IOOM Assignment I

1. Write a C++ code that includes default, parameterized, and copy constructors, with dynamically allocated data members, composition, and proper destruction.

   Class Definition: Define a class named DynamicArray to manage a dynamic array of integers. Include the following private data members-
   arr: A pointer to dynamically allocated integer array.
   size: An integer representing the size of the array.

   Default Constructor: Implement a default constructor that initializes the arr pointer to nullptr and the size to 0.

   Parameterized Constructor: Implement a parameterized constructor that takes an integer n as an argument and allocates memory for an array of size n.

   Copy Constructor: Implement a copy constructor that creates a deep copy of the object, copying the dynamically allocated array and its size.

   Destructor: Properly deallocate the dynamically allocated memory.

   Member Functions: Implement the following member functions:
   setData: Takes an array of integers and its size as arguments and sets the internal array and size accordingly.
   displayData: Displays the elements of the array.
   getSize: Returns the size of the array.

   Testing: In the main function, create instances of the DynamicArray class using the default, parameterized, and copy constructors. Test each constructor and member function to ensure correct functionality. Demonstrate the deep copy behavior of the copy constructor.

   Instructions: Support separate compilation and create a makefile for the above program.

2. Implement a simple student management system using Java classes and objects.

   Class Definitions: Create two classes: Student and StudentManager. The Student class should have the following private data members:
   id: Student ID (integer).
   name: Student name (string).
   grade: Student grade (double).

   Constructor and Methods: Implement a parameterized constructor in the Student class to initialize the student's ID, name, and grade.
   Provide getter methods to access the values of id, name, and grade.
   Implement a displayInfo method in the Student class to display the student's information.

   StudentManager Class: The StudentManager class should have an array (or ArrayList) to store multiple Student objects. Implement a method to add a new student to the system. Implement a method to display information about all students in the system.

   Testing: In the main method, create an instance of StudentManager. Add at least three students to the system using the addStudent method. Display the information of all students using the displayStudents method. Implement a method to calculate and display the average grade of all students. Allow the user to input student information interactively.