Name:M.Ahmed Arshed

Rollno:P176099

Section:A

Assigment #1

# Inter Process Communication using message passing:-

Communication between processes using shared memory requires processes to share some variable and it completely depends on how programmer will implement it. One way of communication using shared memory can be imagined like this: Suppose process1 and process2 are executing simultaneously and they share some resources or use some information from other process, process1 generate information about certain computations or resources being used and keeps it as a record in shared memory. When process2 need to use the shared information, it will check in the record stored in shared memory and take note of the information generated by process1 and act accordingly. Processes can use shared memory for extracting information as a record from other process as well as for delivering any specific information to other process.

# WRITER PROCESS

#include <stdio.h>

#include <sys/ipc.h>

#include <sys/msg.h>

struct mesg\_buffer {

long mesg\_type;

char mesg\_text[100];

} message;

int main()

{

key\_t key;

int id;

key = ftok("progfile", 65);

id = msgget(key, 0666 | IPC\_CREAT);

message.mesg\_type = 1;

printf("Write Data : ");

gets(message.mesg\_text);

msgsnd(id, &message, sizeof(message), 0);

printf("Data send is : %s \n", message.mesg\_text);

return 0;

}

# Reading PROCESS

#include <stdio.h>

#include <sys/ipc.h>

#include <sys/msg.h>

struct mesg\_buffer {

long mesg\_type;

char mesg\_text[100];

} message;

int main()

{

key\_t key;

int id;

key = ftok("progfile", 65);

id = msgget(key, 0666 | IPC\_CREAT);

msgrcv(id, &message, sizeof(message), 1, 0);

printf("Data Received is : %s \n",

message.mesg\_text);

msgctl(id, IPC\_RMID, NULL);

return 0;

}

* The ftok() function uses the identity of the file named by the given

pathname (which must refer to an existing, accessible file)

* msggetfunction is used either to create a new message queue or to locate an existing queue based on a key
* msgsnd() function is used to send a message to the message queue
* The *msgrcv()* function reads a message from the queue associated with the message queue identifier specified by *msqid* and places it in the user-defined buffer pointed to by *msgp*.
* msgctlfunction is used to perform one of several control operations on an UNIX System Services message queue