

Topics covered in this lab:

- 1- Copy Constructor
- 2- Passing objects as an arguments

Section A

What is a constructor in OOP?

- a) A method that allows an object to copy another object's properties
- b) A special method that is used to create and initialize objects of a class
- c) A method used to access private data members of a class
- d) A method that defines the relationship between two classes

What is a class in OOP?

- a) An object created from another object
- b) A template or blueprint for creating objects
- c) A method that allows for method overloading
- d) A type of data structure for organizing data

What is the concept of method overloading in OOP?

- a) Creating multiple methods with the same name but different parameters in a class
- b) Creating methods with the same name and the same parameters in a class
- c) Creating multiple constructors in a class
- d) Inheriting methods from a base class

What is the primary purpose of accessor methods in a class?

- a) To modify the internal state of an object.
- b) To provide a public interface for retrieving the value of a private member variable.

- c) To provide a public interface for changing the value of a private member variable.
- d) To create new objects of the class.

What is the primary purpose of mutator methods in a class?

- a) To provide a public interface for retrieving the value of a private member variable.
- b) To create new objects of the class.
- c) To modify the internal state of an object by changing the value of a private member variable.
- d) To hide the implementation details of a class.

What is the benefit of using accessors and mutators in OOP?

- a) They make the code more efficient.
- b) They allow direct access to private member variables.
- c) They provide a controlled way to access and modify the state of objects.
- d) They eliminate the need for constructors.

Section B

Problem 1:

Book

- title: string
- author: string
- publicationYear: int

- + Book ()
- + Book (string, string, int)
- + Book (Book&)
- + ~Book ()
- + getTitle (): string
- + getAuthor (): string
- + getPublicationYear (): int
- + setTitle (string)
- + setAuthor (string)
- + setPublicationYear (int)
- + DisplayBookInfo ()
- + CompareBooks (Book&): bool

Demonstrate the class in a program that:

1. Create an object named "book1" using the parameterized constructor (title = "To Kill a Mockingbird," author = "Harper Lee" publication year = 1960, page count = 281). Display the book's information using the "DisplayBookInfo" function.
2. Create an object named "book2" using the default constructor. Set the title to "The Great Gatsby," the author to "F. Scott Fitzgerald" and the publication year to 1925 using mutator functions. Display the book's information.
3. Demonstrate the Copy Constructor: In the program, create a new Book object, "book3," by using the copy constructor and passing "book1" as an argument. Then, display the information for "book3" using the "DisplayBookInfo" function.
4. Compare the data of book1 with book2 using "CompareBooks" function.