



INSTITUTO TECNOLÓGICO DE NUEVO LEÓN

Ingeniería en sistemas computacionales

Lenguaje y Autómatas 2

Resumen Unidad 4

“Generación de código objeto”

Catedrático

Juan Pablo Rosas Baldazo

Alumno

Axel Johnary Liñan Estrada

3/05/2018

Introducción

El generador de código objeto transforma el código Intermedio optimizado en código objeto de bajo nivel. Toma código intermedio y genera Código objeto para la máquina considerada. Es la parte más próxima a la arquitectura de la Máquina.

Capítulo 1: Registros

Los registros son la memoria principal de la computadora. Existen diversos registros de propósito general y otros de uso exclusivo. Algunos registros de propósito general son utilizados para cierto tipo de funciones. Existen registros acumuladores, puntero de instrucción, de pila, etc.

Los registros son espacios físicos dentro del microprocesador con capacidad de 4 bits hasta 64 bits dependiendo del microprocesador que se emplee.

Los registros se pueden clasificar de la siguiente forma:

Registros de datos:

AX: Registro acumulador. Es el principal empleado en las operaciones aritméticas.

BX: Registro base. Se usa para indicar un desplazamiento.

CX: Registro contador. Se usa como contador en los bucles.

DX: Registro de datos.

Estos registros son de uso general y también pueden ser utilizados como registros de 8 bits, para utilizarlos como tales es necesario referirse a ellos como por ejemplo: AH y AL, que son los bytes alto (high) y bajo (low) del registro AX. Esta nomenclatura es aplicable también a los registros BX, CX y DX.

Registros de segmentos:

CS: Registro de segmento de código. Contiene la dirección de las instrucciones del programa.

DS: Registro segmento de datos. Contiene la dirección del área de memoria donde se encuentran los datos del programa.

SS: Registro segmento de pila. Contiene la dirección del segmento de pila. La pila es un espacio de memoria temporal que se usa para almacenar valores de 16 bits (palabras).

ES: Registro segmento extra. Contiene la dirección del segmento extra. Se trata de un segmento de datos adicional que se utiliza para superar la limitación de los 64Kb del segmento de datos y para hacer transferencias de datos entre segmentos.

Registros punteros de pila:

SP: Puntero de la pila. Contiene la dirección relativa al segmento de la pila.

BP: Puntero base. Se utiliza para fijar el puntero de pila y así poder acceder a los elementos de la pila.

Registros índices:

SI: Índice fuente.

DI: Índice destino.

Capítulo 2: Lenguaje ensamblador

El ensamblador (del inglés assembler) es un traductor de un código de bajo nivel a un código, ejecutable directamente por la máquina para la que se ha generado.

El programa lee un archivo escrito en lenguaje ensamblador y sustituye cada uno de los códigos mnemotécnicos por su equivalente código máquina. Los programas se hacen fácilmente portables de máquina a máquina y el cálculo de bifurcaciones se hace de manera fácil.

Clasificación:

- **Ensambladores básicos:** Son de muy bajo nivel, y su tarea consiste básicamente, en ofrecer nombres simbólicos a las distintas instrucciones, parámetros y cosas tales como los modos de direccionamiento
- **Ensambladores modulares, o macro ensambladores:** Descendientes de los ensambladores básicos, fueron muy populares en las décadas de los 50 y los 60, fueron antes de la generalización de los lenguajes de alto nivel. Un macroinstrucción es el equivalente a una función en un lenguaje de alto nivel.

Una de las principales ventajas del uso del ensamblador, es que se encarga de administrar de manera transparente para el usuario la creación de memoria, las bifurcaciones y el paso de parámetros. Además nos permite acceder directamente a los recursos de la máquina para un mejor desempeño.

Capítulo 3: Lenguaje máquina

El lenguaje máquina sólo es entendible por las computadoras. Se basa en una lógica binaria de 0 y 1, generalmente implementada por mecanismos eléctricos. En general el lenguaje máquina es difícil de entender para los humanos por este motivo hacemos uso de lenguajes más parecidos a los lenguajes naturales.

Características

- El lenguaje máquina realiza un conjunto de operaciones predeterminadas llamadas microoperaciones.
- Las microoperaciones sólo realizan operaciones del tipo aritmética (+, -, *, /), lógicas (AND, OR, NOT) y de control (secuencial, decisión, repetitiva).
- El lenguaje máquina es dependiente del tipo de arquitectura. Así un programa máquina para una arquitectura Intel x86 no se ejecutará en una arquitectura Power PC de IBM (al menos de manera nativa).

- Algunos microprocesadores implementan más funcionalidades llamado CISC, pero son más lentos que los RISC ya que estos tienen registros más grandes.

Capítulo 4: Administración de memoria

La administración de la memoria es un proceso hoy en día muy importante, de tal modo que su mal o buen uso tiene una acción directa sobre el desempeño de memoria. En general un ensamblador tiene un administrador de memoria más limitado que un compilador; en la mayoría de los lenguajes de programación el uso de punteros no estaba vigilado por lo que se tienen muchos problemas con el uso de memoria. Los lenguajes más recientes controlan el uso de punteros y tienen un programa denominado recolector de basura que se encarga de limpiar la memoria no utilizada mejorando el desempeño.

La memoria principal puede ser considerada como un arreglo lineal de localidades de almacenamiento de un byte de tamaño. Cada localidad de almacenamiento tiene asignada una dirección que la identifica.

Conclusiones

Existen lenguajes de alto nivel, intermedio y bajo. El alto nivel es el que usan los programadores la mayoría de las veces como Java, el intermedio es el ensamblador, que además de programar instrucciones también nos da la opción de jugar con el hardware y por último el bajo nivel que es totalmente binario y es muy difícil que un humano programe en él sin cometer algún error.

La generación de un código objeto ayuda a la maquina entender las instrucciones que nosotros hayamos codificado, en lenguaje de bajo nivel para que se ejecute correctamente.

Conceptos

Objeto: es una unidad dentro de un programa de computadora que consta de un estado y de un comportamiento, que a su vez constan respectivamente de datos almacenados y de tareas realizables durante el tiempo de ejecución.

Bajo Nivel: actúa de una manera directa en comunicación con el hardware.

Mnemotécnico: sistema sencillo utilizado para recordar una secuencia de datos, nombres, números, y en general para recordar listas de ítems que no pueden recordarse fácilmente.

Bifurcaciones: es la creación de un proyecto en una dirección distinta de la principal u oficial tomando el código fuente del proyecto ya existente.

Puntero: es un objeto del lenguaje de programación, cuyo valor se refiere a (o "apunta a") otro valor almacenado en otra parte de la memoria del ordenador utilizando su dirección.

Bibliografía o Referencias

Unidad IV: Generación de código objeto, descrita en

<http://itpn.mx/recursosisc/7semestre/leguajesyautomatas2/Unidad%20IV.pdf>

Generador de código objeto Unidad 4, descrita en

<http://acaurio.blogspot.mx/2016/11/unidad-4-generacion-de-codigo-objeto.html>

Puntero (informática), descrita en

[https://es.wikipedia.org/wiki/Puntero_\(inform%C3%A1tica\)](https://es.wikipedia.org/wiki/Puntero_(inform%C3%A1tica))

Código mnemotécnico, descrita en

https://es.wikipedia.org/wiki/C%C3%B3digo_mnemot%C3%A9cnico

Bifurcación (desarrollo de software), descrita en

[https://es.wikipedia.org/wiki/Bifurcaci%C3%B3n_\(desarrollo_de_software\)](https://es.wikipedia.org/wiki/Bifurcaci%C3%B3n_(desarrollo_de_software))

Objeto (programación), descrita en

[https://es.wikipedia.org/wiki/Objeto_\(programaci%C3%B3n\)](https://es.wikipedia.org/wiki/Objeto_(programaci%C3%B3n))

Lenguaje de bajo nivel, características y funciones, descrita en

<https://www.universidadviu.es/lenguaje-nivel-caracteristicas-funciones/>

Reporte

El código objeto es el que transforma el código intermedio a código bajo nivel para que la computadora pueda ejecutarlo correctamente. Existen varios niveles, el alto nivel, el ensamblador, que se podría considerar intermedio y el bajo nivel.

El alto nivel es donde solemos programar nosotros, ya sea Java, C++, Python, etc. El ensamblador es un traductor del lenguaje de bajo nivel. Con ensamblador podemos hacer programas pero con la diferencia de que podemos definir los registros de memoria que vamos a utilizar. Esto nos permite jugar con el hardware para una mejor optimización, ya que nosotros podemos asignar cada uno de los registros de memoria necesarios para que funcione un programa. Y el lenguaje de bajo nivel es el lenguaje máquina, el lenguaje máquina es binario, donde solo se usan 0 y 1. El lenguaje de bajo nivel es perfectamente entendible para una computadora pero no para los humanos.

Los registros son espacios de la memoria principal que se usan para asignar algún valor, aunque algunos registros que se usan específicamente para una función. Los lenguajes modernos usan punteros para mejorar la administración de memoria, para no guardar varias variables con mismos valores y solo referenciarlos hacia donde están guardados esos valores.