OS LAB EXP 12b:Message Queue

**Program :To perform communication using message queues, following are the steps -**

**Writer Process:**

18CSC205J-Operating Systems Lab

 **Step 1** - Create a message queue or connect to an already existing message queue (msgget())

 **Step 2** – specify the message type as 1.

 **Step 3-** Write into message queue (msgsnd())

 **Step 4**- terminate the process

**Reader Process:**

 **Step 1** - Create a message queue or connect to an already existing message queue (msgget())

 **Step 2** – specify the message type as 1.

 **Step 3** – Read from the message queue (msgrev())

 **Step 4** - Perform control operations on the message queue (msgctl())

 **Step 5** – terminate the reader process

**Writer program:**

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

#include<sys/ipc.h>

#include<sys/msg.h>

int main (int argc, char \*argv [ ])

{int len, mid,i=1;

struct buffer

{long mtype;

char buf[50];

}x;

mid=msgget((key\_t)6,IPC\_CREAT|0666);

x.mtype=atoi(argv[1]);

strcpy(x.buf,argv[2]);

len=strlen(x.buf);

msgsnd(mid,&x,len,0);

printf("Message of size %d sent successfully \n",len);

return 0;

}

**OUTPUT:**

**$./a.out 1 welcome** (note: 1 is messageId and welcome is message)

**Message of size 7 sent successfully**

**Reader program:**

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

#include<sys/ipc.h>

#include<sys/msg.h>

int main(int argc, char \*argv[ ])

{ int len,mid,i=1;

struct buffer

{ long mtype;

char buf[50];

}x;

mid=msgget((key\_t)6,0666);

x.mtype=atoi(argv[1]);

len=atoi(argv[2]);

msgrcv(mid, &x,len,x.mtype,0);

printf("The message is:%s\n",x.buf);

return 0;

}

**OUTPUT:**

**$ ./a.out 1 7** (note: 1 is messageId and 7 is size of the message)

**The message is: welcome**