CSE 1200

Software Development - I

Ashek Seum

seum.cse@aust.edu

016 2324 7656

Structure

Topics:

- Introduction
- Structure Declaration and Use as Variables
- Constructor Functions, Function Arguments and Return Values
- Structure Functions
- Array in Structure
- Array of Structure
- Nested Structure
- Structure with pointer

Introduction

As a goalkeeper of a football game or the info of a particular book in a library management system cannot be implemented through using a single int, char, float, double type variables, we need to resort to a mechanism which can incorporate multiple information and functionality of an entity in a single data-type (i.e. the name, club, country, shot-stopping, reflexes and movement of a goalkeeper).

Structure helps us to define custom-data types according to our requirements.

Structure

Topics:

- Introduction
- Structure Declaration and Use as Variables
- Constructor Functions, Function Arguments and Return Values
- Structure Functions
- Array in Structure
- Array of Structure
- Nested Structure
- Structure with pointer

Our example problem

- We will define a Vector structure for designing a **Point** entity, its properties and functionalities.

- Then we will design a structure **Line** too, which will be extended using Vector class (because you need **two points to define a line**).

Table: Our Requirements

Entity	Data	Functionality
Vector	x,y - coordinates	initialization, Scalar Multiple Self- Scaling Dot Product Information output Vector Addition
Line	Two endpoints (vectors)	Initialization Information Output

Structure Definition, Value Assignment and Retrieval

- We will see some codes (1,2)

- Note to remember: Structure data are by default public- which means you can access them from outside the scope.

Structure

Topics:

- Introduction
- Structure Declaration and Use as Variables
- Constructor Functions, Function Arguments and Return Values
- Structure Functions
- Array in Structure
- Array of Structure
- Nested Structure
- Structure with pointer

Constructor functions

- Constructor functions can be used to declare structure variables with initial values as assigning values structure variables in different statements is a cumbersome process
- Declare functions with same name as the structure inside the declaration of structure
- Pass the initial values as parameters to that function
- Unlike typical structure functions, constructor functions are not called using structure variables with dot(.)- rather they are automatically called when you declare the structure variables.
- Don’t forget to include the empty constructor in structure definition. It will enable us to declare structure variables without initializing them.
- Constructor functions are always defined before all other functions.

Let’s see an example. (3)

Structures as Function Arguments and Return Values

We will see some codes (4)

Structure

Topics:

- Introduction
- Structure Declaration and Use as Variables
- Constructor Functions, Function Arguments and Return Values
- Structure Functions
- Array in Structure
- Array of Structure
- Nested Structure
- Structure with pointer

Structures as Arguments & Returned Values within Structure

- You can use structure variables as structure function arguments and return values.

We will see some codes (5)

Structure

Topics:

- Introduction
- Structure Declaration and Use as Variables
- Constructor Functions, Function Arguments and Return Values
- Structure Functions
- Array in Structure
- Array of Structure
- Nested Structure
- Structure with pointer

Use of Array, Array of Structures and Structures inside a Structure

- We can use array of built-in data types (int, char etc.) as structure data
- We can use array of structures (i.e- a collection of Vectors)
- We can also use structures as data of another structure
- Inferring the three above mentioned features- we can use array of structures as data of another structure

We will see some codes (6)

Structure

Topics:

- Introduction
- Structure Declaration and Use as Variables
- Constructor Functions, Function Arguments and Return Values
- Structure Functions
- Array in Structure
- Array of Structure
- Nested Structure
- Structure with pointer

Pass By Value and Pass By Reference

- When a parameter is **passed by reference**, the caller and the callee **use the same variable** for the parameter. If the callee modifies the parameter variable, the effect is visible to the caller's variable.
- When a parameter is **passed by value**, the caller and callee have **two independent variables** with the same value. If the callee modifies the parameter variable, the effect is not visible to the caller.

We will see some codes (7)