Task 01: Calculator [Estimated time 30 minutes / 20 marks]

Create a C++ program that simulates a simple calculator. The program should have the following features:

- The calculator should be able to perform basic arithmetic operations:
  - addition,
  - Subtraction
  - Multiplication
  - division.

- The program should allow the user to enter two numbers and choose an operation.
- The program should then display the result of the operation. Finally, the program should exit when the user chooses to quit.
- To demonstrate the use of virtual Destructor, you can create a base class called 'Operation', which has virtual Destructor. Then, create four derived classes which inherit from Operation and implement their respective operations. The four classes are: o Addition o Subtraction o Multiplication o Division,
- When the user chooses an operation, the program should create an object of the corresponding derived class and call its calculate() method. Since Operation has a virtual Destructor, the Destructor of the derived class will be called automatically when the object goes out of scope.

Task 02: Student Record [Estimated time 30 minutes / 20 marks]

- By implementing pure virtual function, make a class 'Student' with data members
  - Name
  - department as protected type

Two pure virtual member functions

- get_data() "to input the student record"
- show_data() "display student record".

- Derive three classes 'Medical', 'Engineering', and 'Science' from Student class with its own member functions of get_data() "to input the student record" and show_data() "display student record" while using data members of student class.

- In main create an array of pointers of base class having size three and assign to each index remaining class objects addresses. By using for loop iterate array and take input then again display records using for loop.

Task 03: Shapes Array [Estimated time 30 minutes / 10 marks]

Create base class Shape with virtual functions for area calculation and display. Derive classes like Circle, Square, and Triangle from Shape, implementing their specific area calculation algorithms.

In main have a D-Array of type Shape. Use Polymorphism to add new shapes into this array.
For this overload Add Shape function to have 3 definitions, one for each shape type. Provide a menu based program Asking user what shape they want to insert. Call the respective function and insert that shape in the array. Provide functions display all shapes as well.

Task 04: Animal Sound [Estimated time 20 minutes / 10 marks]

Create a base class called Animal with a public member function called make_sound() that prints "The animal makes a sound."

Create three sub classes called Cat, Dog, and Bird that inherit from the Animal class.

Override the make_sound() function in each subclass to print "Meow" for Cat, "Woof" for Dog, and "Chirp" for Bird.

Finally, create an array of pointers to Animal objects that contains one instance of each subclass, and call the make_sound() function on each object to verify that the correct sound is made for each animal.

Have an array in the main function.