```
# Define a custom exception class
class CustomError(Exception):
    def __init__(self, message="A custom error occurred."):
        self.message = message
        super().__init__(self.message)

def perform_custom_operation(value):
    try:
        # Check a condition, and if it's not met, raise the custom exception
        if value < 0:
            raise CustomError("Value must be a non-negative number.")
        else:
            # Perform the custom operation if the condition is met
            result = value * 2
            print(f"Custom operation result: {result}")

    except CustomError as ce:
        print(f"CustomError: {ce}")
    except Exception as e:
        print(f"An unexpected error occurred: {e}")
    finally:
        print("Custom operation attempt completed.")

# Example usage:
try:
    perform_custom_operation(10)
    perform_custom_operation(-5)  # This should raise the CustomError
except CustomError as ce:
    print(f"CustomError caught outside the function: {ce}")
except Exception as e:
    print(f"An unexpected error occurred outside the function: {e}")
```

```
Custom operation result: 20
Custom operation attempt completed.
CustomError: Value must be a non-negative number.
Custom operation attempt completed.
```

Start coding or generate with AI.