```csharp
using System.Text.RegularExpressions;
using System;
using System.IO;
using static System.Windows.Forms.LinkLabel;
using static System.Windows.Forms.VisualStyles.VisualStyleElement.ListView;

namespace Assignment3
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {

        }
        public delegate bool Validator(string input);
        static void Main()
        {
            string inputFile = @"C:\Users\Ayesha\Documents\student_data.txt";
            string invalidFile = "invalid_entries.txt";

            Validator validateARID = arid => Regex.IsMatch(arid, @"^\d{4}-arid-\d{4}$");
            Validator validateEmail = email => Regex.IsMatch(email, @"^[^@\s]+@[^@\s]+\.[^@\s]+$");

            if (File.Exists(invalidFile))
                File.Delete(invalidFile);

            Console.WriteLine("Processing student_data.txt\n");

            foreach (string line in File.ReadAllLines(inputFile))
            {
                string[] parts = line.Split('|');
                if (parts.Length == 2)
                {
                    string arid = parts[0].Trim();
                    string email = parts[1].Trim();

                    if (validateARID(arid) && validateEmail(email))
                    {
                        Console.WriteLine($"Valid Record: {arid} | {email}");

                    }

                    else
                    {
                        Console.WriteLine($"Invalid Record: {line}");
                        File.AppendAllText(invalidFile, line + Environment.NewLine);
                    }
                }
                else
                {
                    Console.WriteLine($"Invalid Format: {line}");
                    File.AppendAllText(invalidFile, line + Environment.NewLine);
```

```csharp
            }


            Console.WriteLine("\nStep 1 Completed: File processed and invalid entries saved to
'invalid_entries.txt'.");

        }
    }
    }
}
using System.Text.RegularExpressions;
using System;
using System.IO;
using static System.Windows.Forms.LinkLabel;
using static System.Windows.Forms.VisualStyles.VisualStyleElement.ListView;

namespace Assignment3
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {

        }
        public delegate bool Validator(string input);
        static void Main()
        {
            string inputFile = @"C:\Users\Ayesha\Documents\student_data.txt";
            string invalidFile = "invalid_entries.txt";

            Validator validateARID = arid => Regex.IsMatch(arid, @"^\d{4}-arid-\d{4}$");
            Validator validateEmail = email => Regex.IsMatch(email, @"^[^@\s]+@[^@\s]+\.[^@\s]+$");

            if (File.Exists(invalidFile))
                File.Delete(invalidFile);

            Console.WriteLine("Processing student_data.txt\n");

            foreach (string line in File.ReadAllLines(inputFile))
            {
                string[] parts = line.Split('|');
                if (parts.Length == 2)
                {
                    string arid = parts[0].Trim();
                    string email = parts[1].Trim();

                    if (validateARID(arid) && validateEmail(email))
                    {
                        Console.WriteLine($"Valid Record: {arid} | {email}");

                    }
```

```csharp
                    else
                    {
                        Console.WriteLine($"Invalid Record: {line}");
                        File.AppendAllText(invalidFile, line + Environment.NewLine);
                    }
                }
                else
                {
                    Console.WriteLine($"Invalid Format: {line}");
                    File.AppendAllText(invalidFile, line + Environment.NewLine);
                }


                Console.WriteLine("\nStep 1 Completed: File processed and invalid entries saved to 'invalid_entries.txt'.");

            }
        }
    }
}
```