**Shubhan Singh**

**SE-Comps B/Batch C**

**2022300118**

OS Experiment 8: Shared Memory

(All source code files are submitted on moodle)

**Problem statement**: Create a Chat Bot using Shared Memory in Linux  
  
In this assignment, you will implement a chat bot using shared memory in Linux. The chat bot should communicate with the user using the shared memory segment and respond to the user's messages based on the pre-defined rules.  
  
Here are the steps to follow:  
1. Define the message format: Decide on the message format that will be used to communicate between the chat bot and the user. For example, the message format could be a string with a pre-defined prefix for the chat bot's response.  
2. Create a shared memory segment: Use System V IPC mechanisms to create a shared memory segment. Use the shmget() function to create the shared memory segment and get a shared memory identifier.  
3. Attach to the shared memory segment: Use the shmat() function to attach to the shared memory segment and get a pointer to the shared memory segment.  
4. Implement the chat bot logic: Write code that listens for messages from the user and responds appropriately based on pre-defined rules. For example, if the user sends a message starting with "Hi", the chat bot could respond with "Hello, how can I help you today?".  
5. Send the chat bot's response: After the chat bot has generated a response, write the response to the shared memory segment.  
6. Wait for the user's next message: After writing the response to the shared memory segment, wait for the user to send the next message.  
7. Detach from the shared memory segment: Use the shmdt() function to detach from the shared memory segment.  
8. Clean up the shared memory segment: When the chat bot is done, use the shmctl() function to delete the shared memory segment.

**Files used**: *sharedmem\_chatbot.c*

**Output:**

