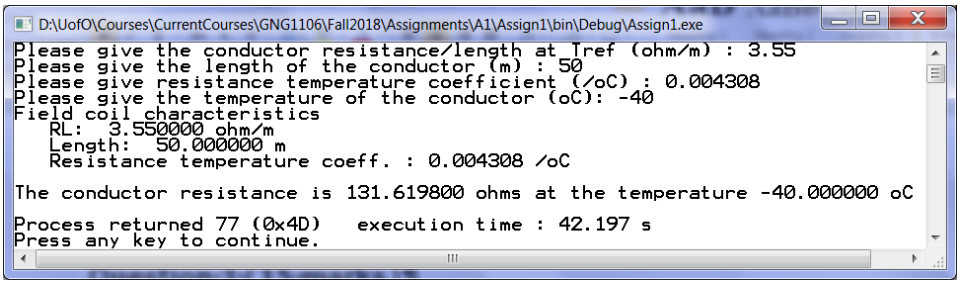
R_L (ohm/m)	Length (m)	Coeff. Alpha (/^oC)	Temperature (^oC)	Rref (ohm)	R (ohm)
3.55	50	0.004308	-40	177.5	131.6198
2.5	25	0.004041	-25	62.5	51.13469
1.2	100	0.004308	0	120	109.6608
0.05	5.5	0.004041	20	0.275	0.275
0.009	2.5	0.004308	31.5	0.0225	0.023615
0.00327	0.01	0.004041	40	3.27E-05	3.53E-05

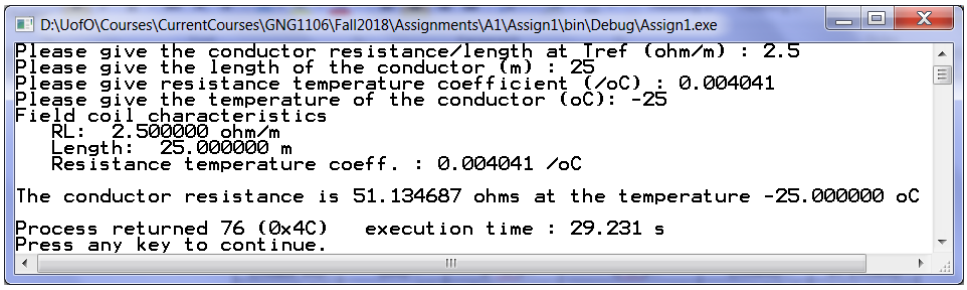
C Source Code

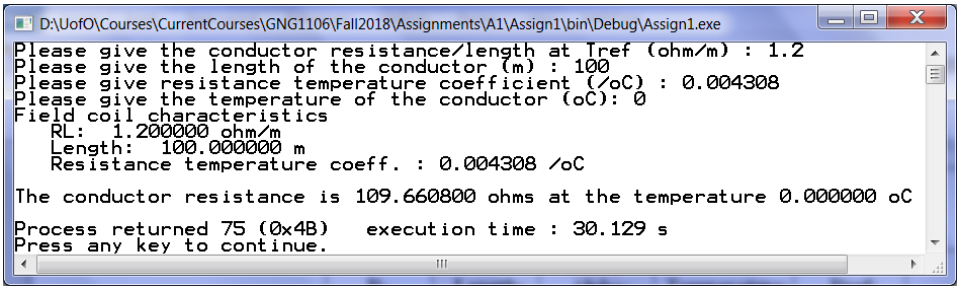
```
/*-------------------------------------------------
File: A1_question3.c
Auteur: Gilbert Arbez
Description: The program calculates the resistance
             value of the conductor in a field coil
             at a given temperature.
-------------------------------------------------------*/
#include <stdio.h>
// Symbolic constant
#define T_REF 20
// Function prototype
double coilResistance(double, double, double, double);
/*-------------------------------------------------------
Fonction: main
Description: Requests from the user the following
                resistance/length, RL, of the coil conductor at 20 oC (Tref)
                length of the conductor (m)
                resistance temperature coefficient (/oC)
                conductor temperature (oC)
             Then calls the function coilResistance to calculate
             the resistance of the field coil conductor at
             the given temperature.
-------------------------------------------------------*/
void main(void)
{
    //declaration of variables
    double rL, length; // resistance/length and conductor length
    double alpha; // resistance temperature coefficient
    double temperature; // temperature of the field coil conductor
    double r;  // resistance of the conductor
    // Data from user
    printf("Please give the conductor resistance/length at Tref (ohm/m) : ");
    scanf("%lf", &rL);
    printf("Please give the length of the conductor (m) : ");
    scanf("%lf", &length);
    printf("Please give resistance temperature coefficient (/oC) : ");
    scanf("%lf", &alpha);
    printf("Please give the temperature of the conductor (oC): ");
    scanf("%lf", &temperature);
    // calculation of resistance
    r = coilResistance(rL, length, alpha, temperature);
    // Display results
    printf("Field coil characteristics\n");
    printf("   RL:  %f ohm/m\n", rL);
    printf("   Length:  %f m\n", length);
    printf("   Resistance temperature coeff. : %f /oC\n", alpha);
    printf("\n");
    printf("The conductor resistance is %f ohms at the temperature %f oC\n",
           r, temperature);
}
```

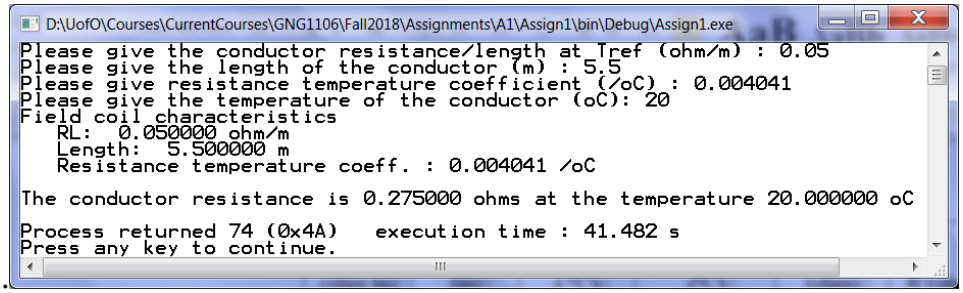
```
/*-------------------------------------------------------------------------------
Function: coilResistance
Parameters:
    rL - resistance/length of the field coil conductor
    len - length of the conductor
    alpha - resistance temperature coefficient
    temp - conductor temperature
Returns:  The temperature of the field coil conductore at the given temperature.
Description: First computes the temperature of the field coil conductor at the
             reference temperature and then at the given temperature.
-------------------------------------------------------------------------------*/
double coilResistance(double rL, double len, double alpha, double temp)
{
    // Declaration of variables
    double r;  // for computation of the temperature
    // Instructions
    // Resistance at the reference temperature
    r = rL * len; // r contiens la valeur de rRef
    // Resistance at the given temperature
    r = r*(1.0 + alpha*(temp - T_REF));
    return(r);
}
```

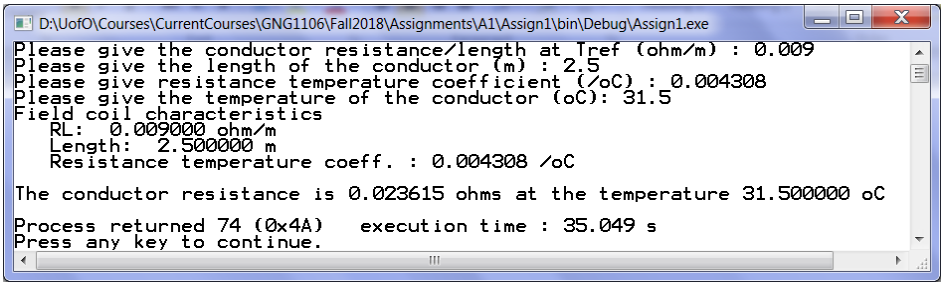
Output

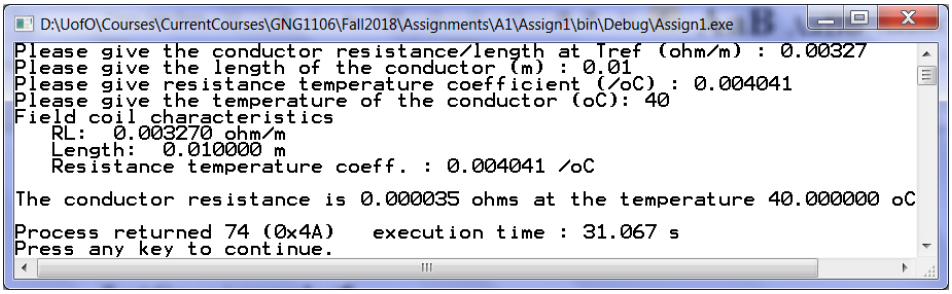












Marking Scheme:

Test Cases — 2.5 marks (0.5 per case for 5 cases)

Deduct 1.0 marks if range of values do not vary sufficiently

Main function

Comments (header)	1 mark
Variable Declaration	1 mark
Input from user	1 mark
Call to **coilResistance**	1 mark
Display results	1 mark

Function for computing the rate of energy radiation (**coilResistance**)

Comments (header)	1 mark
Function header/prototype	1 mark
Variable declaration	1 mark
Calculation	1 mark
Return instruction	1 mark

Output — 2.5 marks (0.5 per output, 5 cases)

Total — 15 marks