**Question: Shared Memory**

You are tasked with designing a program that utilizes shared memory for interprocess communication in a Linux environment. The program involves two processes: Process A and Process B.

1. Define a structure named `SharedData` that includes the following members:

- An integer variable `counter` to keep track of the number of times Process A writes to shared memory.

- An array of characters `message[256]` to store a message sent by Process A to Process B.

2. Write C++ code to create a shared memory segment capable of holding an instance of `SharedData`.

3. Implement Process A:

- Attach to the shared memory segment.

- Prompt the user to enter a message (up to 255 characters) to be sent to Process B.

- Increment the `counter` variable.

- Copy the message into the `message` array in shared memory.

- Detach from the shared memory segment.

4. Implement Process B:

- Attach to the shared memory segment.

- Wait for Process A to write data into shared memory.

- Once data is available, read the message and the counter value from shared memory.

- Print the received message along with the counter value.

- Detach from the shared memory segment.

Ensure your code includes error handling for shared memory creation, attachment, detachment, and synchronization between the processes.