|  |  |
| --- | --- |
| Name | Manish Shashikant Jadhav |
| UID no. | 2023301005 |

|  |
| --- |
| Experiment 9 |

|  |  |
| --- | --- |
| **Aim** | Creating a chatbot and using System V IPC mechanisms to create a shared memory segment. |
| **Code:** | #include <stdio.h>  #include <stdlib.h>  #include <string.h>  #include <unistd.h>  #include <sys/ipc.h>  #include <sys/shm.h>  #define SHM\_SIZE 1024  #define KEY 1234  *// Pre-defined messages*  char \*responses[] = {      "Hello, how can I help you today?",      "I'm sorry, I cannot provide you with real-time stocks price of  TATA MOTORS.",      "HP Victus is currently the best laptop for now.",      "Currently Royal Enfeild Continental GT650 Twin is the best bike."  };  int main() {      int shmid;      char \*shmaddr;      char \*user\_input = NULL;      size\_t len = 0;      ssize\_t read;  *// Creating a shared memory segment*      shmid = shmget(KEY, SHM\_SIZE, IPC\_CREAT | 0666);      if (shmid == -1) {          perror("shmget");          exit(1);      }  *// Attaching to the shared memory segment*      shmaddr = (char \*) shmat(shmid, NULL, 0);      if (shmaddr == (char \*) -1) {          perror("shmat");          exit(1);      }      printf("Chat Bot initialized. Type 'exit' to quit.\n");  *// Chat loop*      while (1) {            printf("You: ");          read = getline(&user\_input, &len, stdin);          if (read == -1) {              perror("getline");              exit(1);          }  *// Responses*          if (strncmp(user\_input, "exit", 4) == 0) {              break;          } else if (strncmp(user\_input, "Hi", 2) == 0) {              strncpy(shmaddr, responses[0], SHM\_SIZE);          } else if (strncmp(user\_input, "What is the current stocks  price of TATA MOTORS?", 37) == 0) {              strncpy(shmaddr, responses[1], SHM\_SIZE);          } else if (strncmp(user\_input, "Which is the best laptop?", 20) == 0) {              strncpy(shmaddr, responses[2], SHM\_SIZE);          } else if (strncmp(user\_input, "Which is the best bike?", 18) == 0) {              strncpy(shmaddr, responses[3], SHM\_SIZE);          }          else {              strncpy(shmaddr, "I'm sorry, I don't understand that.", SHM\_SIZE);          }  *// Wait for a moment*          sleep(1);  *// Read bot's response from shared memory*          printf("Bot: %s\n", shmaddr);      }  *// Detaching from the shared memory segment*      if (shmdt(shmaddr) == -1) {          perror("shmdt");          exit(1);      }  *// Cleaning up the shared memory segment*      if (shmctl(shmid, IPC\_RMID, NULL) == -1) {          perror("shmctl");          exit(1);      }      free(user\_input);      return 0;  } |
| **Output**: |  |
| **Conclusion** | Hence, by completing this experiment I came to know about creating a chatbot and using System V IPC mechanisms to create a shared memory segment. |