LAB LL PARSE

```
using System;
using System.Collections.Generic;

public class LL1Parser
{
    private List<string> tokens;
    private int index;
    private string currentToken;

    public LL1Parser(List<string> tokens)
    {
        this.tokens = tokens;
        this.index = 0;
        this.currentToken = tokens.Count > 0 ? tokens[this.index] : null;
    }

    private void Match(string expectedToken)
    {
        if (currentToken == expectedToken)
        {
            index++;
            currentToken = index < tokens.Count ? tokens[index] : null;
        }
        else
        {
            throw new InvalidOperationException($"Expected {expectedToken}, but got {currentToken}");
        }
    }

    public void Parse()
    {
        S();
        if (currentToken != null)
        {
            throw new InvalidOperationException($"Unexpected token {currentToken} at the end of input.");
        }
        Console.WriteLine("Parsing successful!");
    }

    private void S()
    {
        if (currentToken == "(")
        {
            Match("(");
            C();
            Match("x");
            Match("y");
            S_();
        }
        else if (currentToken == "d")
        {
            Match("d");
            Match("y");
            S_();
        }
        else if (currentToken == "b")
        {
            Match("b");
            S_();
        }
        else
        {
            throw new InvalidOperationException($"Unexpected token {currentToken} in S");
        }
    }

    private void S_()
    {
        if (currentToken == "x")
        {
            Match("x");
            Match("y");
```

```
            S_();
        }
        else if (currentToken == null || currentToken == "y")
        {
            // epsilon case: do nothing
            return;
        }
        else
        {
            throw new InvalidOperationException($"Unexpected token {currentToken} in S'");
        }
    }

    private void C()
    {
        if (currentToken == "e")
        {
            Match("e");
            C_();
        }
        else
        {
            throw new InvalidOperationException($"Unexpected token {currentToken} in C");
        }
    }

    private void C_()
    {
        if (currentToken == "m")
        {
            Match("m");
            C_();
        }
        else if (currentToken == "x")
        {
            // epsilon case: do nothing
            return;
        }
        else
        {
            throw new InvalidOperationException($"Unexpected token {currentToken} in C'");
        }
    }

    public static void Main(string[] args)
    {
        try
        {
            // Ask the user to input tokens
            Console.WriteLine("Enter the tokens separated by spaces:");
            string input = Console.ReadLine();

            // Tokenize the input
            List<string> tokens = new List<string>(input.Split(' '));

            // Create a parser instance
            LL1Parser parser = new LL1Parser(tokens);


             // Parse the tokens
             parser.Parse();
        }
        catch (Exception ex)
        {
            Console.WriteLine($"Parsing failed: {ex.Message}");
        }
    }
}
```

OUTPUT:

```
Enter the tokens separated by spaces:
( e m x y x y
Parsing successful!
```