



# FAST

---

الذى علم بالقلم. علم الانسان ما لم يعلم.

**National University of Computer and Emerging Sciences**

DEPARTMENT OF COMPUTER SCIENCE

# Object Oriented Programming

## Lab-05



# OOP (LAB-05)

## STRUCTURE IN C++

**Engr. Khuram Shahzad**  
**(Instructor)**

# Lab 05 Content

- ☐ **Arrays**
- ☐ **Structure**

# User Defined Data Types

- **Class**

- Member of the class:
  - ✓ Variables
  - ✓ Functions

- **Structure**

- Member of Structure:
  - ✓ Variables

# What a Structure is?

- “A structure is a collection of variables under a single name. These variables can be of different types, and each has a name that is used to select it from the structure”
- There is always a requirement in most of our data processing applications that the **relevant data** should be **grouped** and **handled** as a **group**
- In structure, we introduce a **new data type**

## Some More Info.

- A structure can contain any data type including array and another structure as well.
- It provides a simple method of abstraction and grouping
- Each variable declared inside structure is called member of structure.
- A structure may itself contain structures
- A structure can be assigned to, as well as passed to, and returned from functions.
- We declare a structure using the keyword struct

## Some More Info.

- A structure can contain any data type including array and another structure as well.
- It provides a simple method of abstraction and grouping
- Each variable declared inside structure is called member of structure.
- A structure may itself contain structures
- A structure can be assigned to, as well as passed to, and returned from functions.
- We declare a structure using the keyword struct

# What a Structure is?

**Students**




- Name
- Address
- Date of Birth
- CGPA
- Discipline



# What a Structure is?

Car

- 
- Model
  - Manufacturer company
  - Engine size
  - Number of seats

# What a Structure is?

**Employee**



- Employee ID
- Name
- Department
- Date of Joining
- Salary

## Steps to Create Structure

- Declare Structure
- Initialize Members of Structure
- Access Structure Elements

# Declaration of a Structure

- **1) Declare Structure**
- **struct** keyword is used for creating a structure.
- **Structure Declaration Ways**
- By struct keyword
- By declaring variable at the time of defining structure.

## Declaration of a Structure

- The structure is declared by using the keyword `struct` followed by structure name, also called a tag. Then the structure members (variables) are defined with their type and variable names inside the open and close braces `{` and `}`.
- Finally, the closed braces end with a semicolon denoted as `;` following the statement. The above structure declaration is also called a Structure Specifier.

## Declaration of a Structure

- Structures are syntactically declare with:
  - ✓ Keyword **struct**
  - ✓ Followed by the **name of the structure**
  - ✓ The **data**, contained in the structure, is defined in the **curly braces**
- All the variables that we have been using can be part of structure

# Approach 1(Declaration of a Structure)

The diagram illustrates the components of a C structure declaration. A red arrow points from the label 'Keyword/Tag' to the word 'struct'. Another red arrow points from the label 'Structure Name' to the word 'student'. A red bracket on the right side of the code block groups the four member declarations, with a label 'Components / Members' pointing to it.

```
struct student  
{  
    char name [60];  
    char address [100];  
    char discipline [50];  
    float GPA;  
};
```

Keyword/Tag

Structure Name

Components / Members

## Declaration of a Structure

- **Student** is called the **structure tag**, and is your brand new data type, like int, double or char.
- **name, address, discipline, and GPA** are **structure members**.
- **Note:** Memory is not allocated at the time of its declaration. Memory is allocated when we declare structure variable.



```
struct student
{
    string name;
    string address;
    string discipline;
    float GPA;
};
```

```
struct car
{
    string model;
    string company;
    string engineSize;
    int price;
};
```

```
struct employee
{
    string employeeID;
    string name;
    string department;
    float salary;
};
```

## Declaration of a Structure

- **Student** is called the **structure tag**, and is your brand new data type, like int, double or char.
- **name, address, discipline, and GPA** are **structure members**.
- **Note:** Memory is not allocated at the time of its declaration. Memory is allocated when we declare structure variable.

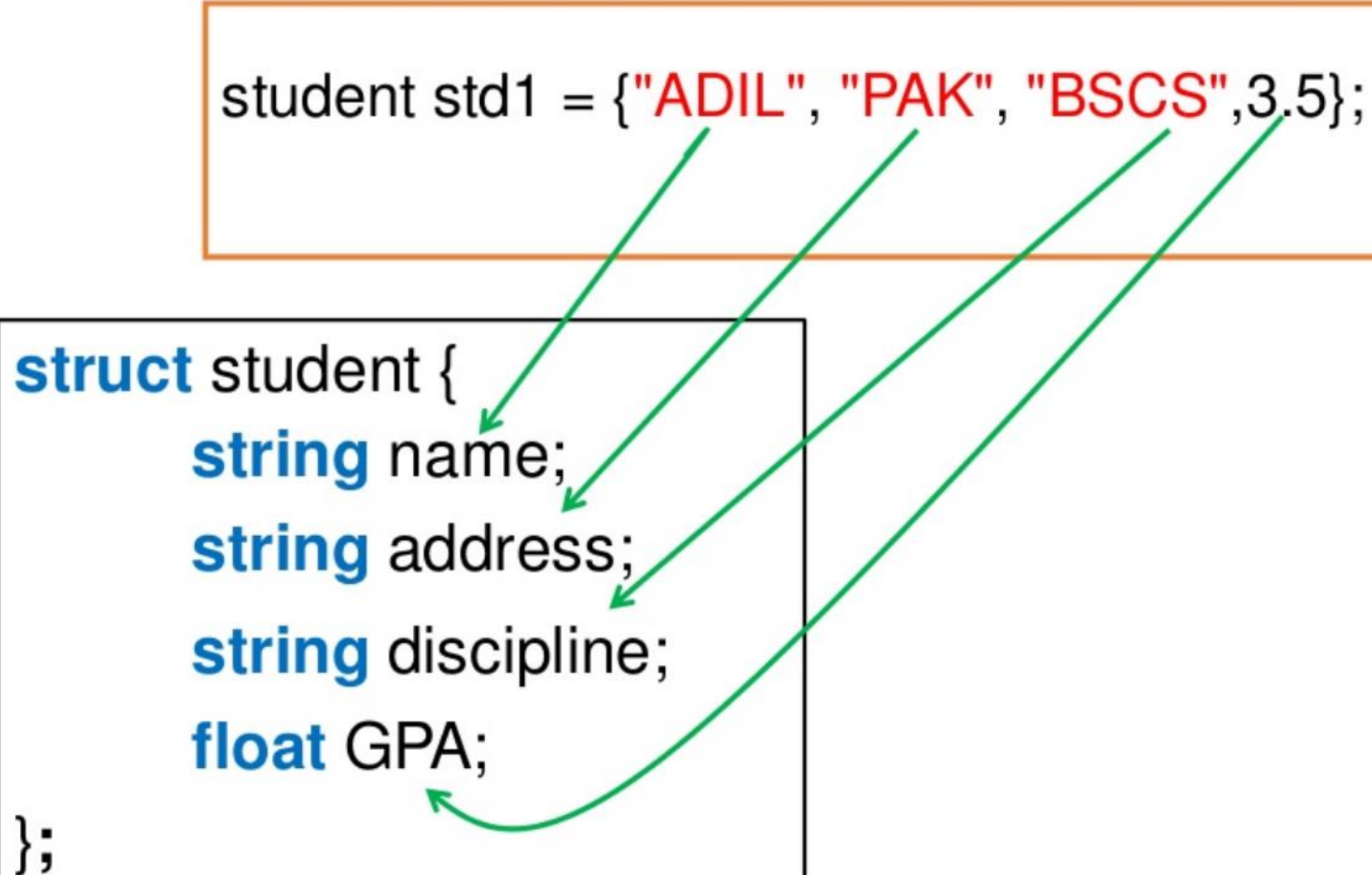
## Accessing Structure Members

- To access any member of a structure, we use the **member access operator (.)**.
- The member access operator is coded as a period between the structure variable name and the structure member that we wish to access.
- Remember we would use **struct** keyword to define variables of structure type.

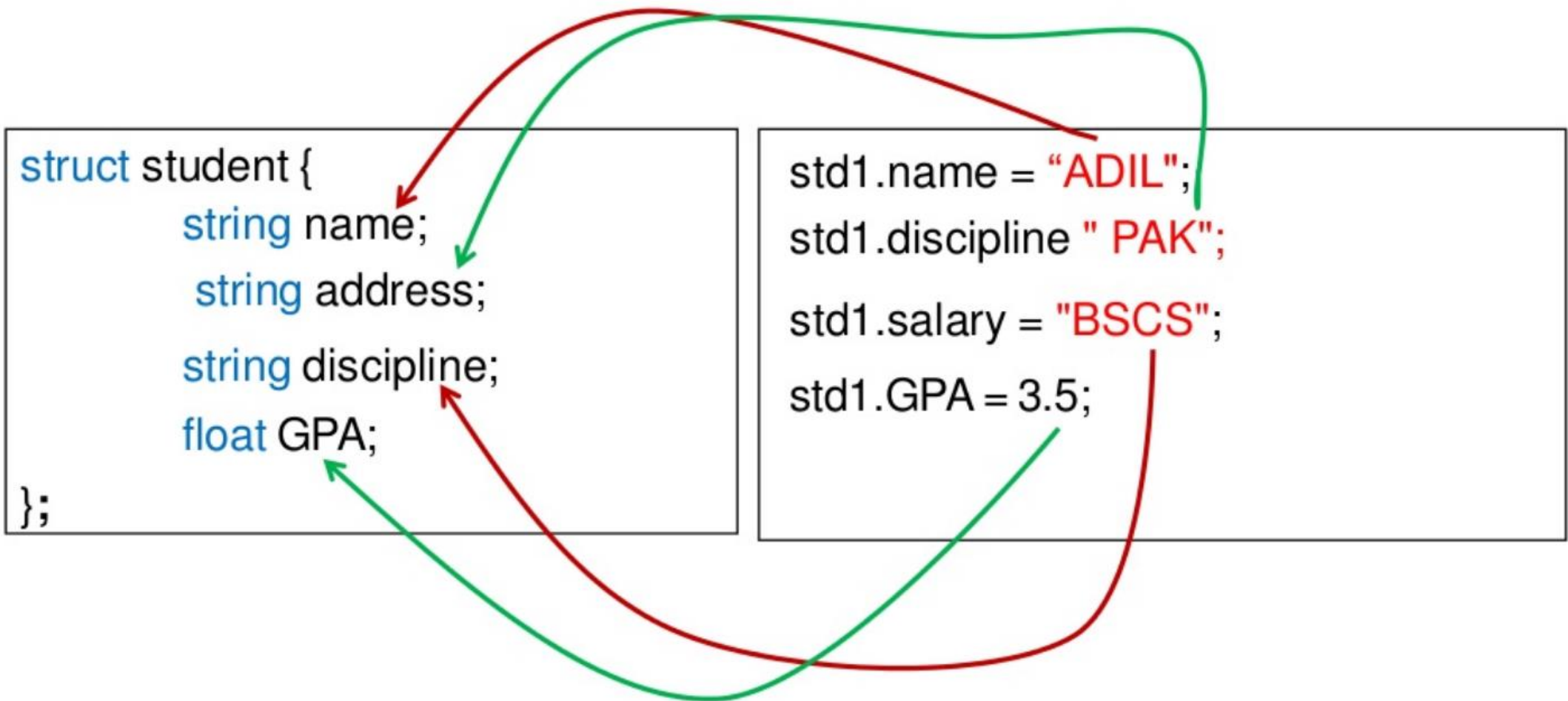
# Initializing Structures(1<sup>st</sup> Way)

```
student std1 = {"ADIL", "PAK", "BSCS", 3.5};
```

```
struct student {  
    string name;  
    string address;  
    string discipline;  
    float GPA;  
};
```



# Initializing Structures(2nd Way)



## Structure Variable in Assignment Statement

`S1 = S2;`

- The statement assigns the value of each member of S2 to the corresponding member of S1. Note that one structure variable can be assigned to another only when they are of the same structure type, otherwise compiler will give an error.

## Limitation with Structures are:

- `S1 + S2;`
- `S1 - S2;`
- `S1 * S2;`
- `S1 / S2;`



- `S1 = S2;`





## Difference Between Array and Structure

Array	Structure
Array is collection of homogeneous data.	Structure is the collection of heterogeneous data.
Array data are access using index.	Structure elements are access using . operator.
Array allocates static memory.	Structures allocate dynamic memory.
Array element access takes less time than structures.	Structure elements takes more time than Array.

# PRACTICAL IMPLEMENTATION

[HTTPS://YOUTU.BE/A6WKB6EWYKE](https://youtu.be/A6WKB6EWYKE)





THANKS 😊

