

Contratos - Alineación y convención de llamada

Organización del Computador II

Segundo Cuatrimestre 2021

En este taller vamos a trabajar con código **C** y **ASM** para ejercitar el uso y adhesión a contratos estructurales y comportamentales. Algunos ejercicios deben completarlos en el código que descargaron para el taller y otros les van a pedir que den una explicación a algún tema específico, en ambos casos intenten expresar con el mayor nivel de detalle y claridad sus respuestas.

Al corregir cada **checkpoint** todos los miembros del grupo deben estar presentes, salvo aquellas instancias en que puedan justificar su ausencia previamente. Si al finalizar la práctica de la materia algún miembro cuenta con mas de un 20 % de ausencia no justificada en la totalidad de los checkpoints se considerará la cursada como desaprobada.

Cada checkpoint se presenta a lx docente asignadx en cada breakout room durante el transcurso de la clase práctica actual o la siguiente, no hace falta entregar los archivos.

1. Conceptos generales

En este checkpoint deberán responder algunas preguntas conceptuales relacionadas a lo que vimos en las clases teóricas y en la presentación de hoy. Las preguntas tienen que ver con **alineación de datos, convención de llamada y bibliotecas compartidas**.

- a) ¿Cómo se alinea un atributo de un **struct** no empaquetado en memoria? ¿Cómo calcularía el desplazamiento a partir del comienzo de la estructura?
- b) ¿Cómo se calcula el tamaño de un **struct** empaquetado? ¿Y el tamaño de uno no empaquetado?
- c) ¿Qué entienden por convención de llamada? ¿Cómo está definida la convención **C** para 64 y 32 bits?
- d) ¿Quién toma la responsabilidad de asegurar que se cumple la convención de llamada en **C**? ¿Quién toma la responsabilidad de asegurar que se cumple la convención de llamada en **ASM**?
- e) ¿Qué es un stack frame? ¿A qué se le suele decir **prólogo y epílogo**?
- f) ¿Cuál es el mecanismo utilizado para almacenar **variables temporales**?
- g) ¿A cuántos bytes es necesario alinear la pila si utilizamos funciones de **libc**? ¿Si la pila está alienada a 16 bytes al realizarse una llamada función, cuál va a ser su alineamiento al ejecutar la primera instrucción de la función llamada?
- h) Vimos que la convención de llamada nos permite interactuar con bibliotecas binarias que respeten el contrato. ¿Qué sucede si una función que se ubica en un código objeto utilizado por nuestro programa (**biblioteca compartida**) cambia su interfaz (parámetros o tipo devuelto) en una actualización del sistema operativo?

Checkpoint 1

2. Estudiando casos de interacción C/ASM

En este checkpoint vamos a estudiar y explicar funciones implementadas en **ASM** que son llamadas desde código **C** haciendo uso de la convención de llamada de 64 bits. Pueden encontrar el código vinculado a este checkpoint en `checkpoint2.c` y `checkpoint2.asm`, la declaración de las funciones se encuentra en `checkpoints.h`.

- Expliquen el funcionamiento de `alternate_sum_4`, `sum_z`, `packed_sum_z`, `product_2` y `product_2_f`.
- ¿Cuál sería la metodología general que utilizarían para implementar una función en **ASM** que pueda ser llamada desde **C**?

Checkpoint 2

3. Ejercicios de interacción C/ASM

En este checkpoint vamos a estudiar e implementar porciones de código vinculado a la inicialización de arreglos, ya sea de forma dinámica o estática. Pueden encontrar el código vinculado a este checkpoint en `checkpoint3.c` y `checkpoint3.asm`, la declaración de las funciones se encuentra en `checkpoints.h`. La ubicación del código que debe completarse se encuentra indicada con la frase **COMPLETAR:**, recuerden retirar los caracteres de comentario simple `//` (en C), `;` (en ASM) o múltiple `/* */`.

- a) Completen la definición de constantes de `complex_sum_z` en `checkpoint3.asm` y escriban las insrucciones faltantes para acceder al atributo indicado y recorrer el arreglo de estructuras de forma consistente con el tamaño de las mismas.
- b) Completen la definición de constantes de `packed_complex_sum_z` en `checkpoint3.asm` y escriban las insrucciones faltantes para acceder al atributo indicado y recorrer el arreglo de estructuras empaquetadas de forma consistente con el tamaño de las mismas.
- c) Indiquen en que registro o posición relativa a `RBP` se encuentran los parámetros de `product_9_f`.
- d) Completen las conversiones de tipo y multiplicaciones de `product_9_f`.

Checkpoint 3