

Escuela Politécnica Nacional

Nombre: Fernando Elicer Huilca Villagomez

Fecha: 04/05/2025

Una parte esencial del proceso de compilación es transformar el código fuente en representación intermedias (IR), que permiten analizar el comportamiento del programa de forma más abstracta y manipulable. Las IR comunes incluyen arbol de sintaxis abstracta, graficos de flujo de control (CFG) y lenguajes intermedios similares a código de tres direcciones.

Estas representaciones facilitan la aplicación de optimizaciones posteriores. El flujo de control de programa se modela usando graficas de flujo de control, donde los nodos representan bloques básicos y los arcos indican posibles transiciones de ejecución. Estos graficos son fundamentales para muchas optimizaciones, como la eliminación de código muerto o la propagación de constantes.

Se introduce la noción de optimización dirigida por el compilador, que se divide en dos grandes tipos: optimización local (dentro de un solo bloque básico) y optimización global (entre bloques básicos). A su vez, se distinguen optimizaciones independientes del objetivo (target-independent) y dependientes del objetivo (target-dependent) siendo las primeras mas generales y aplicables en fase. Uno de los aspectos más relevantes en esta sección es el analisis de flujo de datos, que permite al compilador deducir información útil como que variables estan vivas en que puntos del programa o que expresiones pueden ser reemplazadas por valores constantes.

Se estudian técnicas como el análisis hacia adelante y hacia atrás, que implican recorrer el grafo de flujo de control en diferentes direcciones dependiendo del tipo de información que se desea recolectar.

Se representan varios algoritmos para resolver estos problemas de flujo de datos mediante sistema de ecuaciones y técnicas iterativas, además de conceptos como conjunto IN y OUT, que contienen la introducción que entra o sale de cada bloque.

Otro punto importante tratado es el papel del dominator tree, una estructura que permite determinar que bloques dominan a otros.

Esta información es esencial para ciertas optimizaciones y transformaciones como la traducción de SSA (Static Single Assignment form).

Se destacan algunas optimizaciones específicas que el compilador puede aplicar para mejorar considerablemente el rendimiento del programa.

Una de ellas es la propagación de copias, que consiste en reemplazar el uso de una variable por su valor original si este no ha sido modificado, lo que puede simplificar expresiones y reducir la cantidad de accesos a memoria o registros. También se incluye la eliminación

de código redundante, que detecta y elimina instrucciones que realizan cálculos innecesarios o repiten resultados previamente obtenidos permitiendo así que el código sea más compacto y eficiente. Otra

técnica importante es la movilización de código fuera de los bloques, la cual traslada fuera del cuerpo de un bloque aquellas operaciones cuyo resultado no cambia en cada iteración.

José María López

Video Resumen

Nombre: Fernando Elías Huilca Villagómez Fecha: 04/05/2025

Computación Electrónica: Crash Course Ciencia de computación #2

Este episodio explora la transición de las computadoras electrónicas destacando los avances tecnológicos que permitieron mejorar la velocidad, fiabilidad y eficiencia de los sistemas de cómputo.

A principios del siglo XX el crecimiento de la población y complejidad de las operaciones gubernamentales y comerciales exigieron soluciones más eficientes para el procesamiento de datos. Las primeras computadoras como el Harvard Mark I, utilizaba relés electromagnéticos como componentes principales. El Mark I completado en 1944 contenía aproximadamente 765 000 componentes y 500 millas de cableado. Su rendimiento era limitado podía hacer sumas o restas por segundo mientras que una multiplicación seis segundos. En 1947 mientras operaban el Mark los técnicos encontraron una pelilla atrapada en uno de los relés. desde entonces se conoce como bug. Las necesidades llevó al desarrollo de computadoras en tubo de vacío.

En 1947, la invención del transistor por John Bardeen, Walter y William revolucionó la computación. Los transistores eran pequeños, eficientes y confiables. La adopción de transistores impulsó el desarrollo de la industria de semiconductores, especialmente en Silicon Valley. Este avance facilitó la miniaturización de dispositivos y sentó la base para la era digital moderna.

Fernando Huilca