Before reading past below instructions:

1. Create an account in Github using your name in this format: lastname\_firstname\_section
2. Request access to [Lycevm<3Alabang · GitHub](https://github.com/Lycevm-3Alabang)
3. Upload this file ON YOUR GITHUB ACCOUNT with answer under the title / file name : E3\_Assessment\_\_[Section]\_[LastnameFirstName]  
   example: E3\_Assessment\_\_BSCS32E1\_AlamoNinoFrancisco

Help: [Get started with GitHub documentation - GitHub Docs](https://docs.github.com/en/get-started)

**Sample Assessment for Introduction to Programming**

This assessment is designed to evaluate your understanding of basic programming concepts in C#, HTML, CSS, and JavaScript.

Instructions: Read each question carefully and provide complete and clear answers. Avoid multiple-choice format responses. Focus on demonstrating your understanding through code, explanations, and discussions.

**Part 1: C# (30 points)**

(10 points) Write a C# program that calculates the area of a triangle given its base and height. Include user input for both values and display the calculated area.

using System;

namespace TriangleCalculator

{

class Program

{

static void Main(String[] args)

{

Console.WriteLine("Enter Base of Triangle: ");

double baseLength = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Enter Height of Triangle: ");

double height = Convert.ToDouble(Console.ReadLine());

double area = baseLength \* height / 2;

Console.WriteLine($"The Area of the Triangle is: {area}");

}

}

}

**(10 points) Declare an array of 5 integers and fill it with values based on a user-defined formula (e.g., n^2). Then, print the largest element in the array.**

using System;

class Program

{

static void Main(string[] args)

{

int[] numbers = new int[5];

Console.WriteLine("Enter a formula to generate values for the array (e.g., n^2):");

string formula = Console.ReadLine();

for (int i = 0; i < numbers.Length; i++)

{

int n = i + 1;

numbers[i] = EvaluateFormula(formula, n);

}

int largest = numbers[0];

for (int i = 1; i < numbers.Length; i++)

{

if (numbers[i] > largest)

{

largest = numbers[i];

}

}

Console.WriteLine($"The largest element in the array is: {largest}");

}

static int EvaluateFormula(string formula, int n)

{

int result = 0;

try

{

result = (int)Math.Pow(n, 2);

}

catch (Exception ex)

{

Console.WriteLine($"Error evaluating formula: {ex.Message}");

}

return result;

}

}

**(10 points) Implement a simple for loop that iterates from 1 to 10 and prints each number along with its square root.**

using System;

class Program

{

static void Main(string[] args)

{

for (int i = 1; i <= 10; i++)

{

double squareRoot = Math.Sqrt(i);

Console.WriteLine($"Number: {i}

Square Root: {squareRoot}");

}

}

}

**Part 2: HTML, CSS, and JavaScript (30 points)**

**HTML (10 points):** You are provided with the following incomplete HTML code snippet:

**HTML**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>My Website</title>**

**</head>**

**<body>**

**<h1>Welcome to...</h1>**

**<p>This is a paragraph...</p>**

**<ul>**

**<li>Item 1</li>**

**<li>Item 2</li>**

**</ul>**

**</body>**

**</html>**

Complete the code snippet by adding the following elements:

An image within the <body> tag with a relevant src attribute.

An ordered list (<ol>) with three items.

A hyperlink within a <p> tag that points to an external website.

A CSS styling rule using an inline style attribute to change the font color of the <h3> heading.

CSS (10 points): Create a CSS stylesheet that defines the following styles:

Change the background color of the body element to light blue.

Apply a padding of 20px to all headings (h1, h2, h3).

Set the font size of the <p> tag to 14px.

Make the list items (li) have a bullet point style instead of the default numbers.

<!DOCTYPE html>

<html>

<head>

<title>My Website</title>

<style>

body {

background-color: lightblue;

}

h1, h2, h3 {

padding: 20px;

}

p {

font-size: 14px;

}

li {

list-style-type: disc;

}

</style>

</head>

<body>

<h1>Hello, we are BTS....</h1>

<p>BTS, also known as the Bangtan Boys, is a South Korean boy band formed in 2010. The band consists of seven members: Jin, Suga, J-Hope, RM, Jimin, V, and Jungkook.</p>

<img src=https://wallpaperaccess.com/full/2981735.jpg>

<li>RM</li></li>

<li>JIN</li>

<li>SUGA</li>

<li>J-HOPE</li>

<li>JIMIN</li>

<li>V</li>

<li>JUNGKOOK</li>

</ul>

To learn more about BTS <a href="https://kprofiles.com/bts-bangtan-boys-members-profile/">click here</a>.</p>

<h3 style="color: red;">That all from me! Annyeong!</h3>

</body>

</html>

**JavaScript (10 points):** Write a JavaScript function that takes a number as input and returns a string indicating whether the number is even or odd. Then, add a button to your HTML page that, when clicked, calls this function and displays the result (even or odd) in a paragraph element below the button.

**Part 3: Essay Question (40 points)**

Discuss the importance of object-oriented programming (OOP) concepts in software development. Explain the key principles of OOP (encapsulation, inheritance, polymorphism, abstraction) and provide examples of how they can be used to create more efficient, maintainable, and reusable code. Include real-world scenarios or cases where OOP is particularly valuable.

Object-oriented programming (OOP) is a widely-used programming paradigm that organizes code into objects that interact with one another. Key principles of OOP include encapsulation, inheritance, polymorphism, and abstraction. Encapsulation bundles data and methods within a single unit, ensuring data integrity and security. Inheritance allows objects to inherit attributes and behaviors from parent classes, promoting a hierarchical structure in programming. Polymorphism allows objects to take on different forms or behaviors depending on the context, promoting flexibility and extensibility in software design. Abstraction hides unnecessary details, allowing for simplification and modularity in software design. OOP is particularly valuable in large-scale software projects, such as e-commerce platforms, where code must be organized, scalable, and maintainable. By understanding and applying these principles effectively, developers can build robust, efficient applications that are easier to maintain, extend, and adapt to changing requirements.

Points Distribution:

Each part carries equal weight (30 points).

Code clarity, functionality, and explanations will be considered in grading.

The essay question focuses on understanding and application of OOP concepts.