Lab 09:

Code:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace CCLab09

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void btnParse\_Click(object sender, EventArgs e)

{

string input = txtInput.Text;

lstParseSteps.Items.Clear();

if (ParseS(input))

{

lblResult.Text = "Valid input based on the grammar!";

}

else

{

lblResult.Text = "Invalid input!";

}

}

// Parse S → A B | C

private bool ParseS(string input)

{

lstParseSteps.Items.Add("Starting parse for S...");

// Try to parse as A B

if (ParseA(input))

{

lstParseSteps.Items.Add("Parsed A.");

if (ParseB(input.Substring(input.Length - (input.Length / 2))))

{

lstParseSteps.Items.Add("Parsed B.");

return true;

}

}

// Try to parse as C

if (ParseC(input))

{

lstParseSteps.Items.Add("Parsed C.");

return true;

}

lstParseSteps.Items.Add("Invalid parse for S.");

return false;

}

// Parse A → a A | a

private bool ParseA(string input)

{

if (input.Length > 0 && input[0] == 'a')

{

lstParseSteps.Items.Add("Parsing A...");

return ParseA(input.Substring(1)) || input.Length == 1; // Base case for single 'a'

}

return false;

}

// Parse B → b B | b

private bool ParseB(string input)

{

if (input.Length > 0 && input[0] == 'b')

{

lstParseSteps.Items.Add("Parsing B...");

return ParseB(input.Substring(1)) || input.Length == 1; // Base case for single 'b'

}

return false;

}

// Parse C → c C | d

private bool ParseC(string input)

{

if (input.Length > 0 && input[0] == 'c')

{

lstParseSteps.Items.Add("Parsing C...");

return ParseC(input.Substring(1)) || input.Length == 1; // Base case for multiple 'c's

}

if (input.Length == 1 && input[0] == 'd') // Single 'd' is also valid for C

{

lstParseSteps.Items.Add("Parsed single 'd' in C.");

return true;

}

return false;

}

}

}

Output:



