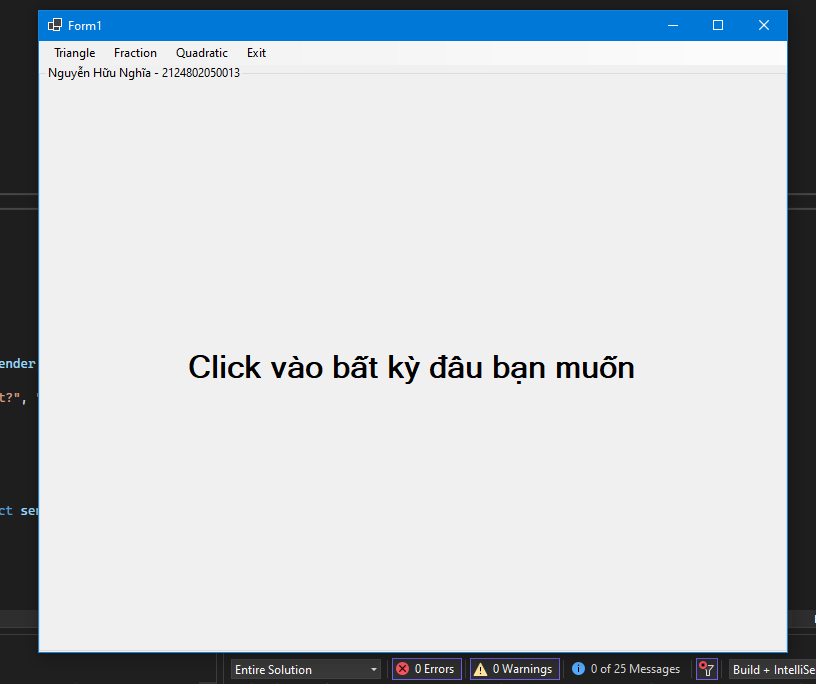
***Nguyễn Hữu Nghĩa – 2124802050013 – D21TTNT01***

1. Giao diện
   1.  Form Main
   2. A screenshot of a computer

      Description automatically generatedFom Triangle
   3. A screenshot of a computer

      Description automatically generatedForm Fraction
   4. Form Quadratic

A screenshot of a computer

Description automatically generated

1. CODE()
   1. Khi click vào mục “Triangle” thì “Form\_Triangle” hiện ra

|  |
| --- |
|  |

* 1. Khi click vào mục “Fraction” thì “Form\_Fraction” hiện ra

|  |
| --- |
|  |

* 1. Khi click vào mục “Quadratic” thì “From\_quadratic” hiện ra

|  |
| --- |
|  |

* 1. Khi nhập số đo ba cạnh tam giác rồi click vào nút “Calculate” thì sẽ hiển thị ra chu vi và diện tích tam giác đó.

Hoặc hiện ra thông báo “Không phải tam giác” nếu 3 số nhập vào không thoã mãn. Click vào nút Retry sẽ làm trống textbox. Click Close sẽ đóng form này lại.

|  |
| --- |
| private bool checkTriangle(int a, int b, int c)  {  if (a + b > c && a + c > b && b + c > a)  {  return true;  }  else  {  MessageBox.Show("Đây không phải 3 cạnh của một tam giác!", "Chú ý");  return false;  }  }  private int cal\_chuvi(int a, int b, int c)  {  return a + b + c;  }  private float cal\_dientich(int a, int b, int c)  {  //s tam giac được tính theo công thức herong  float p = cal\_chuvi(a, b, c) / 2;  float s = (float)Math.Sqrt(p \* (p - a) \* (p - b) \* (p - c));  return s;  }  private void btnCal\_Click(object sender, EventArgs e)  {  try  {  int a = int.Parse(txtA.Text);  int b = int.Parse(txtB.Text);  int c = int.Parse(txtC.Text);  txtP.ReadOnly = txtS.ReadOnly = true;//bật mode readonly của 2 text box chu vi và diện tích  if (checkTriangle(a, b, c))  {  txtP.Text = cal\_chuvi(a, b, c).ToString(); //hiển thị kết quả chu vi lên text box  txtS.Text = cal\_dientich(a, b, c).ToString(); //hiển thị kết quả diện tính lên text box  }  }  catch (Exception ex)  {  MessageBox.Show(ex.Message);//hiển thị lỗi.  }  }  private void btnRetry\_Click(object sender, EventArgs e)  {  txtA.Text = txtB.Text = txtC.Text = string.Empty; // làm trống 3 text box cạnh abc  txtA.Focus();//đặt con trỏ vào text box cạnh a  }  private void btnClose\_Click(object sender, EventArgs e)  {  this.Dispose();//đóng form  }  private void Form\_Triangle\_Load(object sender, EventArgs e)  {  txtA.Focus();//đặt con trỏ vào text box cạnh A  } |
|  |

* 1. Khi nhập vào mẫu và tử rồi click vào nút “Calculate” thì sẽ hiển thị ra phân số tối giản của phân số vừa nhập đó.

Hoặc hiện ra thông báo “Phân số không tồn tại “ nếu mẫu = 0. Click vào nút Retry sẽ làm trống textbox. Click Close sẽ đóng form này lại.

|  |
| --- |
| private int ucln(int a, int b)  {  //tìm ước chung lớn nhất theo phương pháp trừ.  if (a == 0 || b == 0)  {  return a + b;  }  while (a != b)  {  if (a > b)  {  a -= b;  }  else  {  b -= a;  }  }  return a;  }  private void back()  {  txtMau.Text = txtTu.Text = String.Empty;  txtTu.Focus();  }  private void btnCal\_Click(object sender, EventArgs e)  {  try  {  int a = int.Parse(txtTu.Text);  int b = int.Parse(txtMau.Text);  if (b != 0)  {  int u = ucln(a, b);  lbkq.Text = (a / u).ToString() + "/" + (b / u).ToString();  }  else  {  MessageBox.Show("Phân số không tồn tại");  back();  }  }  catch (Exception ex)  {  MessageBox.Show(ex.Message);  }  }  private void btnRetry\_Click(object sender, EventArgs e)  {  back();  }  private void Form\_Fraction\_Load(object sender, EventArgs e)  {  txtTu.Focus();  }  private void btnClose\_Click(object sender, EventArgs e)  {  this.Dispose();  } |
|  |

* 1. Khi nhập vào hệ số của phương trình rồi click vào nút “Calculate” thì sẽ hiển thị ra kết quả tìm nghiệm của phương trình đó. Hoặc hiện ra thông báo “A phải khác 0 “ nếu A = 0. Click vào nút Retry sẽ làm trống textbox. Click Close sẽ đóng form này lại.

|  |
| --- |
| private int gpt(int a, int b, int c, ref float x1, ref float x2)  {  float delta = b \* b - 4 \* a \* c; //tính delta  if (delta < 0)  {  x1 = x2 = 0;  return 0;  }  else if (delta == 0)  {  x1 = x2 = -b / (2 \* a);  return 1;  }  else  {  delta = (float)Math.Sqrt(delta);  x1 = (-b + delta) / (2 \* a);  x2 = (-b - delta) / (2 \* a);  return 2;  }  }  private void btnCal\_Click(object sender, EventArgs e)  {  try  {  float x1 = 0;  float x2 = 0;  int a = int.Parse(txtA.Text);  int b = int.Parse(txtB.Text);  int c = int.Parse(txtC.Text);  if (a == 0)  {  lbkq.Text = "A phải khác 0.";  txtA.Text = string.Empty;  txtA.Focus();  }  else  {  if (gpt(a, b, c, ref x1, ref x2) == 0)  {  lbkq.Text = "Phương trình vô nghiệm.";  }  else if (gpt(a, b, c, ref x1, ref x2) == 1)  {  lbkq.Text = "Phương trình có nghiệm kép: x = " + x1.ToString();  }  else if (gpt(a, b, c, ref x1, ref x2) == 2)  {  lbkq.Text = "Phương trình có 2 nghiêm x1 = " + x1.ToString() + ", x2 = " + x2.ToString();  }  }  }  catch (Exception ex)  {  MessageBox.Show(ex.Message);  }  }  private void btnRetry\_Click(object sender, EventArgs e)  {  txtA.Text = txtB.Text = txtC.Text = string.Empty; txtA.Focus();  }  private void btnClose\_Click(object sender, EventArgs e)  {  this.Dispose(); // đóng form này lại.  } |
|  |