#include <sys/types.h>

#include <sys/ipc.h>

#include <sys/msg.h>

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#define MAXSIZE 128

void die(char \*s) {

perror(s);

exit(1);

}

typedef struct msgbuf {

long mtype; /\* message type, must be > 0 \*/

char mtext[MAXSIZE];

} msgbuf;

int main() {

int msqid;

int msgflg = IPC\_CREAT | 0666;

key\_t key;

struct msgbuf sbuf;

size\_t buflen;

key = 1234; // The key used for message queue

if ((msqid = msgget(key, msgflg)) < 0)

die("msgget");

// Message Type

sbuf.mtype = 1;

printf("Enter a message to add to message queue: ");

// Read a message from the user, allowing spaces

if (scanf("%[^\n]", sbuf.mtext) != 1) {

die("scanf");

}

getchar(); // Consume the newline character left by scanf

buflen = strlen(sbuf.mtext) + 1; // Length of the message, including the null-terminator

// Send the message to the message queue

if (msgsnd(msqid, &sbuf, buflen, IPC\_NOWAIT) < 0) {

printf("%d, %d, %s, %d\n", msqid, sbuf.mtype, sbuf.mtext, buflen);

die("msgsnd");

} else {

printf("Message Sent\n");

}

return 0;

}

//IPC\_msgq\_rcv.c

#include <sys/types.h>

#include <sys/ipc.h>

#include <sys/msg.h>

#include <stdio.h>

#include <stdlib.h>

#define MAXSIZE 128

void die(char \*s)

{

perror(s);

exit(1);

}

typedef struct msgbuf

{

long mtype;

char mtext[MAXSIZE];

} ;

main()

{

int msqid;

key\_t key;

struct msgbuf rcvbuffer;

key = 1234;

if ((msqid = msgget(key, 0666)) < 0)

die("msgget()");

//Receive an answer of message type 1.

if (msgrcv(msqid, &rcvbuffer, MAXSIZE, 1, 0) < 0)

die("msgrcv");

printf("%s\n", rcvbuffer.mtext);

exit(0);

}