Cristian Tagumpay  
BSIT 32E1  
Saturday, 24 February 2024  
Mr. Nino Francisco Alamo  
IT Elective 3 (Full Stack)  
Assessment Prelims

**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. (10 points)**

using System;

class Program

{

static void Main(string[] args)

{

double BaseLength, Height, Area;

// Base

Console.Write("Enter the Base of the Triangle: ");

while (!double.TryParse(Console.ReadLine(), out BaseLength) || BaseLength <= 0)

{

Console.WriteLine("Please enter a valid positive number for the Base.");

Console.Write("Enter the Base of the Triangle: ");

}

// Height

Console.Write("Enter the Height of the Triangle: ");

while (!double.TryParse(Console.ReadLine(), out Height) || Height <= 0)

{

Console.WriteLine("Please enter a valid positive number for the Height.");

Console.Write("Enter the Height of the Triangle: ");

}

//Area

Area = 0.5 \* BaseLength \* Height;

//result

Console.WriteLine($"The Area of the Triangle with Base {BaseLength} and Height {Height} is: {Area}");

}

}

**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. (10 Points)**

using System;

class Program

{

static void Main(string[] args)

{

Console.WriteLine("\nEnter 5 Integers for the array:");

// array of 5 Integers

int[] arr = new int[5];

// enter the array

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

{

Console.Write($"Enter value {i + 1}: ");

if (!int.TryParse(Console.ReadLine(), out arr[i]))

{

Console.WriteLine("Invalid input. Please enter an Integer.");

i--; // Decrement i to retry entering the current element

Console.ReadLine(); // Clear the invalid input from the buffer

}

}

// Print the array elements

Console.WriteLine("\nArray elements:");

foreach (int num in arr)

{

Console.Write(num + " ");

}

Console.WriteLine();

// Find the largest element in the array

int max = arr[0];

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

{

if (arr[i] > max)

{

max = arr[i];

}

}

// Print the largest element

Console.WriteLine("\nLargest element in the array: " + max);

}

}

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

using System;

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Number\tSquare Root");

Console.WriteLine("---------------------");

// Iterate from 1 to 10

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

{

double squareRoot = Math.Sqrt(i);

Console.WriteLine($"{i}\t{squareRoot}");

}

}

}

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

<!DOCTYPE html>

<html>

<head>

<title>My Website</title>

<link rel="stylesheet" type="text/css" href="arandia.css">

</head>

<body>

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

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

<img src="ae.jpg" alt="Description of the image" />

<ul>

<li>Item 1</li>

<li>Item 2</li>

</ul>

<ol>

<li>First Item</li>

<li>Second Item</li>

<li>Third Item</li>

</ol>

<p>

This is a <a href=" https://github.com/Tagumpay-Cristian-BSIT32E1">link</a> to an external website.

</p>

<h3 style="color: red">Heading 3</h3>

<button onclick="checkNumber()">Check Number</button>

<p id="result"></p>

</body>

</html>

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

body

{

background-color: lightblue;

}

h1,

h2,

h3

{

padding: 2.5vh;

}

p

{

font-size: 1.75vh;

}

li

{

list-style-type: disc;

}

**JavaScript: 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. (10 points)**

<script>

function checkNumber() {

var number = parseInt(prompt("Enter a number:"));

if (isNaN(number)) {

document.getElementById("result").textContent =

"Please enter a valid number.";

} else {

if (number % 2 === 0) {

document.getElementById("result").textContent =

number + " is even.";

} else {

document.getElementById("result").textContent = number + " is odd.";

}

}

}

</script>

**Essay Question (40 points)**

The paradigm known as object-oriented programming (OOP) arranges the design of software around objects and data as opposed to functions and logic. Its capacity to offer an organized method for creating, putting into practice, and maintaining complex systems accounts for its significance in the field of software development.

In terms of encapsulation, bundling data and methods that manipulate the data into a single unit called a class is referred to as encapsulation. This idea only exposes the functionality that is necessary through clearly defined interfaces, keeping the internal states of objects hidden from the outside world.

Inheritance, a class (subclass) can inherit methods and properties (behaviour) from another class (superclass) through inheritance. It facilitates the extension and specialization of functionality by allowing the creation of new classes based on existing ones, which encourages code reusability.

Polymorphism, it is possible to treat objects of different classes as belonging to the same superclass thanks to polymorphism. It makes it possible to represent various underlying forms using a single interface.

Abstraction entails using streamlined code representations to model pertinent aspects of a real-world entity. It hides pointless implementation details by concentrating on the functions of an object rather than their methods.