

Instituto tecnológico de Costa Rica

Unidad de Computación

Tarea Corta ASM

Steven Chacón

Leiner Alvarado

Melany Villalobos

Sede San Carlos

28/5/2023

Introducción

El presente trabajo tiene como objetivo principal el desarrollo y la implementación de ejercicios en el entorno Linux, utilizando los lenguajes de programación C++ y ensamblador (ASM). Estas dos herramientas son ampliamente utilizadas en el ámbito de la programación de sistemas y ofrecen un alto grado de flexibilidad y control sobre el hardware..

Linux es un sistema operativo de código abierto que se ha convertido en una plataforma popular para el desarrollo de software, debido a su estabilidad, seguridad y capacidad de personalización. Además, ofrece una amplia gama de herramientas y bibliotecas que facilitan la programación y la depuración de aplicaciones.

C++ es un lenguaje de programación de propósito general conocido por su eficiencia y su capacidad para trabajar a bajo nivel. Proporciona características avanzadas, como la orientación a objetos, la gestión de memoria y la manipulación de estructuras de datos complejas, lo que lo convierte en una elección popular para el desarrollo de aplicaciones de alto rendimiento.

El lenguaje ensamblador (ASM) es un lenguaje de programación de bajo nivel que permite interactuar directamente con el hardware de un sistema. Al utilizar ASM, se puede acceder a instrucciones específicas del procesador y optimizar el rendimiento de las aplicaciones. Este lenguaje es especialmente útil en casos donde se requiere un control detallado sobre la ejecución de los programas y el aprovechamiento de los recursos de hardware disponibles.

A lo largo de este trabajo, se explorarán diferentes ejercicios que permitirán afianzar los conocimientos en programación de sistemas utilizando Linux, C++ y ASM.

Análisis del problema

El problema planteado consiste en el desarrollo de una serie de tareas en el entorno Linux utilizando los lenguajes de programación C++ y ensamblador (ASM). A continuación, se realiza un análisis detallado de las tareas y requerimientos planteados:

1. Implementación de una interfaz de comandos

- Se debe desarrollar una interfaz que permita al usuario ejecutar las distintas funciones solicitadas a través de comandos.
- La interfaz debe ser interactiva y permitir una interacción fluida con el usuario, solicitando los parámetros necesarios para cada tarea y mostrando los resultados correspondientes.

2. Funciones de consulta de hora del sistema

- Se debe implementar una función que obtenga la hora actual del sistema en formato "HH:MM:SS".
- Para esto, es necesario utilizar las funciones proporcionadas por el sistema operativo para obtener la hora actual y formatearla de acuerdo al formato requerido.

3. Funciones de operaciones aritméticas

- Se deben implementar funciones para realizar operaciones aritméticas, considerando números de punto flotantes.
- Las operaciones a implementar incluyen suma, resta, división, multiplicación y potencia.
- Es necesario permitir la utilización de más de 2 operadores en una misma operación.
- Para las operaciones combinadas, se deben seguir las reglas de orden de ejecución y utilizar paréntesis para agrupar operaciones.

4. Función para calcular el factorial de un número

- Se debe implementar una función que calcule el factorial de un número ingresado por el usuario.
- El cálculo del factorial se realiza mediante multiplicaciones sucesivas.

5. Funciones de conversión de unidades

- Se deben implementar funciones para realizar conversiones entre diferentes sistemas numéricos, unidades de longitud, masa, monedas y temperatura.
- Para las conversiones de sistemas numéricos, se deben implementar conversiones entre decimal y binario, decimal y complemento base dos, y decimal e IEEE 754.
- Para las conversiones de longitud, se deben permitir conversiones entre metros, kilómetros, millas, pulgadas y pies.

- Para las conversiones de masa, se deben permitir conversiones entre gramos, kilogramos y libras.
- Para las conversiones de monedas, se deben permitir conversiones entre colón, euro y dólar, y se debe implementar un mecanismo para actualizar automáticamente el tipo de cambio utilizando un WebService o Web Scraping.
- Para las conversiones de temperatura, se deben permitir conversiones entre Celsius, Fahrenheit y Kelvin.

El análisis del problema identifica las principales tareas y requerimientos que deben ser abordados en el desarrollo del proyecto. Cada tarea implica implementar funciones específicas utilizando los lenguajes de programación C++ y ensamblador (ASM), y se requiere un manejo adecuado de las estructuras de datos y algoritmos necesarios para realizar las operaciones solicitadas.

Además, se destaca la importancia de trabajar en el entorno Linux, lo cual implica comprender su funcionamiento y utilizar las herramientas y bibliotecas adecuadas para interactuar con el sistema operativo y realizar las tareas requeridas.

Solución del problema

La solución propuesta para el problema planteado implica la implementación de funciones en C++ y ensamblador (ASM) para realizar las tareas solicitadas. A continuación, se presenta una descripción de la solución teniendo en cuenta los requerimientos específicos:

1. Implementación de la interfaz de comandos
 - Se desarrolla una interfaz de línea de comandos en C++ que permite al usuario seleccionar las diferentes funciones y opciones del programa.
 - El usuario puede interactuar con el programa ingresando comandos y parámetros específicos para ejecutar las tareas deseadas.
2. Funciones de operaciones aritméticas
 - Se implementan las operaciones aritméticas en C++ utilizando el código proporcionado por el docente.
3. Función para calcular el factorial de un número

- Se implementa una función en C++ que recibe un número como parámetro y calcula su factorial utilizando un bucle y multiplicaciones sucesivas.

4. Funciones de conversión de unidades

- Para las conversiones de unidades, se utiliza un mapa en C++ que almacena los valores de equivalencia entre las diferentes unidades.
- El usuario puede seleccionar la conversión deseada a través del menú de opciones.
- Para las conversiones de unidades que requieren ensamblador, se implementa una función en ensamblador que realice la multiplicación utilizando los valores de equivalencia correspondientes y retorne el resultado.

5. Funciones de conversión de temperaturas

- Se implementan las fórmulas de conversión de Celsius a Fahrenheit y Kelvin utilizando variables de punto flotante y los cálculos correspondientes

En resumen, la solución del problema implica desarrollar funciones en C++ y ensamblador para realizar las tareas solicitadas. Se utilizan estructuras de control y algoritmos adecuados para implementar cada función específica, teniendo en cuenta los requerimientos y considerando el uso de ensamblador para operaciones que requieran un rendimiento optimizado.

Análisis de resultados

Tarea	Estado	Observaciones
Interfaz	Completo	n/a
Consultar hora del sistema	Incompleto	No se realizó
Operaciones aritméticas	Parcialmente completo	No se realizaron las operaciones combinadas
Factorial de un número	Completo	n/a
Conversión de unidades	Parcialmente completo	No se realizó la solución de sistemas numéricos

Conclusiones

El lenguaje ensamblador es un lenguaje complicado y de bajo nivel que requiere un mayor nivel de conocimiento y comprensión del hardware subyacente. Sin embargo, a pesar de su complejidad, es importante familiarizarse con él, ya que puede ser de gran utilidad en el desarrollo de sistemas desde cero o en situaciones donde se requiera un control preciso del hardware.

En el presente proyecto, aunque se utilizó principalmente el lenguaje C++ para implementar las funciones y operaciones, la integración del lenguaje ensamblador en ciertas partes del código ha demostrado su compatibilidad y sinergia con lenguajes de alto nivel. La combinación de ambos lenguajes puede conducir a soluciones más eficientes y optimizadas, aprovechando al máximo el rendimiento del hardware subyacente.

Además, es fundamental destacar la importancia del desarrollo en el entorno Linux. Linux es ampliamente utilizado por los desarrolladores debido a su estabilidad, flexibilidad y capacidad de personalización. Aprender a trabajar en este entorno brinda una base sólida para desarrollar aplicaciones robustas y escalables.

En resumen, el desarrollo de ejercicios en el entorno Linux utilizando C++ y ASM ha brindado la oportunidad de adquirir un conocimiento profundo en la programación de sistemas y la optimización del software. Estas habilidades son altamente valoradas en el campo de la informática y sientan las bases para enfrentar tareas más complejas en el futuro. Al comprender el funcionamiento interno del sistema operativo y poder aprovechar al máximo los recursos del hardware, los desarrolladores pueden crear soluciones eficientes y de alto rendimiento que satisfagan las necesidades de los usuarios y contribuyan al avance tecnológico.

Bibliografía

Barbone, V. G. (s/f). *El compilador GCC - Tutorial*. Edu.uy. Recuperado el 29 de mayo de 2023, de <https://iie.fing.edu.uy/~vagonbar/gcc-make/gcc.htm>

Pk, J. (2006, octubre 15). *Using inline assembly in C/C++*. CodeProject.
<https://www.codeproject.com/Articles/15971/Using-Inline-Assembly-in-C-C>