

Proyecto III: Evaluación de expresiones matemáticas

Instituto Tecnológico de Costa Rica
Escuela de Ingeniería en Computadores
Algoritmos y Estructuras de Datos II (CE 1103)

Verano 2024-2025

1. Descripción General

Sistema cliente/servidor para evaluación de expresiones matemáticas utilizando árboles de expresión binarios. Desarrollado para el curso de Algoritmos y Estructuras de Datos II del TEC.

2. Tecnologías y Herramientas

- C# (.NET 8)
- Windows Forms (interfaz gráfica del cliente)
- Sockets TCP (comunicación cliente/servidor)
- CSV Helper (manejo de archivos CSV)
- Git/GitHub
- Azure DevOps (gestión del proyecto)
- Visual Studio 2022

3. Estructura del Proyecto

```
ExpressionCalculator/  
  Server/  
    Program.cs          # Punto de entrada del servidor  
    Models/  
      ExpressionTree.cs # Implementación del árbol  
      Node.cs           # Nodo del árbol  
      Operation.cs      # Registro de operaciones  
    Services/  
      ServerSocket.cs   # Manejo de conexiones TCP  
      ExpressionEvaluator.cs # Evaluación de expresiones  
      OperationLogger.cs # Registro en CSV  
  Client/  
    Program.cs          # Punto de entrada del cliente  
    Forms/  
      MainForm.cs       # Interfaz gráfica  
    Services/  
      ClientSocket.cs    # Conexión con servidor  
      ExpressionValidator.cs # Validación de entrada
```

4. División de Responsabilidades

4.1. Steven

1. Cliente TCP y GUI (ID 001 parcial, ID 002)
 - Interfaz gráfica en Windows Forms
 - Conexión con el servidor
 - Validación de expresiones
2. Servidor TCP (ID 001 parcial)
 - Implementación del servidor multithreading
 - Manejo de múltiples conexiones
 - Protocolo de comunicación
3. Árboles de Expresión (ID 001 parcial)
 - Implementación de ExpressionTree
 - Algoritmo de evaluación recursiva
 - Conversión a notación postfija
4. Operaciones Matemáticas (ID 001 parcial)
 - Implementación de operadores (+, -, *, /...)
 - Manejo de paréntesis
 - Precedencia de operadores matemáticos
5. Implementación Azure DevOps (ID 005 parcial)
 - Configuración del proyecto en Azure
 - Implementación de los sprints definidos
 - Seguimiento de tareas

4.2. Julián

1. Operaciones Lógicas (ID 001 parcial)
 - Implementación de operadores (and, or, not, xor)
 - Integración con operadores matemáticos
 - Precedencia de operadores lógicos
2. Registro de Operaciones (ID 003)
 - Sistema de logging
 - Manejo de archivos CSV
 - Formato de almacenamiento
3. Visualización de Historial (ID 004)
 - Tabla de expresiones evaluadas
 - Despliegue de fecha, expresión y resultado
 - Actualización al recibir nuevos resultados
4. Historias de Usuario (ID 005 parcial)
 - Creación de user stories para cada requerimiento
 - Definición de los tres sprints
 - Documentación de criterios de aceptación

5. Orden de Implementación

5.1. Steven

1. Implementación básica del servidor TCP
2. Setup inicial de la GUI del cliente
3. Conexión básica cliente-servidor
4. Implementación del árbol de expresión base
5. Implementación de la conversión a notación postfija
6. Implementación de operaciones matemáticas (+, -, *, /...)
7. Manejo de múltiples clientes
8. Refinamiento de la GUI
9. Implementación de Azure DevOps según historias

5.2. Julián

1. Definición de historias de usuario
2. Planificación de sprints
3. Implementación de operaciones lógicas básicas (and, or)
4. Implementación de operaciones lógicas avanzadas (not, xor)
5. Integración con el árbol de expresión
6. Sistema básico de logging
7. Implementación del registro CSV
8. Visualización de expresiones evaluadas

6. Control de Versiones

- Cada funcionalidad se desarrolla en una rama separada de nombre "feature/funcionalidad"
- Pull requests a develop cuando el feature esté completo
- Code review obligatorio para cada PR

7. Reuniones Programadas

- 17 enero — 7:00 PM: Planificación
- Organizar más reuniones por telegram