

Diferencias entre struct y unión:

Struct: El compilador asigna una dirección de memoria para cada elemento asignado dentro del struct, como resultado la cantidad de memoria total ocupada por la estructura es igual a la suma del tamaño de sus elementos. Al alterar el valor de uno de sus elementos no afecta a otros dentro de la estructura. Permite inicializar varios elementos de la estructura a la vez.

Union: El compilador asigna una dirección de memoria considerando el tamaño que ocupa el elemento de mayor tamaño, como resultado la cantidad de memoria total ocupada por la unión es igual al tamaño ocupado por su elemento de mayor tamaño. Alterar el valor de uno de los elementos afecta a otros dentro de la unión. Solo permite inicializar el primer elemento de la unión.

12. Dado el siguiente tipo de datos y teniendo en cuenta que la dirección del miembro a es

0x00546334, ¿cuál sería la dirección de b?

*union aux {*

*int a;*

*char b;*

*}*

La dirección ocupada por b es 0x00546334

13. Cuál es el tamaño de la union del punto anterior?

El tamaño ocupado es de 4 bytes

14. Dado el siguiente código, ¿qué se mostraría por pantalla al correrlo?

*#include <stdio.h>*

*union aux {*

*int a;*

*char b;*

*};*

*int main(void){*

*union aux var;*

*var.a = 77;*

*printf("a: %d\n", var.a);*

*printf("b: %c\n", var.b);*

*}*

a: 77

b: M