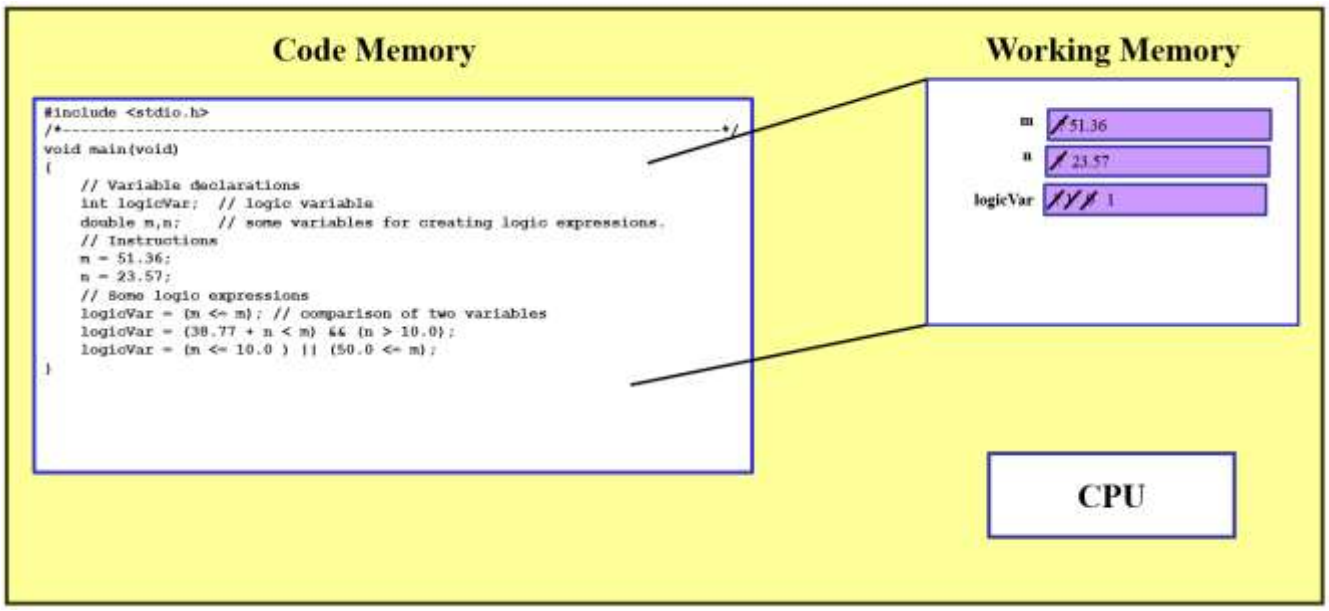
GNG1106 Winter 2018 - Assignment 2 – Solution/Marking Scheme

Question 1 (15 marks)

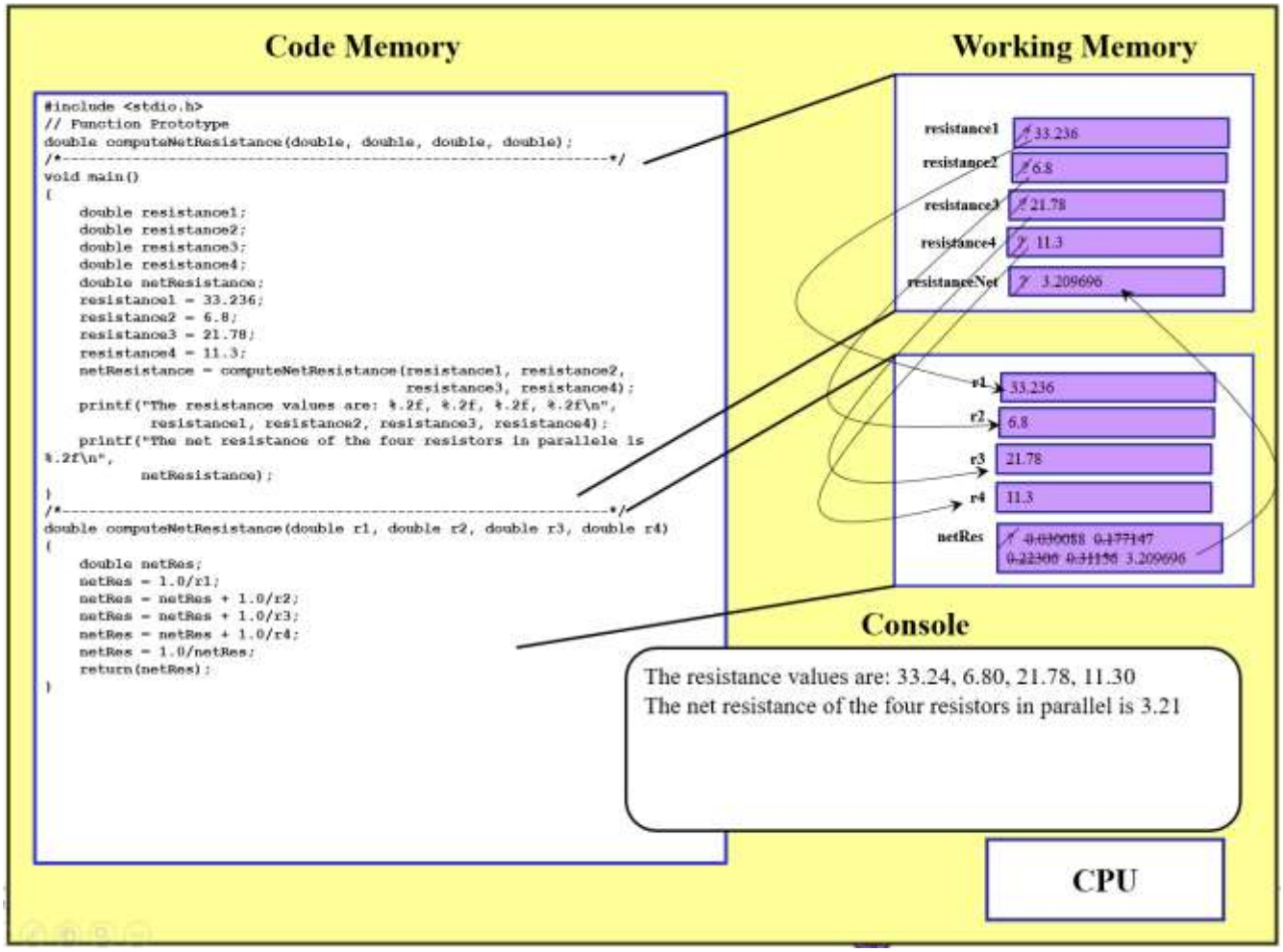
(a) (5 marks)



Marking Scheme:

Defining the three variables in memory	1 marks
Assigning values to the variables m,n	1 marks (0.5 for each value, including ?)
Results for each operation (each value of logicVar)	3 marks (0.75 for each value, including ?)
Total	5 marks

(b) (10 marks)



Marking Scheme:

Variables in working memory for main	2.5 marks
1.25 for variables (1/4 mark each)	
1.25 for values (0.25 for each correct value, deduct 0.5 if ? missing)	
Variables in working memory for function computeNetResistance	3.5 marks
1.25 for parameters and local variable (1/4 mark each)	
1 for parameter values (1/4 mark each, ? should not be present – deduct 0.5 if present)	
1.25 for values of netRes (0.5 for ? and 0.25 for updated values)	
Exchange of values between working memory	2.5 marks
0.5 for each arrow	
Console Output	
0.75 mark for each output line	1.5 marks
Deduct 0.5 if numbers not formatted properly	
Total	10 marks

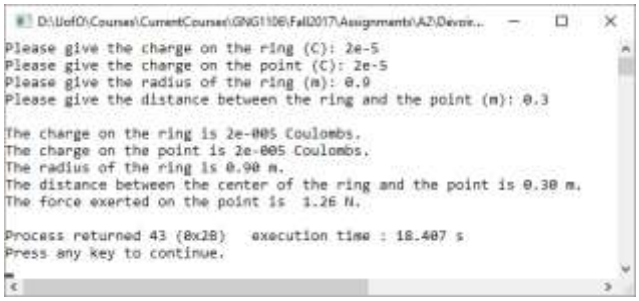
Question 2 (15 marks)

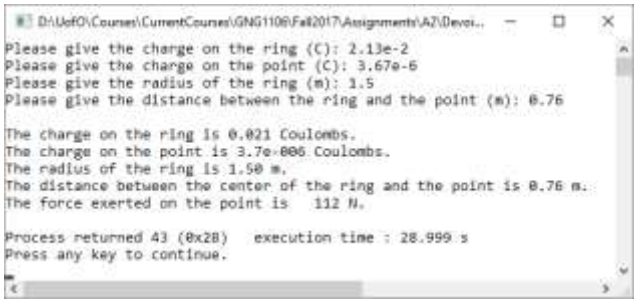
C Source Code

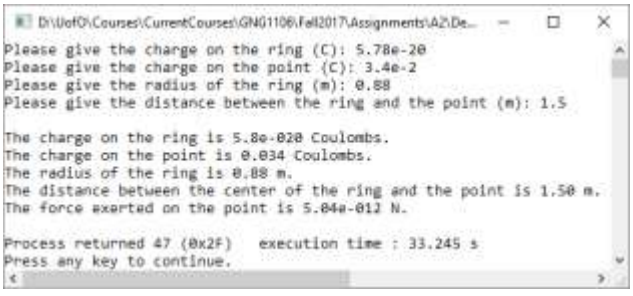
```
/*-------------------------------------------------------
File: Devoir2_question2.c
Author: Gilbert Arbez
Description: This programme calculates the force exerted
             on the point charge by a charged ring.
-------------------------------------------------------*/
#include <stdio.h>
#include <math.h>
// Symbolic constants
#define E0 8.85e-12  // free space permittivity
// Function prototype
double force(double, double, double, double);
/*-------------------------------------------------------
Function: main
Description: Requests the following from the user:
                The charge on the ring and point charge.
                The ring radius.
                The distance between the point and the ring.
             Then calls the function force() to compute the force
             on the point charge.
             The result is displayed to the user.
-------------------------------------------------------*/
void main(void)
{
    // Declaration of variables
    double qRing, qPoint; // the charges (Coulombs)
    double radius; // ring radius (m)
    double x; // distance between the ring and the point (m)
    double f;  // force on the point (N)
    // Obtain data from user
    printf("Please give the charge on the ring (C): ");
    scanf("%lf", &qRing);
    printf("Please give the charge on the point (C): ");
    scanf("%lf", &qPoint);
    printf("Please give the radius of the ring: ");
    scanf("%lf", &radius);
    printf("Please give the distance between the ring and the point (m): ");
    scanf("%lf", &x);
    // Calculate the force
    f = force(qRing, qPoint, radius, x);
    // Display results
    printf("\nThe charge on the ring is %4.2g Coulombs.\n",qRing);
    printf("The charge on the point is %4.2g Coulombs.\n", qPoint);
    printf("The radius of the ring is %4.2f m.\n", radius);
    printf("The distance between the center of the ring and the point is %4.2f m.\n",x);
    printf("The force exerted on the point is %5.3g N.\n",f);
}
```

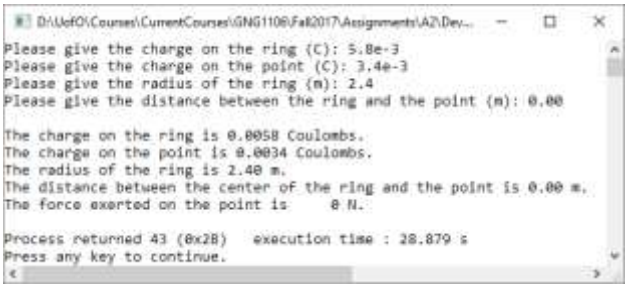
```
/*-------------------------------------------------------------------------------
Function: force
Parameters:
    qR, qP - charge on the ring and point respectivement (Coulomb)
    rad - radius of the ring
    x - distance between the ring and the point (m)
Returns:  The force (N) exerted on the point.
Description: Uses the following equation to compute the force exerted
             on the point charge by the ring.
                  F = (1/(4 PI E0))* qQx / (x^2 + a^2)^1.5
-------------------------------------------------------------------------------*/
double force(double qR, double qP, double rad, double x)
{
    // déclaration des variables
    double f;  // force sur le point
    // Instructions
    f = x*x + rad*rad;
    f = qR*qP*x/pow(f,1.5);
    f = f/(4*M_PI*E0);
    return(f);
}
```

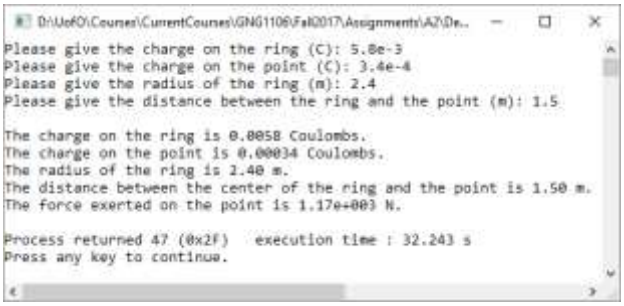
Output











Marking Scheme:

C Program	
Symbolic Constants (E0)	1 marks
Main function	
Comments (header)	1 mark
Variable Declarations	1 mark
Input from user	1 mark
Calls to force	1 mark
Display results	1 mark
Function force	
Comments (header)	1 mark
Function header/prototype	2 mark
Parameters	1 mark
Variable declarations	0.5 mark
Calculation of f (0.25 for each instruction)	1 marks
Return instruction	1 mark
Output (1/2 per output)	2.5 marks
Total	15 marks