



Nombre: Eduardo Quetzal Delgado Pimentel

Fecha: 03/02/2024

Código: 217239716

Materia: Traductores de lenguajes II

Tarea: Analizador Sintáctico (Implementación usando Objetos)

Introducción

En esta práctica utilizaras una pila de objetos en lugar de enteros, de esta forma al momento que imprimas la pila aparecerán los símbolos de forma similar a cuando realizas el análisis manualmente.

Desarrollo

Se ha implementado una jerarquía de clases utilizando herencia y clases abstractas para representar elementos de una pila, como **Terminal**, **NoTerminal**, y **Estado**.

La clase **Pila** se encarga de gestionar una lista que actúa como una pila, con métodos para agregar, eliminar y mostrar elementos.

Se han creado objetos de las clases definidas, insertándolos en la pila y mostrando su contenido.

La estructura general del código sigue principios de programación orientada a objetos (OOP) con encapsulamiento y polimorfismo.

La función de ejemplo demuestra la manipulación básica de la pila y cómo se reflejan los cambios en su contenido.

La implementación utiliza buenas prácticas, como el uso de clases abstractas para garantizar la implementación de métodos específicos en las clases derivadas.

Resultados

```
Pila:  
Estado: 2  
No terminal: S  
Terminal: +  
Estado: 1  
Terminal: a  
Estado: 0
```

```
Pila:  
No terminal: S  
Terminal: +  
Estado: 1  
Terminal: a  
Estado: 0
```

Conclusión

En este código, se ha implementado un sistema de clases y una pila que representa elementos específicos, como terminales, no terminales y estados. Se ha seguido una estructura orientada a objetos, utilizando clases y herencia para modelar la relación entre los elementos de la pila.

El uso de una clase abstracta **ElementoPila** asegura que todas las clases derivadas implementen el método **muestra**, proporcionando coherencia y estructura al diseño. Cada clase concreta (**Terminal**, **NoTerminal**, **Estado**) tiene su propia implementación de este método.

La clase **Pila** gestiona la manipulación de la pila, proporcionando métodos para insertar, eliminar y mostrar elementos. La implementación demuestra el encapsulamiento y la interacción entre objetos de diferentes tipos en la pila.

La función de ejemplo ilustra la creación de una pila, la inserción de varios elementos y la visualización de su contenido. Además, muestra la eliminación de un elemento y cómo afecta al estado de la pila.

En resumen, este código proporciona una base sólida para representar y manipular una pila con elementos específicos en un entorno orientado a objetos, siguiendo principios de diseño y programación.