

1. Write a program in which a message is passed by program1 to the message queue and program2 prints the same to verify it.

p5.c

```
#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];
};
int main()
{
    int msqid,i;
    int msgflg = IPC_CREAT | 0666;
    key_t key;
    struct msgbuf sbuf;
    size_t buflen;
    key = 1503;
    if ((msqid = msgget(key, msgflg )) < 0) //Get the message queue ID for the given key
        die("msgget");
    //Message Type
    sbuf.mtype = 1;
    printf("Enter a message to add to message queue : ");
    scanf("%s",sbuf.mtext);
    getchar();
    buflen = strlen(sbuf.mtext) + 1 ;
    if (msgsnd(msqid, &sbuf, buflen, IPC_NOWAIT) < 0)
    {
        printf ("%d, %ld, %s, %d \n", msqid, sbuf.mtype, sbuf.mtext, buflen);
        die("msgsnd");
    }
    else{
        printf("Message Sent\n");
    }
    exit(0);
}
```

p6.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,j;
    key_t key;
    struct msgbuf rcvbuffer;
    key = 1503;
    if ((msqid = msgget(key, 0666)) < 0)
        die("msgget()");

    if (msgrcv(msqid, &rcvbuffer, MAXSIZE, 1, 0) < 0)
        die("msgrcv");
    printf("%s\n", rcvbuffer.mtext);
    exit(0);
}
```



```

#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];
};
int main()
{
    int msqid,i;
    int msgflg = IPC_CREAT | 0666;
    key_t key;
    struct msgbuf sbuf;
    size_t buflen;
    key = 1400;
    if ((msqid = msgget(key, msgflg )) < 0) //Get the message queue ID for the given key
        die("msgget");
    //Message Type
    sbuf.mtype = 1;
    printf("Enter a message to add to message queue : ");
    scanf("%s",sbuf.mtext);
    getchar();
    buflen = strlen(sbuf.mtext) + 1 ;
    if (msgsnd(msqid, &sbuf, buflen, IPC_NOWAIT) < 0)
    {
        printf ("%d, %ld, %s, %d \n", msqid, sbuf.mtype, sbuf.mtext, buflen);
        die("msgsnd");
    }
    sbuf.mtype = 2;
    printf("Enter a message to add to message queue : ");
    scanf("%s",sbuf.mtext);
    getchar();
    buflen = strlen(sbuf.mtext) + 1 ;
    if (msgsnd(msqid, &sbuf, buflen, IPC_NOWAIT) < 0)
    {
        printf ("%d, %ld, %s, %d \n", msqid, sbuf.mtype, sbuf.mtext, buflen);
        die("msgsnd");
    }
    sbuf.mtype = 3;

    printf("Enter a message to add to message queue : ");
    scanf("%s",sbuf.mtext);
    getchar();
    buflen = strlen(sbuf.mtext) + 1 ;
    if (msgsnd(msqid, &sbuf, buflen, IPC_NOWAIT) < 0)
    {

```

```

printf ("%d, %ld, %s, %d \n", msqid, sbuf.mtype, sbuf.mtext, buflen);
die("msgsnd");
}
sbuf.mtype = 4;
printf("Enter a message to add to message queue : ");
scanf("%[^\n]",sbuf.mtext);
getchar();
buflen = strlen(sbuf.mtext) + 1 ;
if (msgsnd(msqid, &sbuf, buflen, IPC_NOWAIT) < 0)
{
printf ("%d, %ld, %s, %d \n", msqid, sbuf.mtype, sbuf.mtext, buflen);
die("msgsnd");
}
sbuf.mtype = 5;
printf("Enter a message to add to message queue : ");
scanf("%[^\n]",sbuf.mtext);
getchar();
buflen = strlen(sbuf.mtext) + 1 ;
if (msgsnd(msqid, &sbuf, buflen, IPC_NOWAIT) < 0)
{
printf ("%d, %ld, %s, %d \n", msqid, sbuf.mtype, sbuf.mtext, buflen);
die("msgsnd");
}
else{
printf("Message Sent\n");
}
exit(0);
}

```

p2.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,j=1;
key_t key;

```

```
struct msgbuf rcvbuffer;
key = 1400;
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);
if ((msqid = msgget(key, 0666)) < 0)
die("msgget()");
//Receive an answer of message type 1.
if (msgrcv(msqid, &rcvbuffer, MAXSIZE, 2, 0) < 0)
die("msgrcv");
printf("%s\n", rcvbuffer.mtext);
if ((msqid = msgget(key, 0666)) < 0)
die("msgget()");
//Receive an answer of message type 1.
if (msgrcv(msqid, &rcvbuffer, MAXSIZE, 3, 0) < 0)
die("msgrcv");
printf("%s\n", rcvbuffer.mtext);
if ((msqid = msgget(key, 0666)) < 0)
die("msgget()");
//Receive an answer of message type 1.
if (msgrcv(msqid, &rcvbuffer, MAXSIZE, 4, 0) < 0)
die("msgrcv");
printf("%s\n", rcvbuffer.mtext);
if ((msqid = msgget(key, 0666)) < 0)
die("msgget()");
//Receive an answer of message type 1.
if (msgrcv(msqid, &rcvbuffer, MAXSIZE, 5, 0) < 0)
die("msgrcv");
printf("%s\n", rcvbuffer.mtext);
exit(0);
}
```



```

{
perror(s);
exit(1);
}
typedef struct msgbuf
{
long mtype; /* message type, must be > 0 */
char mtext[MAXSIZE];
};
int main()
{
int msqid,i;
int msgflg = IPC_CREAT | 0666;
key_t key;
struct msgbuf sbuf;
size_t buflen;
key = 1502;
if ((msqid = msgget(key, msgflg )) < 0) //Get the message queue ID for the given key
die("msgget");
//Message Type
sbuf.mtype = 1;
printf("Enter a message to add to message queue : ");
scanf("%[^\n]",sbuf.mtext);
getchar();
buflen = strlen(sbuf.mtext) + 1 ;
if (msgsnd(msqid, &sbuf, buflen, IPC_NOWAIT) < 0)
{
printf ("%d, %ld, %s, %d \n", msqid, sbuf.mtype, sbuf.mtext, buflen);
die("msgsnd");
}
sbuf.mtype = 2;
printf("Enter a message to add to message queue : ");
scanf("%[^\n]",sbuf.mtext);
getchar();
buflen = strlen(sbuf.mtext) + 1 ;
if (msgsnd(msqid, &sbuf, buflen, IPC_NOWAIT) < 0)
{
printf ("%d, %ld, %s, %d \n", msqid, sbuf.mtype, sbuf.mtext, buflen);
die("msgsnd");
}
sbuf.mtype = 3;

printf("Enter a message to add to message queue : ");
scanf("%[^\n]",sbuf.mtext);
getchar();
buflen = strlen(sbuf.mtext) + 1 ;
if (msgsnd(msqid, &sbuf, buflen, IPC_NOWAIT) < 0)
{
printf ("%d, %ld, %s, %d \n", msqid, sbuf.mtype, sbuf.mtext, buflen);
die("msgsnd");
}
sbuf.mtype = 4;

```



```

printf("Enter a message to add to message queue : ");
scanf("%[^\n]",sbuf.mtext);
getchar();
buflen = strlen(sbuf.mtext) + 1 ;
if (msgsnd(msqid, &sbuf, buflen, IPC_NOWAIT) < 0)
{
printf ("%d, %ld, %s, %d \n", msqid, sbuf.mtype, sbuf.mtext, buflen);
die("msgsnd");
}
sbuf.mtype = 5;
printf("Enter a message to add to message queue : ");
scanf("%[^\n]",sbuf.mtext);
getchar();
buflen = strlen(sbuf.mtext) + 1 ;
if (msgsnd(msqid, &sbuf, buflen, IPC_NOWAIT) < 0)
{
printf ("%d, %ld, %s, %d \n", msqid, sbuf.mtype, sbuf.mtext, buflen);
die("msgsnd");
}
else{
printf("Message Sent\n");
}
exit(0);
}

```

## p2.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,j;
key_t key;
struct msgbuf rcvbuffer;
key = 1502;
if ((msqid = msgget(key, 0666)) < 0)
die("msgget()");

```

```
printf("which message do u want to receive");
scanf("%d",&j);
if (msgrcv(msgqid, &rcvbuffer, MAXSIZE, j, 0) < 0)
die("msgrcv");
printf("%s\n", rcvbuffer.mtext);
exit(0);
}
```

```

user@ASUS-TUF-Gaming-A17-FA706ICB-FA706ICB - user@ASUS-TUF-Gaming-A17-FA706ICB-FA706ICB -
user@ASUS-TUF-Gaming-A17-FA706ICB-FA706ICB: $ gcc p3.c
p3.c:16:11: warning: useless storage class specifier in empty declaration
16 | }
   | ^~
p3.c: In function 'main':
p3.c:35:24: warning: format '%d' expects argument of type 'int', but argument 5 has type 'size_t' (aka 'long unsigned int') [-Wformat=]
35 | printf("%d, %ld, %s, %d\n", msqid, sbuf.ntype, sbuf.ntext, bufLen);
   |                ~~~~~^~~~~
   |                |             |
   |                int          size_t (aka long unsigned int)
p3.c:45:24: warning: format '%d' expects argument of type 'int', but argument 5 has type 'size_t' (aka 'long unsigned int') [-Wformat=]
45 | printf("%d, %ld, %s, %d\n", msqid, sbuf.ntype, sbuf.ntext, bufLen);
   |                ~~~~~^~~~~
   |                |             |
   |                int          size_t (aka long unsigned int)
p3.c:56:24: warning: format '%d' expects argument of type 'int', but argument 5 has type 'size_t' (aka 'long unsigned int') [-Wformat=]
56 | printf("%d, %ld, %s, %d\n", msqid, sbuf.ntype, sbuf.ntext, bufLen);
   |                ~~~~~^~~~~
   |                |             |
   |                int          size_t (aka long unsigned int)
p3.c:66:24: warning: format '%d' expects argument of type 'int', but argument 5 has type 'size_t' (aka 'long unsigned int') [-Wformat=]
66 | printf("%d, %ld, %s, %d\n", msqid, sbuf.ntype, sbuf.ntext, bufLen);
   |                ~~~~~^~~~~
   |                |             |
   |                int          size_t (aka long unsigned int)
p3.c:76:24: warning: format '%d' expects argument of type 'int', but argument 5 has type 'size_t' (aka 'long unsigned int') [-Wformat=]
76 | printf("%d, %ld, %s, %d\n", msqid, sbuf.ntype, sbuf.ntext, bufLen);
   |                ~~~~~^~~~~
   |                |             |
   |                int          size_t (aka long unsigned int)
user@ASUS-TUF-Gaming-A17-FA706ICB-FA706ICB: $ ./a.out
Enter a message to add to message queue : rf
Enter a message to add to message queue : fe
Enter a message to add to message queue : dscf
Enter a message to add to message queue : drdf
Enter a message to add to message queue : df
Message Sent
user@ASUS-TUF-Gaming-A17-FA706ICB-FA706ICB: $ █

```

```
user@user-ASUS-TUF-Gaming-A17-FA706ICB-FA706ICB: ~  
user@user-ASUS-TUF-Gaming-A17-FA706ICB-FA706ICB: $ gcc p4.c  
p4.c:18:11: warning: useless storage class specifier in empty declaration  
18 | }  
|  
p4.c:19:11: warning: return type defaults to 'int' [-Wimplicit-int]  
19 | main()  
|  
user@user-ASUS-TUF-Gaming-A17-FA706ICB-FA706ICB: $ ./a.out  
which message do u want to receive2  
fa  
user@user-ASUS-TUF-Gaming-A17-FA706ICB-FA706ICB: $
```