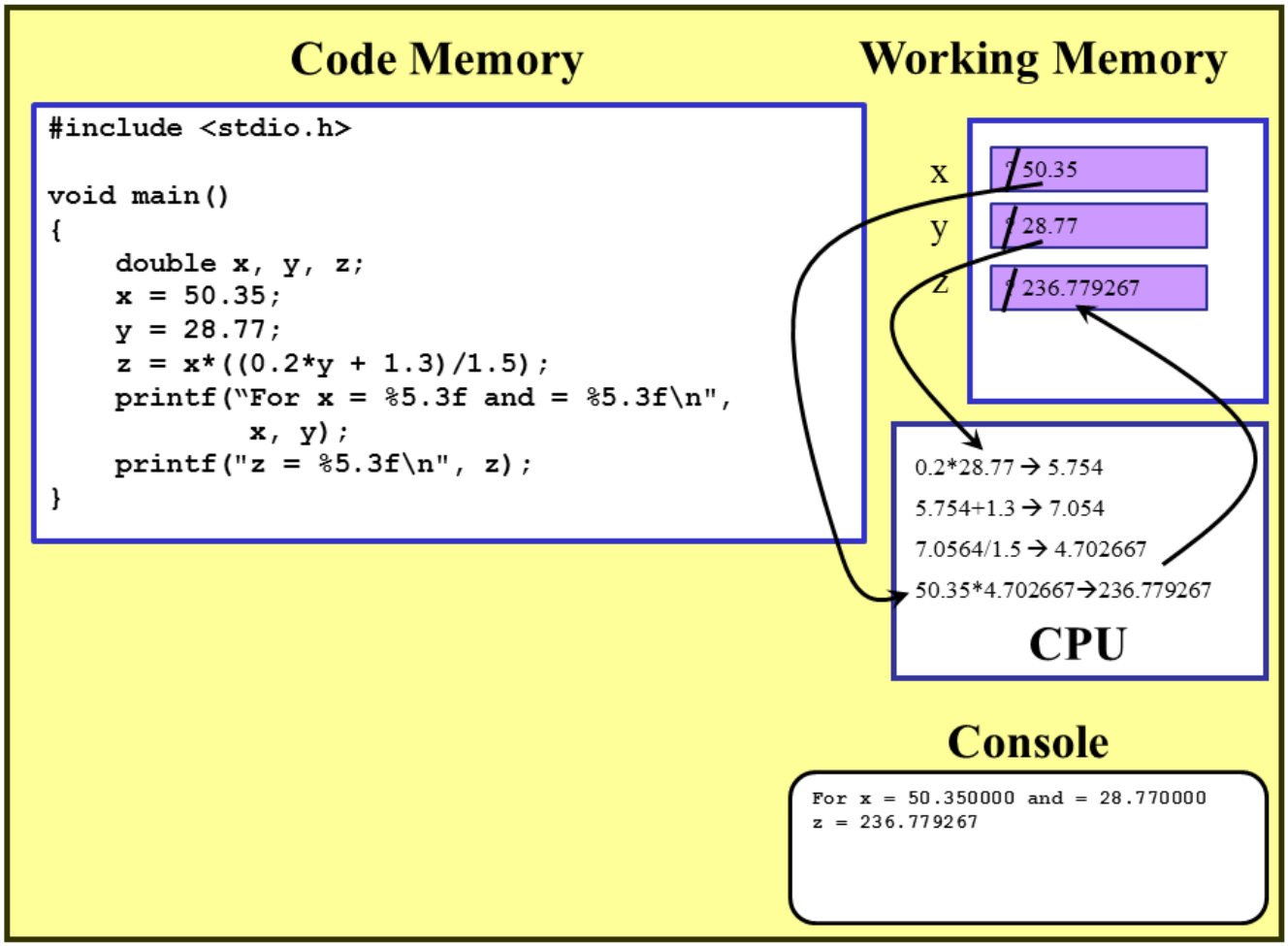
GNG1106 Winter 2018 - Assignment 1 - Solution

Question 1 (10 marks)

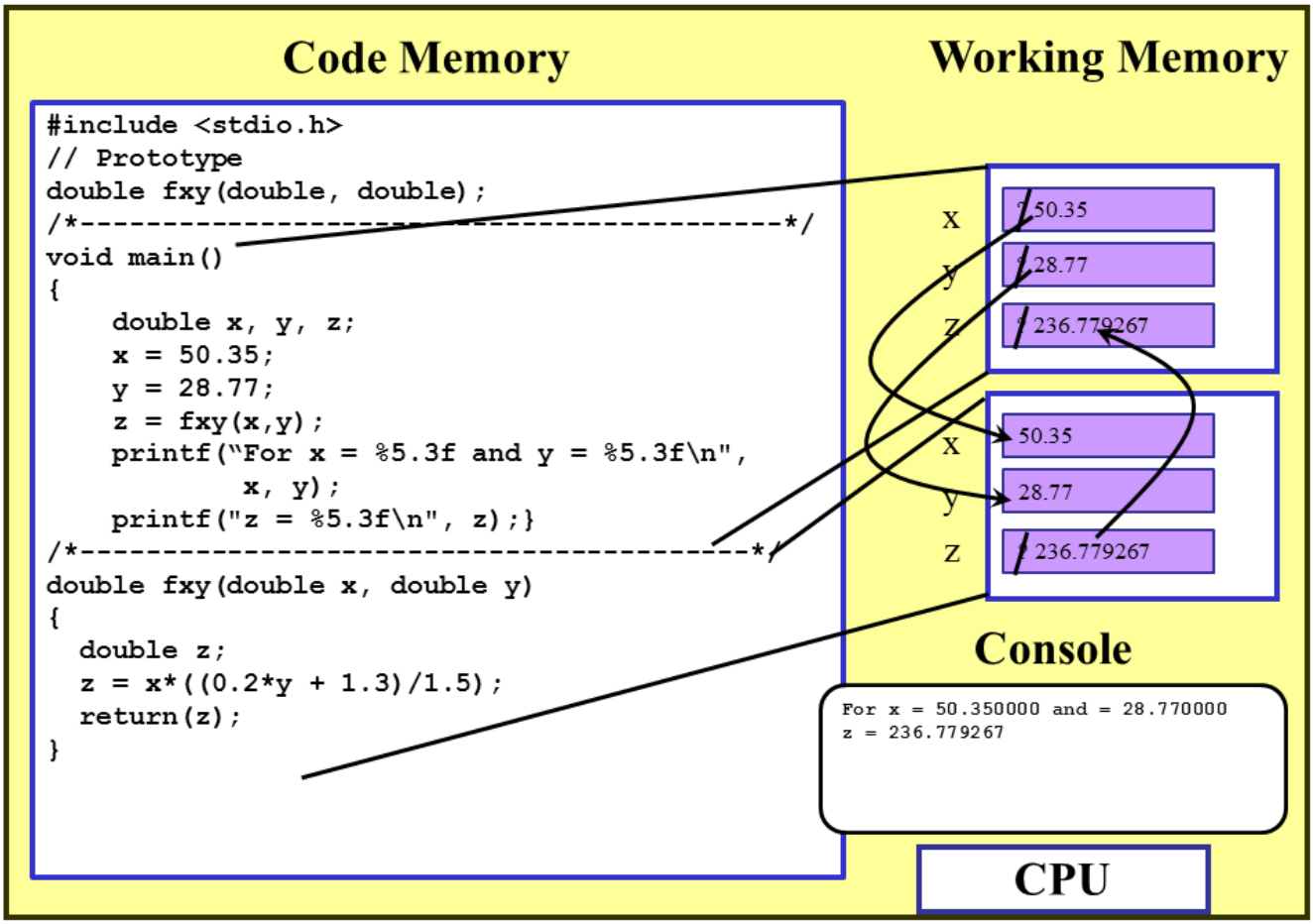
(a) (5 marks)



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 (0.5 line 1, 0.5 line 2)
Total	5 marks

(b) (5 marks)



Marking Scheme:

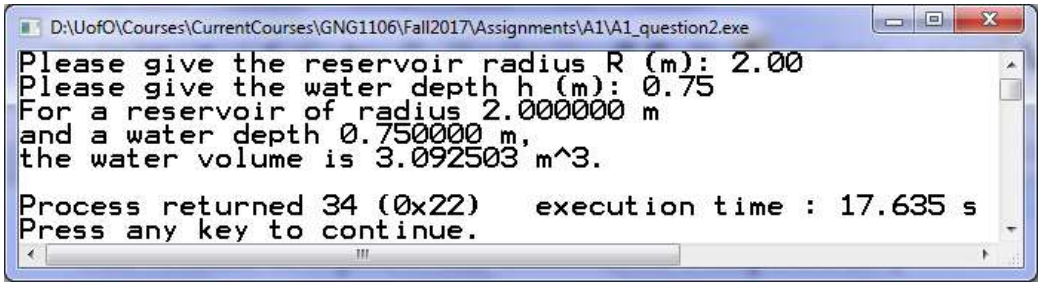
Variables in working memory for main	1 marks
0.25 for drawing variables	
0.25 for ?	
0.5 for values	
Variables in working memory for function fxy	2.0 marks
0.5 for drawing parameters and their values (? should not be present deduct 0.25)	
0.5 for drawing variable z	
1 for values in z (0.5 for ? and 0.5 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

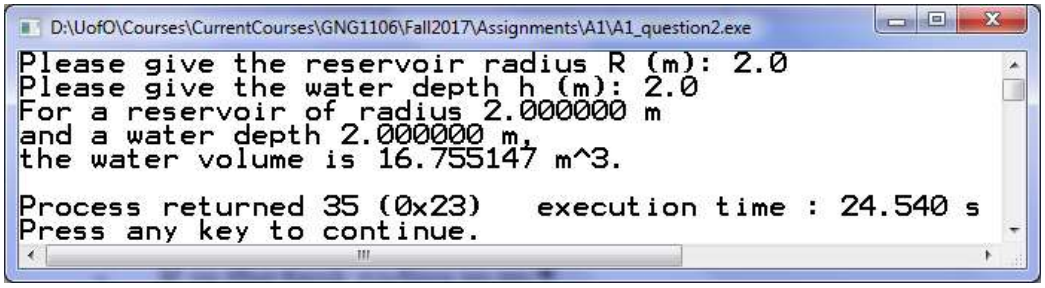
Question 2 (10 marks)

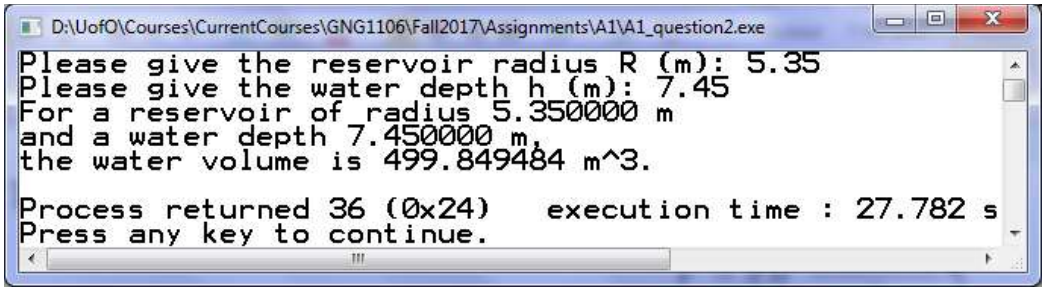
C Source Code

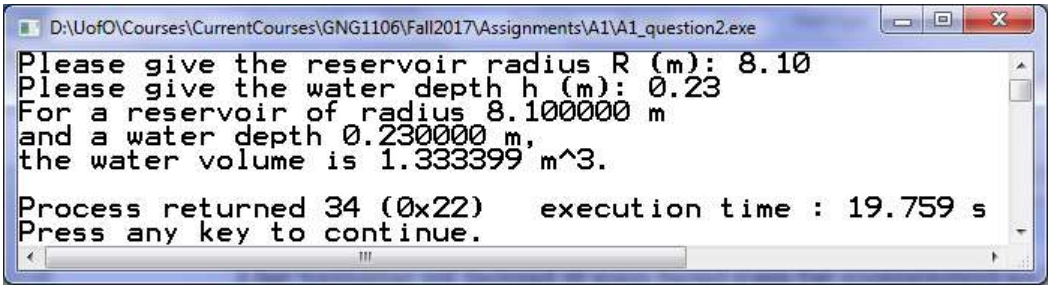
```
/*-------------------------------------------------------
File: A1_question2.c
Auteur: Gilbert Arbez
Description: The program calculates the .
-------------------------------------------------------*/

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

/*-------------------------------------------------------
Function: main
Description: Asks the user to give the radius of the
             reservoir and the depth of the water. Calls
             calculateVolume to calculate the water volume
             in the reservoir.
-------------------------------------------------------*/
void main()
{
     //Declaration of variables
    double r, h, vol;

    //Initialization of variables
    printf("Please give the reservoir radius R (m): ");
    scanf("%lf",&r);
    printf("Please give the water depth h (m): ");
    scanf("%lf", &h);

    // Call function to calculate volume of water.
    vol = calculateVolume(r,h);

    //affichage des resultats
    printf("For a reservoir of radius %f m\n",r);
    printf("and a water depth %f m,\n",h);
    printf("the water volume is %f m^3.\n",vol);
}

/*-------------------------------------------------------------
Function: calculeVolume
Parameters:
    r - reservoir radius
    h - water depth
Retour: The volume of the water in the reservoir.
Description: Calculates the voluem of the water in a spherical
             reservoir with radius r.  The depth of the water is h.
-------------------------------------------------------------*/
double calculateVolume(double r, double h)
{
     //Déclaration of variables
    double vol;  // volume de l'eau
    // Calculate
    vol = 3.14159*h*h*(3.0*r-h)/3;
    return(vol);
}
```

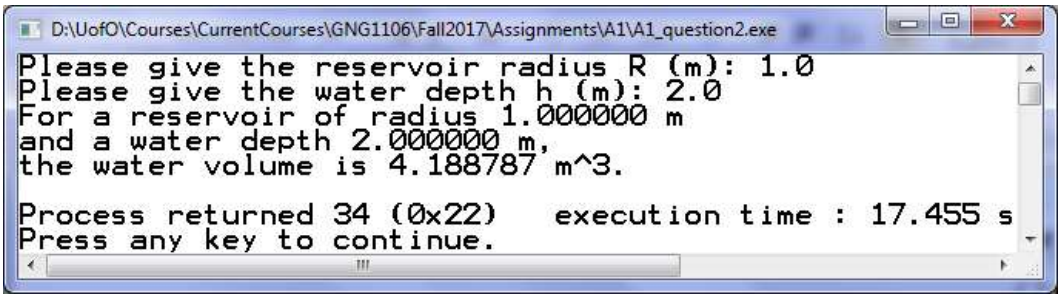
Output


D:\UofO\Courses\CurrentCourses\GNG1106\Fall2017\Assignments\A1\A1_question2.exe
Please give the reservoir radius R (m): 2.00
Please give the water depth h (m): 0.75
For a reservoir of radius 2.000000 m
and a water depth 0.750000 m,
the water volume is 3.092503 m^3.
Process returned 34 (0x22) execution time : 17.635 s
Press any key to continue.


D:\UofO\Courses\CurrentCourses\GNG1106\Fall2017\Assignments\A1\A1_question2.exe
Please give the reservoir radius R (m): 2.0
Please give the water depth h (m): 2.0
For a reservoir of radius 2.000000 m
and a water depth 2.000000 m,
the water volume is 16.755147 m^3.
Process returned 35 (0x23) execution time : 24.540 s
Press any key to continue.


D:\UofO\Courses\CurrentCourses\GNG1106\Fall2017\Assignments\A1\A1_question2.exe
Please give the reservoir radius R (m): 5.35
Please give the water depth h (m): 7.45
For a reservoir of radius 5.350000 m
and a water depth 7.450000 m,
the water volume is 499.849484 m^3.
Process returned 36 (0x24) execution time : 27.782 s
Press any key to continue.


D:\UofO\Courses\CurrentCourses\GNG1106\Fall2017\Assignments\A1\A1_question2.exe
Please give the reservoir radius R (m): 8.10
Please give the water depth h (m): 0.23
For a reservoir of radius 8.100000 m
and a water depth 0.230000 m,
the water volume is 1.333399 m^3.
Process returned 34 (0x22) execution time : 19.759 s
Press any key to continue.


D:\UofO\Courses\CurrentCourses\GNG1106\Fall2017\Assignments\A1\A1_question2.exe
Please give the reservoir radius R (m): 1.0
Please give the water depth h (m): 2.0
For a reservoir of radius 1.000000 m
and a water depth 2.000000 m,
the water volume is 4.188787 m^3.
Process returned 34 (0x22) execution time : 17.455 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`**	1 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	2.5 marks (0.5 per output)
Total	10 marks

Question 3 (15 marks)

Test Cases (examples)

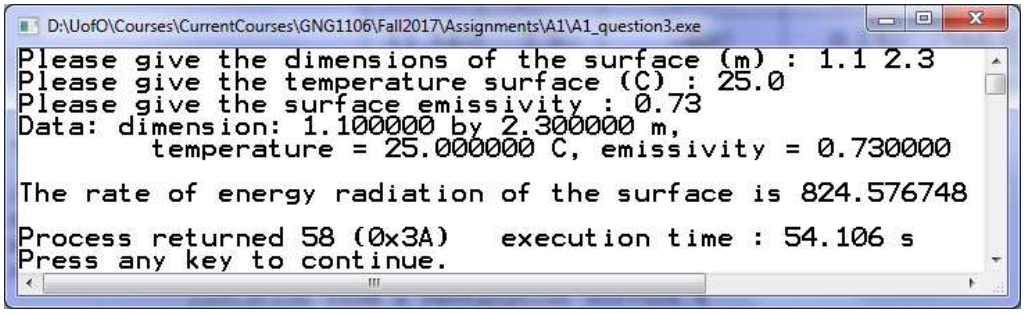
Length	Width	Temperature (centigrade)	Emissivity	H (Watts)
1.1	2.3	25	0.73	824.5767
52.33	81.75	1	0.01	13653.37
0.022	0.01	100	0.45	0.108447
11.34	2.3	-40	0.13	566.0755
0.44	0.83	-80	0.96	27.56953

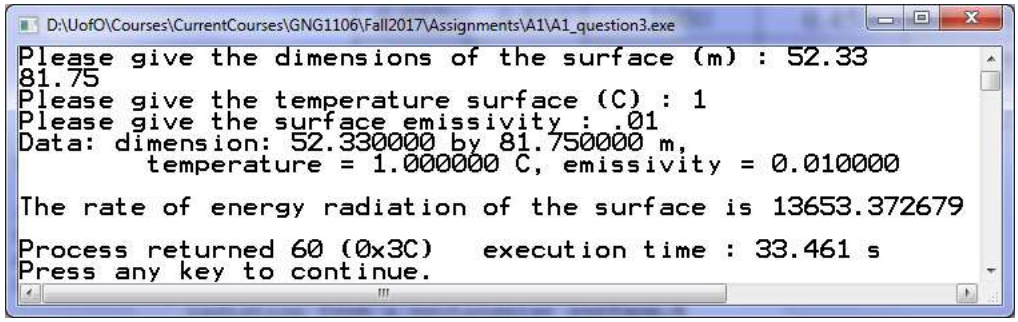
C Source Code

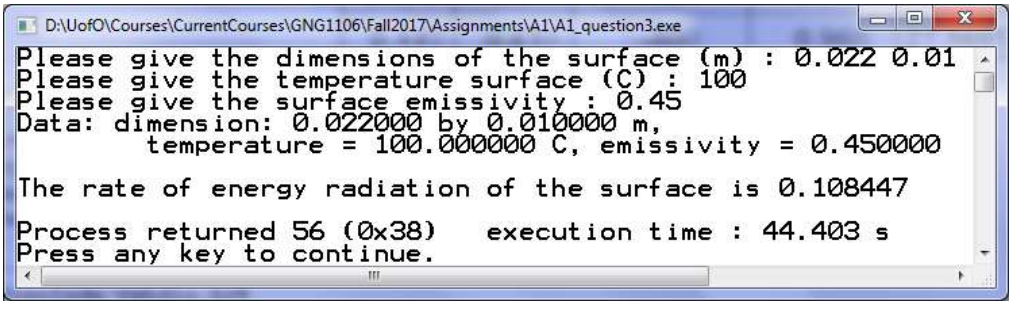
```
/*-------------------------------------------------------
File: A1_question3.c
Auteur: Gilbert Arbez
Description: The program calculates the rate of energy
             radiation from a rectangular surface.
-------------------------------------------------------*/
#include <stdio.h>
// Function prototype
double energyRate(double, double, double, double);
/*-------------------------------------------------------
Fonction: main
Description: Requests from th euser the following
                length and width of the rectangular surface
                surface temperature (en centigrade)
                surface emissivity
             Then calls the function energyRate to calculate
             the rate of energy radiation from the surface.
             The result is displayed to the user.
-------------------------------------------------------*/
void main(void)
{
    //declaration of variables
    double length, width; // dimensions of the surface (m)
    double tempC; // temperature (Centigrade)
    double e; // emmissivity
    double h;  // rate of energy (watts)
    // Data from user
    printf("Please give the dimensions of the surface (m) : ");
    scanf("%lf%lf", &length, &width);
    printf("Please give the temperature surface (C) : ");
    scanf("%lf", &tempC);
    printf("Please give the surface emissivity : ");
    scanf("%lf", &e);
    // calculation of the rate
    h = energyRate(length, width, tempC, e);
    // Display results
    printf("Data: dimension: %f by %f m,\n        temperature = %f C, emissivity = %f\n",
           length, width, tempC, e);
    printf("\n");
    printf("The rate of energy radiation of the surface is %f\n", h);
}
```

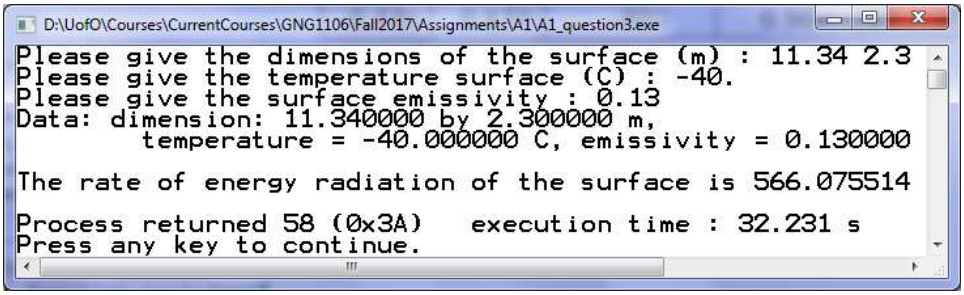
```
/*-------------------------------------------------------------------------------
Fonction: energyRate
Paramètres:
    len, wid - length and width of the surface
    tempC - temperature (in Centigrade) of the surface
    e - emissitivy of surface
Retourne:  The rate of energy radiation in Watts.
Description: Use the following equation to compute the rate of
             radition with the paramters.
                  H = A*e*sigma*tempK^4
             The area A is the multiplication of parameters len and wid. sigma
             is the Steffan-Boltzmann constant = 5.65e-8.
-------------------------------------------------------------------------------*/
double energyRate(double len, double wid, double tempC, double e)
{
    // Declaration of variables
    double h;  // rate of energy radiation (watts)
    double tempK;   // température in Kelvin
    // Instructions
    tempK = tempC + 273.15;
    h = len*wid*e*5.65e-8*tempK*tempK*tempK*tempK;
    return(h);
}
```

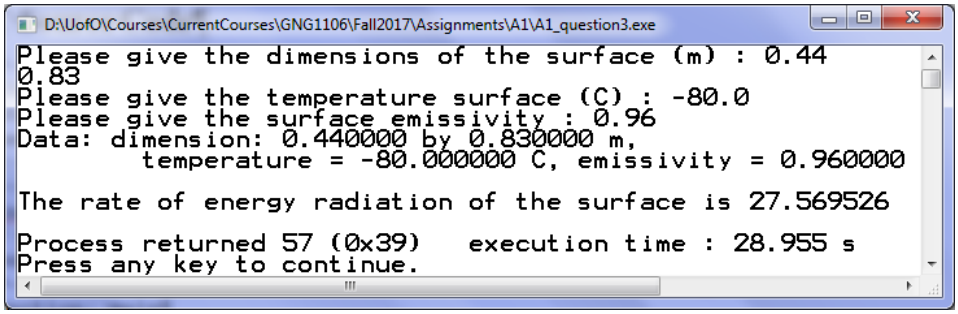
Output











Marking Scheme:

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

Deduct 0.5 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 **rateEnergy**	1 mark
Display results	1 mark

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

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