

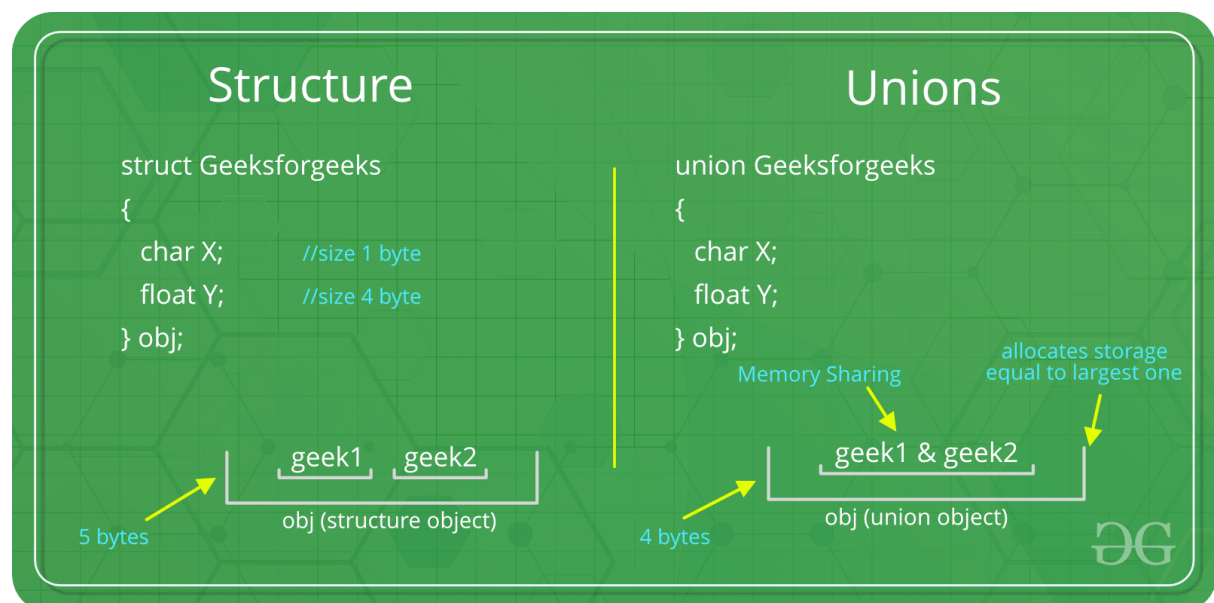
What Are The Major Difference Between Structure And Union

C provides five methods for creating custom data. These are "bit-field," "union," "structure," "enumeration," and "typedef." The purpose of this essay is to shed additional light on the key differences between structure and union in C. Continue reading to learn more about [difference between structure and union](#).

Union and structure are container data types in C that can hold any data. In C, an essential distinction between structures and unions is that structures have a separate memory address for each member. Members of a union, on the other hand, share the same memory location.

Structure is a datatype that has been defined by the user. It is used to aggregate many data kinds into a single type. It might have several members as well as structure variables. Structures are defined in C using the keyword "struct." The dot(.) operator can be used to access structure members regarding difference between structure and union.

Union is a user-defined datatype as well. The memory location is shared by all union members. The size of the union is determined by the size of the union's largest member. Union is the ideal option if you wish to use the same memory space for two or more members.



Unions are comparable to the structure. Union variables are constructed in the same way that structure variables are. In C, unions are defined with the keyword "union."

Because a union shares memory space among its members, there is no need to assign memory to all members. Shared memory space is assigned, which is equal to the size of the

member with the most memory. Structure members do not exchange memory. As a result, a structure requires independent memory space for each of its components, implying that each member has unique storage concerning difference between structure and union.

The keyword 'union' is used to define Union.

The keyword 'struct' is used to define Structure.

Initialization. Only the first Union member can be initiated. All structure members can be initialised.

Size.

The size of the union is equal to the size of the largest member. The structure's size is more than the sum of the sizes of its members.

In order to access structure members in C, the structure variable must be declared. By default, all structure members are "public." The structural members were all preserved in nearby memory regions. Under a single name, the structure type variable can hold many data objects of differing data kinds.

A structure's declaration is always preceded by the keyword "struct," which informs the compiler that the structure has been declared. The dot(.) operator can be used to acquire structure members. The struct statement creates a new data type with more than one member regarding difference between structure and union.

A union is another of the user-defined data types available in C and C++. A union, like a structure, can store several variables of different data kinds. However, in a union, each data member stated is not allocated memory individually.

We save extra memory by allocating memory of the largest size data variable as the union's memory. When we edit a data member, all other data members with the same data type are updated with the same new value, or a garbage value is assigned to them. You must know this regarding difference between structure and union.

Structures tie together multiple pieces of data regarding the same subject in the same location.

It is useful when gathering data of comparable data types and attributes, such as first name, last name, and so on. It is incredibly simple to maintain because we can represent the entire record with a single name.

Union takes up less memory than structure.

Only the last variable can be directly accessed while using union. Union is utilised when two or more data members must share the same memory location. It allows you to store data from only one data member. Its assigned space is equal to the data member's maximum size about difference between structure and union.