

DAILY ASSESSMENT FORMAT

Date:	22-06-2020	Name:	Abhishek
Course:	C Tutorial by SOLOLEARN	USN:	4al17ec001
Topic:	1] Structures & Unions 2] Memory Management	Semester & Section:	6 & 'A'
Github Repository:	Abhishek-online-courses		

FORENOON SESSION DETAILS

Image of session

The screenshot shows a mobile application interface for a C programming course. At the top, there is a blue header bar with a hamburger menu icon, a diagram icon, the title 'Structures & Unions', and a blue circle containing the number '143'. Below the header, the main content area is divided into four cards. Each card has a title, a progress indicator (e.g., '1/4'), and a green bar at the bottom indicating the number of questions and a checkmark. The cards are: 'Structures' (1/4, 5 questions), 'Working With Structures' (2/4, 4 questions), 'Unions' (3/4, 3 questions), and 'Working With Unions' (4/4, 3 questions). At the bottom left, there is a 'Module 5 Quiz' card (5 questions).

Topic	Progress	Questions	Status
Structures	1/4	5 questions	✓
Working With Structures	2/4	4 questions	✓
Unions	3/4	3 questions	✓
Working With Unions	4/4	3 questions	✓
Module 5 Quiz		5 questions	✓

Report –

Structures & Unions

- A structure is a user-defined data type that groups related variables of different data types.
- A structure declaration includes the keyword struct, a structure tag for referencing the structure, and curly braces { } with a list of variable declarations called members.
- The typedef keyword creates a type definition that simplifies code and makes a program easier to read. typedef is commonly used with structures because it eliminates the need to use the keyword struct when declaring variables.
- A union declaration uses the keyword union, a union tag, and curly braces { } with a list of members. Union members can be of any data type, including basic types, strings, arrays, pointers, and structures. You access the members of a union variable by using the . dot operator between the variable name and the member name.

Memory Management

- The stdlib.h library includes memory management functions.
- Many algorithms implement a dynamic array because this allows the number of elements to grow as needed. Because elements are not allocated all at once, dynamic arrays typically use a structure to keep track of current array size, current capacity, and a pointer to the elements.