CODING TAKES PRACTICE!  
**Assignment 3: Arduino Code**In this assignment, you get to practice writing applied C code against your Arduino. A note on collaboration -- it's totally fine (and encouraged) for you to work together with classmates /roommates /pets on the assignment. There is nothing wrong with sharing knowledge.  
However, as you know I want to see individual effort on the assignment submissions. Share knowledge, not answers. Support and encourage the education of your friends and peers.

**WHAT TO DO**

Here's a checklist to help you complete this individual effort assignment.

**Do not use TinkerCad!** It is not a completely accurate simulation of the Arduino.

**Connect your Arduino.** We'll be working in the Arduino IDE and you'll need to be connected to your actual Arduino to make this code run. You don't need to wire anything up.

**Paste in the starter code.** Much of the code is already written for you! Your job is to complete it to make it run. So go ahead and copy the code block below directly into your Arduino IDE code window. Then modify this code until it runs without any errors. There are lots of hints. You'll be adding your code wherever you see three dots.

//  
// Arduino Code  
//  
//  
// CalcAreaOfCircle  
// Complete this function so that it computes and returns the // area of a circle given the radius.  
// Hint: look up how to do exponentiation in Arduino C.  
// Hint: look up how to represent Pi in Arduino C (no it is not // 22/7).  
//  
float CalcAreaOfCircle(float Radius) {  
float Answer = ...  
return (Answer);  
}  
//  
// CalcFactorial  
// Complete this function so that it returns the factorial of // Value.  
// Hint: look up what a factorial is if you don't remember.  
// Hint: look up what an "edge case" means.  
// Hint: look at the preceding function for how to return a // value.  
// Hint: look up how a while loop works.  
//  
unsigned int CalcFactorial(unsigned int Value) {  
// Test for edge case where Value is at or below zero. If so,

// return zero.  
if (Value <= 0) ...  
// Ok, we know Value is at least one so we can start multiplying // factors. Create an unsigned int variable called Answer and // set its initial value to 1.  
...  
// Keep going as long as Value is greater than zero and keep // multiplying factors. Use a  
// while loop to keep repeating this step until Value hits zero.  
while (Value > 0) {  
// Multiply Answer times Value and place the result in Answer.  
...  
// Subtract one from Value and place the result in Value.  
...  
}  
// Return final result.  
return (Answer);  
}  
//  
// CalcSlopeAndIntercept  
// Complete this function so it calculates the slope and // intercept of a line based on two points.  
// Notice that it uses pass-by-reference variables for Slope and // Intercept.  
// Hint: review "pass by reference" to ensure you understand how // it works.  
// Hint: look up what it means to return "void" from a function.  
// Hint: look up how to calculate the absolute value of a number // in Arduino C.  
//  
void CalcSlopeAndIntercept(float X1, float Y1, float X2, float Y2, float &Slope, float &Intercept) {  
// Set the Slope and Intercept return parameters to zero before // we do anything else.  
Slope = ...  
Intercept = ...  
// Test for edge case where X2-X1 is zero (slope is undefined, // line is vertical). If we detect  
// this edge case, just return immediately. Remember we are not // supposed to compare floating point values (conversion // errors). Instead, calculate the absolute value of the //difference  
// between X2 and X1, and test if it is "close" to zero.  
float Difference = ...  
if (Difference < 0.00001) return;  
// Ok, we know we have a non-zero run, so let's calculate it, along with rise.  
float Rise = ...  
float Run = ...  
// Slope is rise over run.  
Slope = ...  
// Intercept is y - mx, using (X1, Y1) for (x, y).  
Intercept = ...  
// We don't need to return anything here, since we've been using pass-by-reference variables  
// for Slope and Intercept. So we're done!  
}  
// -------------------------------------------------------------// -------------------------------------------------------------  
// Don't change anything after this point. It's the usual Arduino setup() and loop() functions  
// but with some test code added.  
//  
//  
// setup()  
//  
void setup()  
{  
Serial.begin(9600);  
Serial.println("Arduino Code Practice");  
Serial.print("Area of circle with radius 5 = ");  
Serial.println(CalcAreaOfCircle(5));  
Serial.print("Factorial of 5 = ");  
Serial.println(CalcFactorial(5));  
float Slope;  
float Intercept;  
Serial.print("Line through (1.5, 2.75) and (3.65, 9.25) has slope, intercept = ");  
CalcSlopeAndIntercept(1.5, 2.75, 3.65, 9.25, Slope, Intercept);  
Serial.print(Slope); Serial.print(", "); Serial.println(Intercept);  
Serial.println();  
}  
//  
// loop()  
//  
void loop()  
{ }

**Submit your Assignment as a zip folder containing your Arduino code using D2L.** Feel free to do online research to discover details; it is expected and required.

Assignment Name: Arduino Code

**Upload the zip file to D2L.** Use **Assignment 3 Dropbox: Arduino Code**.

**You're all done!** Thank you for completing this assignment.

**HOW TO GET TOP MARKS**

Make sure your Arduino code is functional and is "printing" the correct responses out to the Serial Monitor; troubleshoot if not; avoid submitting incomplete or superficial work; follow submission guidelines properly, and submit on time!

**ASSESSMENT CHECKLIST**

|  |  |
| --- | --- |
| Implements the practice functions correctly | /30 |
| Total | /30 |
| Plagiarized or unoriginal work (0% + **academic sanction**) |  |
| Total adjusted | /30 |