UNIT - IV: Structure and Pointers

Author : Computer Science Department **Date :** 2025

Structures

A structure is a user-defined data type that groups related variables of different types.
Syntax: struct student {int roll_no; char name[50];};

Structure Declaration and Initialization

- Declaring a structure variable: struct student s1;
- Initializing a structure: struct student s1 = {101, 'John'};

Structure within a Structure

Also known as nested structures. Example:

```
struct department {
int dept_id;
struct student s1;
};
```

Self-Referential Structures

A structure that contains a pointer to itself. Used in linked lists and trees.

Pointers

A pointer is a variable that stores the address of another variable.
Syntax: int *ptr;
Pointer Arithmetic: ptr++, ptr--, ptr+2, ptr-2.

Pointers and Arrays

An array name acts as a pointer. Example:

```
int arr[5] = {1,2,3,4,5};
int *ptr = arr; // ptr points to arr[0]
```

Pointer to Function

A pointer that stores the address of a function. Used for callback functions in C.
Syntax: int (*func_ptr)(int, int);

Pointer and Structure

A pointer can store the address of a structure.
Syntax: struct student *sptr;
sptr = &s1;