

Trabajo de clase

Descripción general

A continuación se definen los lineamientos de una actividad teórico-práctica para los estudiantes del curso sistemas operativos, en particular, el direccionamiento de memoria.

Objetivos

Durante el desarrollo de las actividades se logra:

- Identificar la memoria física.
- Aplicar los conocimientos adquiridos en la programación en C++

Antes de empezar

- Verifique la instalación de docker y de la imagen GCC o PUJGCC

En este trabajo de clase, se explorará el concepto de direccionamiento de memoria en C++. A través de una serie de ejercicios, practicarás cómo trabajar con variables, punteros y direcciones de memoria para acceder y modificar datos en la memoria de la computadora.

Actividad No. 1

1. Cree un programa en C++, declare y asigne una variable entera (int)
2. Presente en pantalla la dirección de memoria de esa variable.
3. Modifique el valor de la variable indirectamente utilizando punteros
4. Presente nuevamente el valor de la variable y la dirección de memoria

Actividad No. 2

1. Cree un programa en C++, declare un puntero a una variable .
2. Utilice el puntero para acceder y modificar el valor de la variable.
3. Cree una referencia a la variable y utilicela para modificar el valor
4. Presente las direcciones de memoria del puntero y la referencia

Actividad No. 3

1. Cree un programa en C++, declare un array de números enteros (int)
2. Utilice punteros para acceder a los elementos del array y modificar su contenido.
3. Presente la dirección de memoria del array y del puntero

Actividad No. 4

1. Cree un programa que use la asignación dinámica de memoria con *new*
2. Cree una matriz de enteros dinámica de 2D.
3. Llenen la matriz con datos
4. Libere la memoria con *delete* cuando haya terminado.

Extra: Presente el stack, el heap y el code usando C++

Entrega: Aloje el código en un repositorio público de Github y envíe un informe de la implementación y las salidas a través del aula digital.