What is the Difference Between Structure Union and Enum in C ?

- Structure is a data type that stores different data types in the same memory location; the total memory size of the structure is the summation of memory sizes of all its members.
- In contrast, Union is a data type that stores different data types in the same memory location; the total memory size depends on the memory size of its largest elements.
- Meanwhile, Enum is a data type that store integral constants. That is the main difference between structure union and enum in C.

What is Structure ?

- A structure is a single variable that can hold data of multiple types. It is a set of variables of dissimilar data types.
- Variable declaration : `struct structureName structureVariable;`
- Accessing Memebers of a Structure : `structureVariable.memberVariable`
- Variable Initialization :`struct structureName = { value1, value2,...};`

Example:

```
typedef struct{

        char name[20];
        int roll;
        char gender;
        int marks[5];
    }STUDENT;
```

```
void main(){
STUDENT st1 = { "Alex", 43, 'M', {76, 78, 56, 98, 92}};
STUDENT st2 = { "Max", 33, 'M', {87, 84, 82, 96, 78}};
}
printf("Name: %s\n", st1.name);
 printf("Roll: %d\n", st1.roll);
 printf("Gender: %c\n", st1.gender);
 for( int i = 0; i < 5; i++)
    printf("Marks in %dth subject: %d\n", i, st1.marks[i]);
```

Use of structure:

Structures (also called structs) are a way to group several related variables into one place. Each variable in the structure is known as a member of the structure. Unlike an array, a structure can contain many different data types (int, float, char, etc.).

What is Union ?

Union allows storing various data types in the same memory location. For example, an Employee can have properties such as name, salary, and, city. Instead of creating variables for each of them, it is possible to use a union. It compacts all different data types into a single unit.

Example:

```
union Data {
  int i;
  float f;
  char str[20];
};

int main( ) {

  union Data data;

  data.i = 10;
```

```
   data.f = 220.5;
   strcpy( data.str, "C Programming");

   printf( "data.i : %d\n", data.i);
   printf( "data.f : %f\n", data.f);
   printf( "data.str : %s\n", data.str);
```

Output:

data.i : 1917853763

data.f : 4122360580327794860452759994368.000000

```
data.str : C Programming
```

Here, we can see that the values of i and f members of union got corrupted because the final value assigned to the variable has occupied the memory location and this is the reason that the value of str member is getting printed very well.The same example once again where we will use one variable at a time which is the main purpose of having unions −

```
union Data {
   int i;
   float f;
   char str[20];
};
int main( ) {

   union Data data;

   data.i = 10;
   printf( "data.i : %d\n", data.i);

   data.f = 220.5;
   printf( "data.f : %f\n", data.f);
   strcpy( data.str, "C Programming");
   printf( "data.str : %s\n", data.str);
```

Output:

```
data.i : 10
data.f : 220.500000

data.str : C Programming
```

Here, all the members are getting printed very well because one member is being used at a time.

Use of union:

A union is a special data type available in C that allows to store different data types in the same memory location. You can define a union with many members, but only one member can contain a value at any given time. Unions provide an efficient way of using the same memory location for multiple-purpose.

What is Enum ?

Enum stands for enumeration. It is a user-defined data type that consists of integral constants.

Example:

```
enum weekdays{Sunday=1,Monday,Tuesday,Wednesday,Thursday,Friday,Saturday};
int main()
{
enum weekdays w;//variable declaration of weekdays type
w=Monday;//assigning value of Monday to w.
printf("The value of w is %d",w);
```

In the above code, we create an enum type named as weekdays, and it contains the name of all the seven days. We have assigned 1 value to the Sunday, and all other names will be given a value as the previous value plus one.

Use of enum:

The use of enum in C to name the integer values makes the entire program easy to learn, understand, and maintain by the same or even different programmer.

Keyword

The keyword to declare a Structure is ‘struct’ while the keyword to declare a Union is ‘union’, and the keyword to declare an Enum is ‘enum’.

Usage

There is a difference between structure union and enum in C based on their usage as well. Both structure and union help to store data of different types as a single unit while enum helps to assign constants to a set of names to make the program easier to read, maintain and understand.

Conclusion

Structure is a data type that stores different data types in the same memory location; the total memory size of the structure is the summation of memory sizes of all its members. In contrast, Union is a data type that stores different data types in the same memory location; the total memory size depends on the memory size of its largest elements. Meanwhile, Enum is a data type that store integral constants. Thus, this is the main difference between structure union and enum in C.