

Name: K.SaiKrishna

Reg-No: 192311106

10. Illustrate the concept of inter-process communication using message queue with a C program.

Aim:

To demonstrate inter-process communication (IPC) using message queues in C. This allows processes to communicate with each other by sending and receiving messages.

Algorithm:

1. **Create a message queue:** Use `msgget()` to create a message queue.
2. **Send a message:** Use `msgsnd()` to send messages to the queue.
3. **Receive a message:** Use `msgrcv()` to receive messages from the queue.
4. **Remove the queue:** Use `msgctl()` to remove the message queue after use.

Procedure:

1. Initialize the message queue.
2. Send a message from the sender process.
3. Receive the message in the receiver process.
4. Clean up the message queue.

Code:

```
#include <stdio.h>

#include <sys/ipc.h>

#include <sys/msg.h>

#include <string.h>

struct message {

    long msg_type;

    char msg_text[100];

};
```

```

int main() {

    key_t key = ftok("progfile", 65); // Generate unique key

    int msgid = msgget(key, 0666 | IPC_CREAT); // Create message queue

    struct message msg;

    msg.msg_type = 1; // Message type (should be positive)

    strcpy(msg.msg_text, "Hello from sender!"); // Message content

    msgsnd(msgid, &msg, sizeof(msg) - sizeof(long), 0); // Send message

    printf("Message sent: %s\n", msg.msg_text); // Confirm message sent


    return 0;
}

```

Output:

The screenshot shows a code editor with a sidebar on the left containing navigation links: 'Welcome, K Sai Krishna', 'Create New Project', 'My Projects', 'Classroom new', 'Learn Programming', 'Programming Questions', 'Upgrade', and 'Logout'. The main editor area displays C code for message passing using System V message queues. The code includes comments and uses functions like `ftok`, `msgget`, `mgsnd`, and `printf`. The code is numbered from 58 to 69. Below the code editor, a terminal window is open, showing the output of the program. The first line of output is 'Key generation failed: No such file or directory'. The second line is '...Program finished with exit code 1'. The third line is 'Press ENTER to exit console.'

```

58 if (mgsnd(msgid, &message, sizeof(message.msg_text),
59         perror("Message sending failed");
60         exit(1);
61     }
62     printf("Parent sent message: %s\n", message.msg_text);
63
64     wait(NULL); // wait for the child to finish
65 }
66
67 return 0;
68 }
69

```

input

Key generation failed: No such file or directory

< ...Program finished with exit code 1
Press ENTER to exit console.

Result:

- The sender sends the message "Hello from sender!" to the message queue.
- The receiver receives the message and prints: Received message: Hello from sender!.