

Question 1**Producer**

/*

Write a producer and consumer program in C using the FIFO queue.
 The producer should write a set of 4 integers into the FIFO queue
 and the consumer should display the 4 integers.

*/

```
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<sys/types.h>
#include<limits.h>
#include<fcntl.h>
#include<sys/msg.h>
#include<sys/stat.h>
#include<string.h>
```

```
#define FIFO_NAME "my_fifo"
#define BUFFER_SIZE 1000
```

```
int main(int argc, char *argv[])
{
    int pipe_fd;
    int res;
    int open_mode=O_WRONLY;
    int n=0;
    char buffer[BUFFER_SIZE+1];

    if(access(FIFO_NAME,F_OK)==-1)
    {
        res=mknfif(FIFO_NAME,0777);
        if(res!=0)
        {
            fprintf(stderr, "Could not create file%s\n",FIFO_NAME );
            exit(EXIT_FAILURE);
        }
    }

    printf("Process %d opening FIFO O_WRONLY\n",getpid());
    pipe_fd=open(FIFO_NAME,open_mode);
    printf("Process %d result %d\n",getpid(),pipe_fd);

    if (pipe_fd!=-1)
    {
        printf("Enter 4 numbers\n");
```

```

while(n<4)
{
    scanf("%s",buffer);
    res=write(pipe_fd,buffer,BUFFER_SIZE);

    if(res==-1)
    {
        fprintf(stderr, "Write Error on Pipe\n");
        exit(EXIT_FAILURE);
    }
    n++;
}
(void)close(pipe_fd);
}
else
    exit(EXIT_FAILURE);

printf("Process %d Finished\n",getpid() );
exit(EXIT_SUCCESS);

}

```

Consumer

```

/*
Write a producer and consumer program in C using the FIFO queue.
The producer should write a set of 4 integers into the FIFO queue
and the consumer should display the 4 integers.
*/

```

```

#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<sys/types.h>
#include<limits.h>
#include<fcntl.h>
#include<sys/msg.h>
#include<sys/stat.h>
#include<string.h>

#define FIFO_NAME "my_fifo"
#define BUFFER_SIZE 1000

int main(int argc, char *argv[])
{
    int pipe_fd;
    int res;
    int open_mode=O_RDONLY;
    int n=0;
    char buffer[BUFFER_SIZE+1];
    memset(buffer,'\0',sizeof(buffer));

```

```

printf("Process %d opening FIFO O_RDONLY\n",getpid());
pipe_fd=open(FIFO_NAME,open_mode);
printf("Process %d result %d\n",getpid(),pipe_fd);

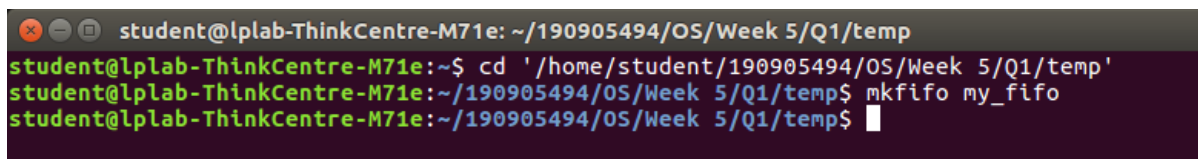
if (pipe_fd!=-1)
{
    do
    {
        res=read(pipe_fd,buffer,BUFFER_SIZE);
        printf("%s\n",buffer );
        n++;
    }while(n<4);

    (void)close(pipe_fd);
}

else
    exit(EXIT_FAILURE);

printf("Process %d Finished, %d bytes read\n",getpid(),n );
exit(EXIT_SUCCESS);
}

```



```

student@lplab-ThinkCentre-M71e: ~/190905494/OS/Week 5/Q1/temp
student@lplab-ThinkCentre-M71e:~$ cd '/home/student/190905494/OS/Week 5/Q1/temp'
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q1/temp$ mkfifo my_fifo
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q1/temp$

```

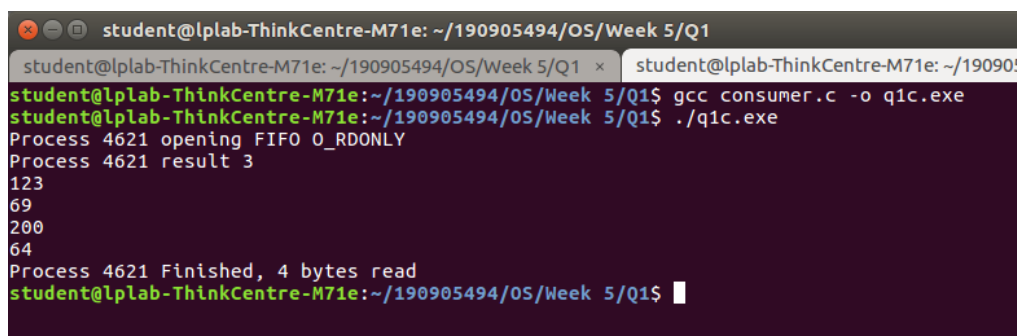
Creating FIFO Pipe



```

student@lplab-ThinkCentre-M71e: ~/190905494/OS/Week 5/Q1
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q1$ gcc producer.c -o q1p.exe;
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q1$ ./q1p.exe
Process 4620 opening FIFO O_WRONLY
Process 4620 result 3
Enter 4 numbers
123
69
200
64
Process 4620 Finished
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q1$

```



```

student@lplab-ThinkCentre-M71e: ~/190905494/OS/Week 5/Q1
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q1$ gcc consumer.c -o q1c.exe
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q1$ ./q1c.exe
Process 4621 opening FIFO O_RDONLY
Process 4621 result 3
123
69
200
64
Process 4621 Finished, 4 bytes read
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q1$

```

Question 2

// Demonstrate creation, writing to, and reading from a pipe.

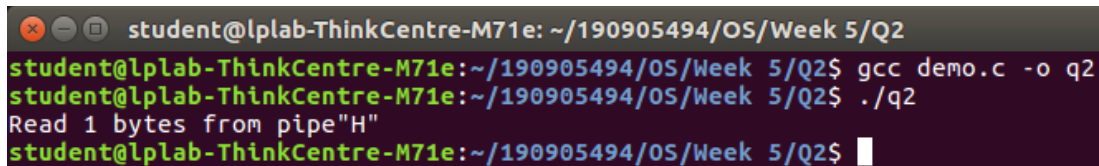
```
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<sys/types.h>
#include<sys/ipc.h>
#include<sys/msg.h>
#include<string.h>

int main(int argc, char *argv[])
{
    int n, i=0;
    int fd[2];
    char buf[1025];
    char *data = "Hello How Are You";

    // creating
    pipe(fd);

    // Writing
    while(i<strlen(data)){
        write(fd[1],data,strlen(data));
        i++;
    }

    // reading
    i=0;
    while(i<strlen(data)){
        if(n=read(fd[0],buf,1024)>=0)
        {
            buf[n]='\0';
            printf("Read %d bytes from pipe\"%s\"\\n",n,buf);
        } else
            perror("Read");
        exit(0);
        i++;
    }
}
```



```
student@lplab-ThinkCentre-M71e: ~/190905494/OS/Week 5/Q2
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q2$ gcc demo.c -o q2
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q2$ ./q2
Read 1 bytes from pipe"H"
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q2$
```

Question 3

Parent

```
/*
Write a C program to implement one side of FIFO.
*/

#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<sys/types.h>
#include<limits.h>
#include<fcntl.h>
#include<sys/msg.h>
#include<sys/stat.h>
#include<string.h>

#define FIFO_NAME "my_fifo"
#define BUFFER_SIZE 10000

int main(int argc, char *argv[])
{
    int pipe_fd;
    int res;
    int open_mode1=O_WRONLY;
    int open_mode2=O_RDONLY;
    int n=0;
    char buffer[BUFFER_SIZE+1];

    if(access(FIFO_NAME,F_OK)==-1)
    {
        res=mknod(FIFO_NAME,0777);

        if(res!=0)
        {
            fprintf(stderr, "Could not create file%s\n",FIFO_NAME);
            exit(EXIT_FAILURE);
        }
    }

    printf("You can start chatting with User 2 now\n");

    while(1)
    {
        pipe_fd=open(FIFO_NAME,open_mode2);

        printf("\nText from User 1: ");
        res=read(pipe_fd,buffer,BUFFER_SIZE);
        printf("%s\n",buffer );
    }
}
```

```

        close(pipe_fd);

        printf("Wait for User 1 reply...\n");
        pipe_fd=open(FIFO_NAME,open_mode1);

        printf("\nEnter Text to send User 1: ");
        fgets(buffer,BUFFER_SIZE,stdin);
        res=write(pipe_fd,buffer,BUFFER_SIZE);

        close(pipe_fd);

    }

    (void)close(pipe_fd);

    printf("Process %d Finished\n",getpid());
    exit(EXIT_SUCCESS);
}

```

Child

```

/*
Write a C program to implement one side of FIFO.
*/

```

```

#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<sys/types.h>
#include<limits.h>
#include<fcntl.h>
#include<sys/msg.h>
#include<sys/stat.h>
#include<string.h>

#define FIFO_NAME "my_fifo"
#define BUFFER_SIZE 10000

int main(int argc, char *argv[])
{
    int pipe_fd;
    int res;
    int open_mode1=O_WRONLY;
    int open_mode2=O_RDONLY;
    int n=0;
    char buffer[BUFFER_SIZE+1];

    if(access(FIFO_NAME,F_OK)==-1)

```

```

{
    res=mkfifo(FIFO_NAME,0777);

    if(res!=0)
    {
        fprintf(stderr, "Could not create file%s\n",FIFO_NAME);
        exit(EXIT_FAILURE);
    }
}

printf("You can start chatting with User 2 now\n");

while(1)
{
    pipe_fd=open(FIFO_NAME,open_mode1);

    printf("\nEnter Text to send User 2: ");
    fgets(buffer,BUFFER_SIZE,stdin);
    res=write(pipe_fd,buffer,BUFFER_SIZE);

    close(pipe_fd);

    printf("Wait for User 2 reply...\n");
    pipe_fd=open(FIFO_NAME,open_mode2);

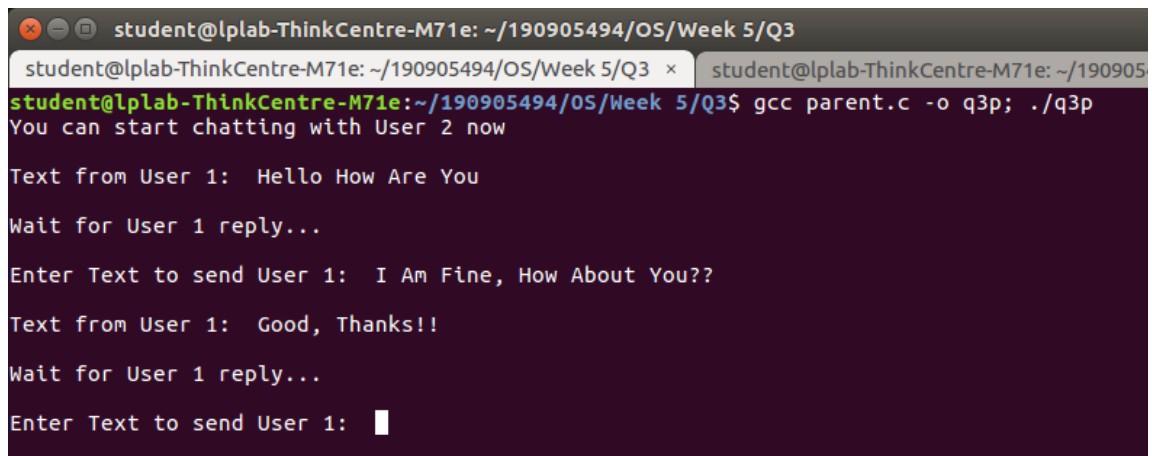
    printf("\nText from User 2: ");
    res=read(pipe_fd,buffer,BUFFER_SIZE);
    printf("%s\n",buffer );
    close(pipe_fd);

}

(void) close(pipe_fd);

printf("Process %d Finished\n",getpid());
exit(EXIT_SUCCESS);
}

```



```

student@lplab-ThinkCentre-M71e: ~/190905494/OS/Week 5/Q3
student@lplab-ThinkCentre-M71e: ~/190905494/OS/Week 5/Q3 x student@lplab-ThinkCentre-M71e: ~/190905
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q3$ gcc parent.c -o q3p; ./q3p
You can start chatting with User 2 now

Text from User 1:  Hello How Are You

Wait for User 1 reply...

Enter Text to send User 1:  I Am Fine, How About You??

Text from User 1:  Good, Thanks!!

Wait for User 1 reply...

Enter Text to send User 1:  █

```

```
student@lplab-ThinkCentre-M71e: ~/190905494/OS/Week 5/Q3
student@lplab-ThinkCentre-M71e: ~/190905494/OS/Week 5/Q3 x student@lplab-ThinkCentre-M71e: ~/19090
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q3$ gcc child.c -o q3c; ./q3c
You can start chatting with User 2 now

Enter Text to send User 2: Hello How Are You
Wait for User 2 reply...

Text from User 2: I Am Fine, How About You??

Enter Text to send User 2: Good, Thanks!!
Wait for User 2 reply...
```

Question 4

/*
Write a C program reading and writing a binary file in C.
*/

```
#include<stdio.h>
#include<stdlib.h>

int main()
{
    FILE* fptr;
    int num=0;
    fptr=fopen("demo.bin","wb+");

    printf("Enter some numbers : \n");

    for(int i=0;i<4;i++)
    {
        scanf("%d",&num);
        fwrite(&num,sizeof(int),1,fptr);
    }

    printf("Writing done!\n");
    fclose(fptr);

    fptr=fopen("demo.bin","rb");

    for(int i=0;i<4;i++)
    {
        fread(&num,sizeof(int),1,fptr);
        printf("%d\n",num);
    }
}
```



```
student@lplab-ThinkCentre-M71e: ~/190905494/OS/Week 5/Q4
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q4$ gcc bin.c -o q4; ./q4
Enter some numbers :
10
20
30
40
Writing done!
10
20
30
40
student@lplab-ThinkCentre-M71e:~/190905494/OS/Week 5/Q4$
```

‘demo.bin’ file created to read and write from



bin.c



demo.bin



q4