SNS COLLEGE OF TECHNOLOGY

Coimbatore-35
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF AIML

23ITT101-PROGRAMMING IN C AND DATA STRUCTURES

I YEAR - II SEM

UNIT 3 – ARRAYS AND INTRODUCTION TO DATA STRUCTURES

TOPIC 4 –Structures and Union

INTRODUCTION

- **Structure** is another user defined data type available in C that allows to combine data items of different kinds.

- Defining a Structure

```
struct [structure tag]
{
member definition;
member definition;
...
member definition;
} [one or more structure variables];
```

Structures Example

```
struct Books
{
char title[50];
char author[50];
char subject[100];
int book_id;
} book;
```

UNDERSTANDING STRUCTURE VARIABLE

```
struct Books
{ char title[50];50
 char author[50];50
 char subject[100];100
 int book_id; };4 204

int main( ) {
struct Books Book1; /* Declare Book1 of type Book */
struct Books Book2; /* Declare Book2 of type Book */
}
```

Accessing Structure Variable

- /* book 1 specification */

```
scanf(“%s”,& Book1.title);
scanf(“%s”,& Book1.author);
scanf(“%s”,& Book1.subject);
Book1.book_id = 6495407;

strcpy( Book1.title, “C Programming");
strcpy( Book1.author, “Nuha Ali");
 strcpy( Book1.subject, “C ProgrammingTutorial");
Book1.book_id = 6495407;
```

Accessing Structure Variable

- /* book 2 specification */

```
scanf(“%s”,& Book2.title);
scanf(“%s”,& Book2.author);
scanf(“%s”,& Book2.subject);
Book2.book_id = 6495407;

strcpy( Book2.title, "Telecom Billing");
strcpy( Book2.author, "Zara Ali");
 strcpy( Book2.subject, "Telecom Billing Tutorial");
Book2.book_id = 6495700.
```

Printing Structure Variable

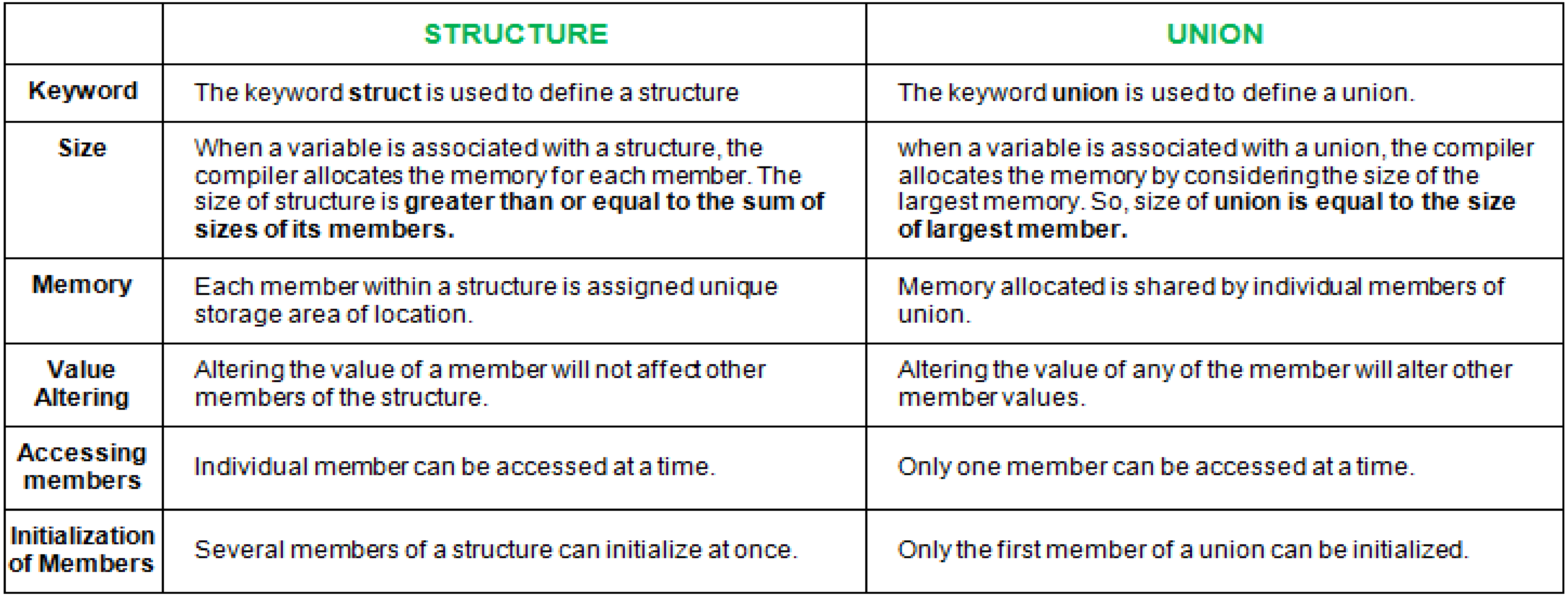
```
/* print Book1 info */
printf( "Book 1 title : %s\n", Book1.title);
printf( "Book 1 author : %s\n", Book1.author);
printf( "Book 1 subject : %s\n", Book1.subject);
printf( "Book 1 book_id : %d\n", Book1.book_id);

/* print Book2 info */
printf( "Book 2 title : %s\n", Book2.title);
printf( "Book 2 author : %s\n", Book2.author);
printf( "Book 2 subject : %s\n", Book2.subject);
 printf( "Book 2 book_id : %d\n", Book2.book_id);
```

UNDERSTANDING UNION

- A **union** is a special data type available in C that allows to store different data types in the same memory location.
- we 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.

Structure Vs Union



	STRUCTURE	UNION
Keyword	The keyword **struct** is used to define a structure	The keyword **union** is used to define a union.
Size	When a variable is associated with a structure, the compiler allocates the memory for each member. The size of structure is **greater than or equal to the sum of sizes of its members.**	when a variable is associated with a union, the compiler allocates the memory by consideringthe size of the largest memory. So, size of **union is equal to the size of largest member.**
Memory	Each member within a structure is assigned unique storage area of location.	Memory allocated is shared by individual members of union.
Value Altering	Altering the value of a member will not affect other members of the structure.	Altering the value of any of the member will alter other member values.
Accessing members	Individual member can be accessed at a time.	Only one member can be accessed at a time.
Initialization of Members	Several members of a structure can initialize at once.	Only the first member of a union can be initialized.