

LAB 2

DISEÑO CÓDIGO INTERMEDIO

El código de tres direcciones en el lenguaje YAPL (Yet Another Programming Language) se utiliza como una representación intermedia para las expresiones y las operaciones dentro del lenguaje. Este código se crea y manipula mediante una estructura de datos específica diseñada para gestionar de manera eficiente los temporales y las instrucciones de tres direcciones generadas durante la compilación o interpretación del código YAPL.

Estructura de Datos

Para implementar el código de tres direcciones en YAPL, se propone la siguiente estructura de datos:

Lista de Temporales: Esta lista mantiene un registro de los temporales que se han creado durante la ejecución del programa. Cada temporal se identifica por un nombre único.

Stacks para Códigos Generados: Se utiliza una pila (stack) para almacenar las instrucciones de tres direcciones generadas durante la compilación o interpretación del código YAPL. Esta pila permite mantener un registro ordenado de las instrucciones y facilita la generación del código final.

Contador para Nombres de Temporales: Un contador se utiliza para generar nombres únicos para los temporales cuando se crean nuevos. Esto asegura que no haya conflictos de nombres.

Métodos Propuestos

Se proponen las siguientes funcionalidades contenidas dentro de métodos genéricos para operar sobre la estructura de datos mencionada:

Sobre Registros Temporales:

- **Generador de Nombres Temporales:** Este método genera un nombre único para un nuevo temporal y lo devuelve para su uso en el código de tres direcciones. También se encarga de actualizar el contador de nombres de temporales.
- **Agregación de Temporales:** Agrega un nuevo temporal a la lista de temporales creados, lo que ayuda a realizar un seguimiento de los temporales disponibles.
- **Eliminación de Temporales:** Permite eliminar un temporal de la lista de temporales cuando ya no es necesario.
- **Búsqueda de Temporales:** Busca un temporal específico dentro de la lista de temporales para verificar su existencia o uso previo.

Sobre Instrucciones

- **Agregación de instrucciones:** Agrega una instrucción de tres direcciones a la pila de códigos generados.
- **Pop de instrucciones:** Retira la última instrucción de tres direcciones de la pila, lo que facilita la recuperación de las instrucciones en orden inverso, como suele requerirse en la generación de código final.

Estos métodos y la estructura de datos proporcionan una base sólida para la generación y manipulación eficiente del código de tres direcciones en YAPL, lo que facilita la implementación del compilador o intérprete para el lenguaje. La utilización de nombres únicos para temporales y la gestión de instrucciones en una pila contribuyen a la coherencia y eficacia de la generación del código intermedio.