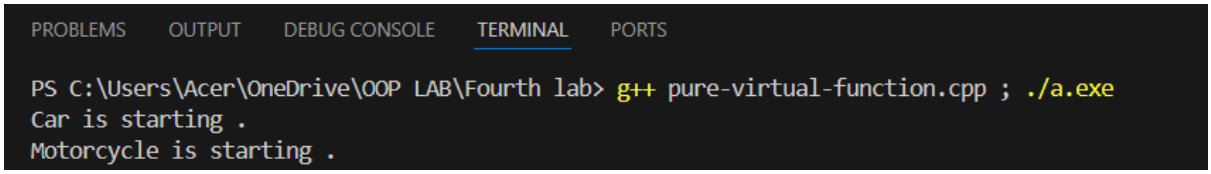
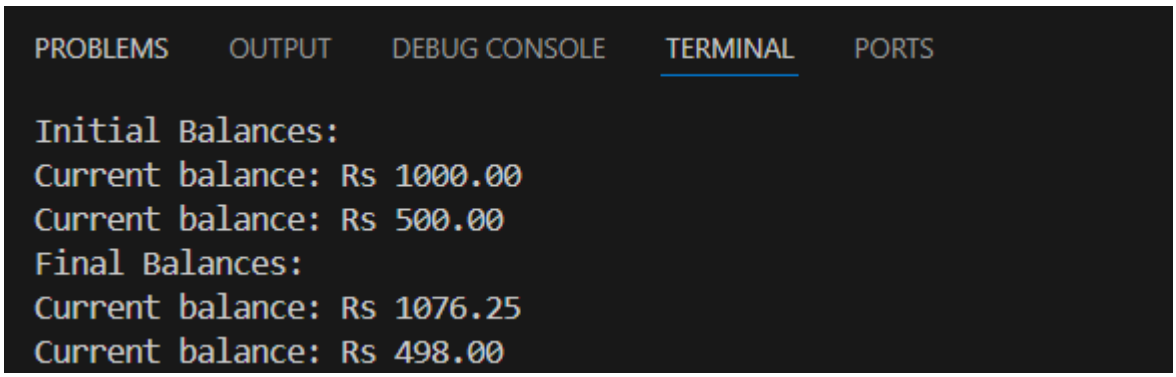
Create an abstract base class Vehicle with a pure virtual function void start(). Derive two classes, Car and Motorcycle, from Vehicle. Implement the start() function differently in each derived class. Write a program to create objects of both Car and Motorcycle and call their start() functions.

Output:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS C:\Users\Acer\OneDrive\OOP LAB\Fourth lab> g++ pure-virtual-function.cpp ; ./a.exe
Car is starting .
Motorcycle is starting .

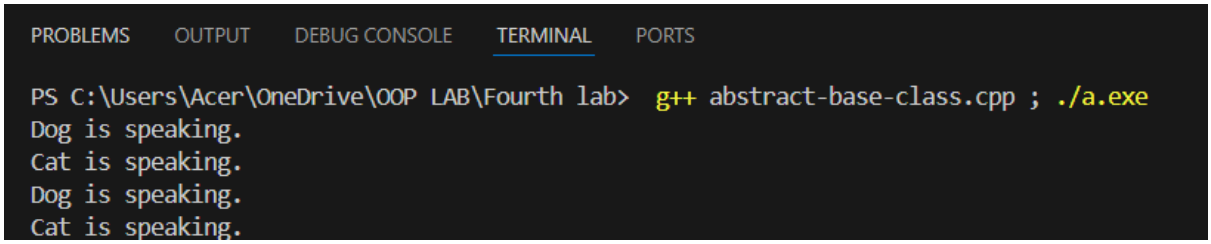
Create a C++ program that models a simple banking system. Implement a base class Account and derived classes Savings Account and Checking Account. Use virtual functions to perform operations like deposit, withdrawal, and interest calculation.

Output:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Initial Balances:
Current balance: Rs 1000.00
Current balance: Rs 500.00
Final Balances:
Current balance: Rs 1076.25
Current balance: Rs 498.00

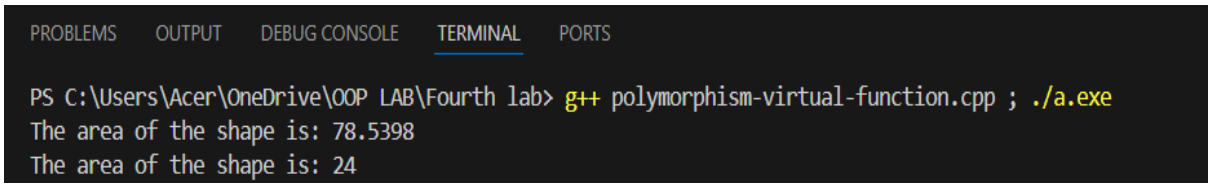
Write a program that defines an abstract base class Animal with a pure virtual function void speak(). Create two derived classes, Dog and Cat, which implement the speak() function. Use an array of Animal pointers to store instances of both Dog and Cat. Write a loop to make all animals in the array speak.

Output:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS C:\Users\Acer\OneDrive\OOP LAB\Fourth lab> g++ abstract-base-class.cpp ; ./a.exe
Dog is speaking.
Cat is speaking.
Dog is speaking.
Cat is speaking.

Write a C++ program that demonstrates the concept of polymorphism using virtual functions. Create a base class Shape with a virtual function area(). Create two derived classes, Circle and Rectangle, each with their own implementation of the area() function. Calculate and display the area of different shapes using polymorphism.

Output:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS C:\Users\Acer\OneDrive\OOP LAB\Fourth lab> g++ polymorphism-virtual-function.cpp ; ./a.exe
The area of the shape is: 78.5398
The area of the shape is: 24