import java.util.Stack;

public class ReverseStack {

// Function to insert an element at the bottom of the stack

public static void insertAtBottom(Stack<Integer> stack, int item) {

if (stack.isEmpty()) {

stack.push(item);

} else {

int top = stack.pop();

insertAtBottom(stack, item);

stack.push(top);

}

}

// Function to reverse the stack using recursion

public static void reverse(Stack<Integer> stack) {

if (!stack.isEmpty()) {

int top = stack.pop();

reverse(stack);

insertAtBottom(stack, top);

}

}

// Main function to test the reverse function

public static void main(String[] args) {

Stack<Integer> stack = new Stack<>();

// Push elements into the stack

stack.push(1);

stack.push(2);

stack.push(3);

stack.push(4);

System.out.println("Original Stack: " + stack);

// Reverse the stack

reverse(stack);

System.out.println("Reversed Stack: " + stack);

}

}