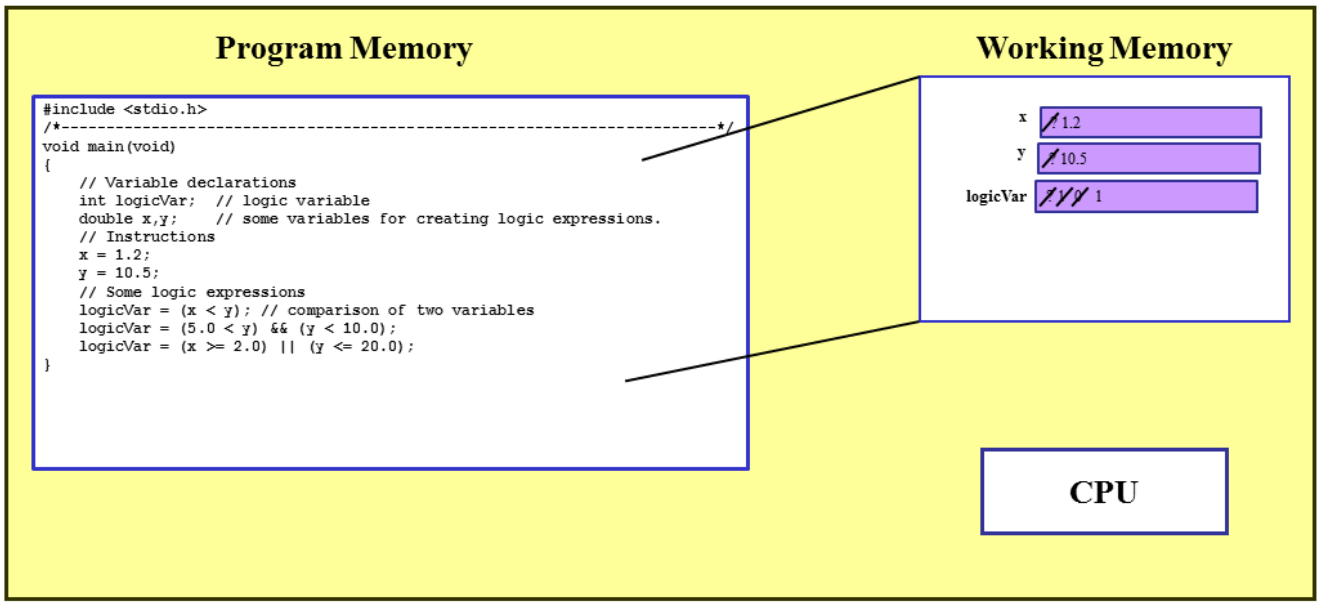
GNG1106 Fall 2016 - Assignment 2 – Solution/Marking Scheme

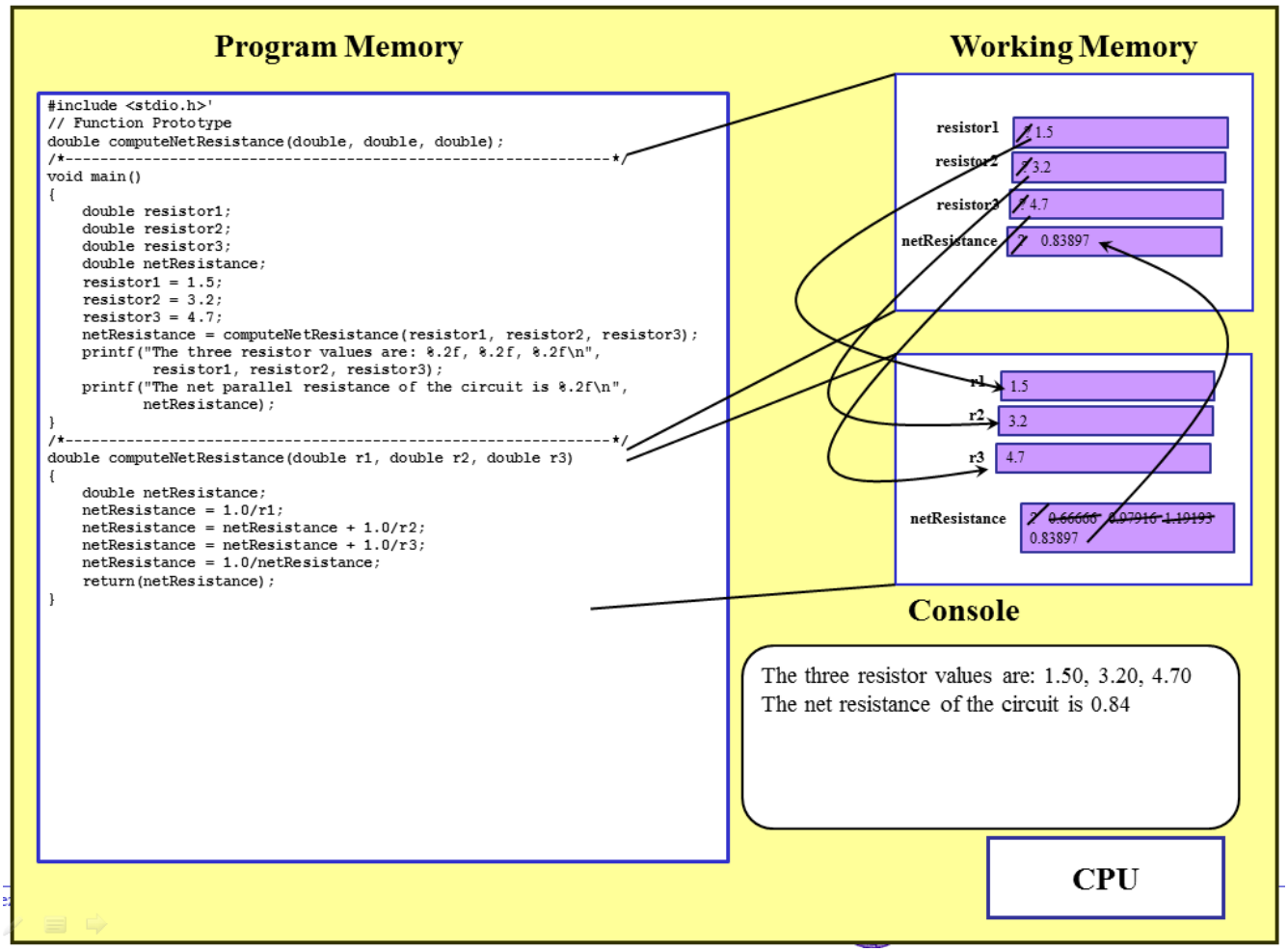
Question 1 (15 marks)

(a)



Marking Scheme:

Defining the three variables in memory	1 marks
Assigning values to the variables x, y	1 marks (0.5 for each value, including ?)
Results for each operation (each value of logicVar)	2 marks (0.5 for each value, including ?)
Total	5 marks



Marking Scheme:

Variables in working memory for main	2.5 marks
1 for variables (1/4 mark each)	
0.5 for ?	
1 for values (0.25 for each correct value)	
Variables in working memory for function computeNetResistance	3.5 marks
1 for parameters and local variable (1/4 mark each)	
1 for parameter values (? should not be present – deduct 0.5 if present)	
1.5 for values of netResistance (0.5 for ? and 0.25 for updated values)	
Exchange of values between working memory	2 marks
0.5 for each arrow	
Console Output	
1 mark for each output message	2 marks
Total	10 marks

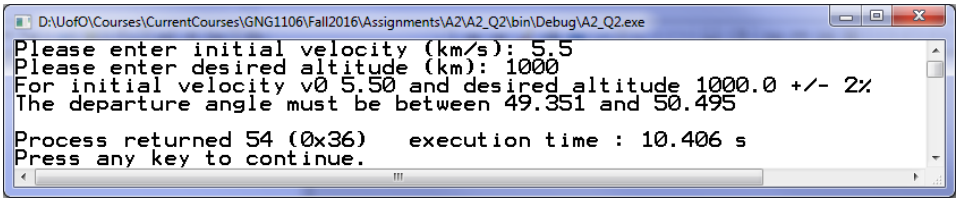
Question 2 (10 marks)

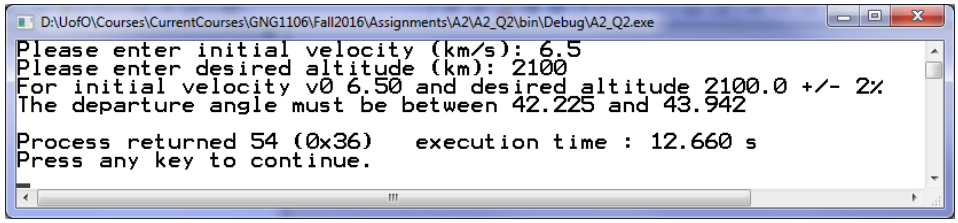
C Source Code

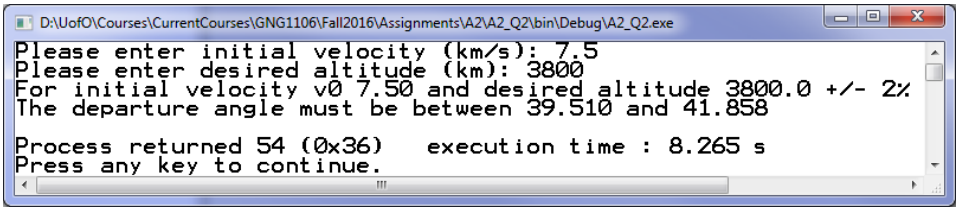
```
/*-------------------------------------------------------------------
File: A2_Q2.c
Author: Gilbert Arbez
Description: Computes initial angle phi0 for lanching missile to
             desired altitude.
-------------------------------------------------------------------*/
#include <stdio.h>
#include <math.h>

// Symbolic Constants
#define R 6371    // km
#define VE 11.2   // km/s
// function prototypes
double angle(double, double);
/*-------------------------------------------------------------------
Function: main
Description: Gets from user initial velocity v0 and desired altitude alt
             and computes angle of departure phi0.
-------------------------------------------------------------------*/
void main(void)
{
    // Variable declarations
    double v0, alt; // initial veloctiy and desired altitude
    double ph0min, ph0max;  // minimum and maximum angle of departure
    // Get User input
    printf("Please enter initial velocity (km/s): ");
    scanf("%lf", &v0);
    printf("Please enter desired altitude (km): ");
    scanf("%lf", &alt);
    // Compute minimum and maximum angles
    ph0min = angle(v0, alt*1.02);
    ph0max = angle(v0, alt*0.98);
    // Display recults
    printf("For initial velocity v0 %.2f and desired altitude %.1f +/-
2%%\n", v0, alt);
    printf("The departure angle must be between %.3f and %.3f\n",ph0min,
ph0max);
}
```

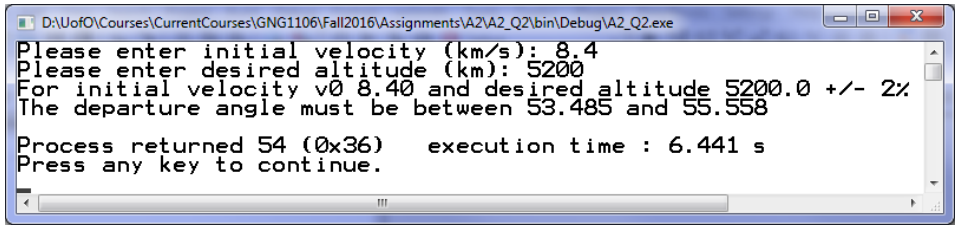
```
/*-------------------------------------------------------------------------
Function: angle
Parameters:
    v0 - initial velocity (km/s)
    alt - desired altitude (km)
Return:  Angle of departure in degrees.
Description: Computes the angle of departure in degrees of a missile  with
             an initial velocity v0 to reach a desired altitude.
-------------------------------------------------------------------------*/
double angle(double v0, double alt)
{
    // Variable declarations
    double alpha;   // Ratio of desired altitude to earth radius
    double phi0;    // angle of departure
    // Determine alpha
    alpha = alt/R;
    // Compute ph0 using accumuilation of intermediate values
    phi0 = pow(VE/v0, 2);
    phi0 = (alpha/(1+alpha))*phi0;
    phi0 = sqrt(1-phi0);
    phi0 = asin((1+alpha)*phi0);
    phi0 = phi0*(180/M_PI);
    // Return value
    return(phi0);
}
```

Output









Marking Scheme:

C Program	
Symbolic Constants (0.5 mark each, R, VE)	1 marks
Main function	
Comments (header)	1 mark
Variable Declarations	1 mark
Input from user	1 mark
Calls to angle	1 marks
Display results	1 mark
Function angle	
Comments (header)	1 mark
Function header/prototype	2 mark
Variable declarations	1 mark
Calculations (1/3 for each calculation, including alpha)	2 marks
Return instruction	1 mark
Output (0.5 per output)	2 marks
Total	15 marks