

Informe Tarea 4

Arquitectura de Computadores

Primer Semestre de 2014

Información del Documento

HISTORIA DEL DOCUMENTO			
Nombre del Documento:	Informe Tarea 4		
Preparado por:	Ignacio Tolosa, Daniel Morales		
Responsable del Documento:	Ignacio Tolosa Daniel Morales	Fecha de Creación:	06-Jul-2014
Aprobado por:		Fecha de Aprobación:	

CONTROL DE VERSIONES			
Versión:	Fecha de Vigencia:	Preparado por:	Descripción:
1.0	06-Jul-2014	Ignacio Tolosa	Creación y primera versión del documento.
2.0	07-Jul-2014	Daniel Morales	Revisión y últimos detalles del documento

Resumen Ejecutivo

El presente documento es el informe correspondiente a la cuarta tarea del curso de Arquitectura de Computadores de la carrera de Ingeniería Civil en Informática de la UTFSM.

El documento se divide en varios ejercicios que se debían resolver en la tarea. De los cuales se busca responder la siguiente interrogante: “¿A qué se deben las diferencias entre el código Assembly generado por un compilador y el creado?”

El resultado de cada ejercicio fue implementado en un software externo, por lo que, junto con el presente documento, se adjuntan estos trabajos.

Tabla de Contenido

Resumen Ejecutivo.....	3
Tabla de Contenido.....	4
1. Introducción.....	5
2. Resultados.....	6
3. Conclusiones.....	7
4. Bibliografía.....	8
Anexo I: Acrónimos.....	9
Anexo II: Glosario.....	10

1. Introducción

La tarea consiste en hacer 2 ejercicios que deben ser resueltos utilizando dos herramientas muy utilizadas en el ámbito de la ingeniería, las cuales son el lenguaje C y su análogo en bajo nivel el lenguaje

Assembly. Todos los ejercicios se debieron realizar utilizando el método de la recursión de cola, el cual consiste en guardar resultados intermedios de cada iteración en un parámetro que es pasado a la función llamada internamente, con el objetivo de liberar y optimizar la carga y funcionamiento del procesador y la memoria.

Las pruebas fueron realizadas en los sistemas operativos Linux y UNIX (OS X), utilizando para ello las herramientas GCC para la compilación del código en el lenguaje C y para la generación de lenguaje Assembly se utilizó la herramienta QTSpm (emulador de los procesadores MIPS).

2. Resultados

A continuación se presentan los resultados, en donde se señala la diferencia que existe entre los dos códigos Assembly generados tanto por el compilador GCC y los contruidos manualmente.

El código Assembly que compila GCC es más extenso que el construido manualmente debido a que el que se construyó manualmente se hizo de manera ad-hoc, es decir, hubo un conocimiento mayor del funcionamiento que debía cumplir el programa, por el otro lado, el compilador al no tener esta noción, intenta ser lo más general posible al traducir o compilar el código de C a Assembly, por lo que se puede notar, que existen muchas redundancias en el código final, además de diversas secciones de código que son específicas del sistema operativo en que es compilado, de la arquitectura en específico para el procesador en cuestión, y de diversas instrucciones que proveen de mayor compatibilidad al código al momento de ser ejecutado en el sistema manteniendo el ambiente de integridad.

3. Conclusiones

Mediante el siguiente trabajo se logró entender la forma en que trabajan los compiladores al transformar código a máquina y además que el código Assembly es más óptimo si se crea manualmente, en otras palabras, es mejor crear el software para que se ejecute rápidamente en Assembly que en lenguajes como C.

4. Bibliografía

1. Apuntes del ramo Arquitectura de Computadores. Prof. Mauricio Solar.

Anexo I: Acrónimos

GCC	GNU Compiler Collection (colección de compilador GNU).
UNIX	Sistema operativo propiedad de los laboratorios Bell
LINUX	Sistema operativo de código abierto similar a los sistemas operativos UNIX

Anexo II: Glosario

Mapa de Karnaugh	Mapa utilizado para simplificar ecuaciones. Representa el cuerpo de un “torito” en tercera dimensión, donde en su superficie existen valores transversales y adyacentes entre sí.
Circuito Combinacional	Circuito compuesto por compuertas lógicas
Compuertas Lógicas	Compuerta que recibe entradas y que luego de aplicar una operación booleana retorna una salida exclusiva.
Circuito Secuencial	Circuito compuesto por una memoria y un componente combinacional