Pseudo CODE IPO PLAN

NICE BONUS Assignment

Brayden

INPUT:

// If radio button rectangular is check.

if (rdbRectangularCalculator.Checked)

{

// if txt.LengthValue is numeric output it to userLength.

if (double.TryParse(txtLegthValue.Text, out userLength) == false)

{

// If user input is not a valid numeric entry, display error message "Please ensure input is numeric".

MessageBox.Show("Please ensure input is numeric");

// Set focus to Length Text Box so user can re-enter valid input.

txtLegthValue.Focus();

}

// Check to see if user entered width is a valid numeric entry.

else if (double.TryParse(txtWidthValue.Text, out userWidth) == false)

{

// If user input is not a valid numeric entry, display error message "Please ensure input is numeric".

MessageBox.Show("Please ensure input is numeric");

// Set focus to Width Text Box so user can re-enter valid input.

txtWidthValue.Focus();

}

Process:

else

{

// Set calculatedRectangularArea variable equal to userLength times userWidth rounded to 2 decimal places.

calculatedRectangularArea = Math.Round(userLength \* userWidth, 2);

// Set calculatedRectangularPerimeter variable equal to userLength times 2 plus userWidth times 2, rounded to 2 decimal places.

calculatedRectangularPerimeter = Math.Round((userLength \* 2) + (userWidth \* 2), 2);

// Set lblRectangularAreaDisplay.Text equal to calculatedRectangularArea variable.

lblRectangularAreaDisplay.Text = calculatedRectangularArea.ToString();

// Set lblRectangularPerimeterDisplay.Text equal to calculatedRectangularPerimeter variable.

lblRectangularPerimeterDisplay.Text = calculatedRectangularPerimeter.ToString();

}

Output:

}

// If radio button Circular is checked.

else if (rdbCircularCalculator.Checked)

{

// If txtRadiusValue.Text is numeric, output it to userRadius

if (double.TryParse(txtRadiusValue.Text, out userRadius) == false)

{

// If user input is not a valid numeric entry, display error message "Please ensure input is numeric".

MessageBox.Show("Please ensure input is numeric");

// Set focus to Radius Text Box so user can re-enter valid input.

txtRadiusValue.Focus();

}

// If user entries are numeric

else

{

// Set calculatedCircularArea equal to Pi times radius squared, rounded to 2 decimal places.

calculatedCircularArea = Math.Round(Math.PI \* (Math.Pow(userRadius, 2)), 2);

// Set calculatedCircularCircumference equal to 2 times Pi times Radius, rounded to 2 decimal places.

calculatedCircularCircumference = Math.Round(2 \* Math.PI \* userRadius, 2);

// Set lblCircularAreaDisplay.Text equal to calculatedCircularArea variable.

lblCircularAreaDisplay.Text = calculatedCircularArea.ToString();

// Set lblCircularCircumferenceDisplay.Text equal to calculatedCircularCircumference variable.

lblCircularCircumferenceDisplay.Text = calculatedCircularCircumference.ToString();

}

}

// If the user has not selected either "Rectangular" or "Circular" radio button

else

{

// Display error message "Please select either rectangular or circular."

MessageBox.Show("Please select either rectangular or circular.");

}

}