```
/*
        Mihir Kulkarni
        L9 33132
        Aim: Inter process communication in Linux using FIFOs
*/
```

```c
#include <stdio.h>
#include <string.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>

int main()
{
        int fd;
        char *myfifo = "./myfifo";

        // Creating the named file(FIFO)
        mkfifo(myfifo, 0666);
        int i = 0,size=100;;
        char arr1[size], data[size], ch;

                // Open FIFO for write only
                fd = open(myfifo, O_WRONLY);
                i = 0;
                // fgets(arr2, 80, stdin);
                printf("Enter Data : \n");
                while ((ch = getchar()) != '$')
                {
                        data[i] = ch;
                        i++;
                }

                printf("data sended..\n");
                data[i] = '\0';
                write(fd, data, size);
                close(fd);

                // Open FIFO for Read only
                fd = open(myfifo, O_RDONLY);

                // Read from FIFO
                read(fd, arr1, size);

                // Print the read message
                printf("\nData from process2 :: \n%s\n", arr1);
                close(fd);

        return 0;
}
```

```c
#include <stdio.h>
#include <string.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>

int main()
{
        int fd1;

        // FIFO file path
        char * myfifo = "./myfifo";

         FILE *fptr;
      fptr = fopen("namedPipe.txt", "w");

        mkfifo(myfifo, 0666);

        int i=0,no_of_words=0,size=100,no_of_lines=0;
        char str1[size], dataRecv[size],ch;

                printf("Waiting for process1 to send data.....\n");
                // First open in read only and read
                fd1 = open(myfifo,O_RDONLY);
                read(fd1, dataRecv, size);

                // Print the read string and close
                printf("Data from process1 ::\n%s\n", dataRecv);
                close(fd1);

                while(dataRecv[i]!='\0'){
                        if(dataRecv[i]=='\n' && dataRecv[i+1]!='\n'){
                                no_of_words+=1;
                                no_of_lines+=1;
                        }
                        else if((dataRecv[i]==' ' && dataRecv[i+1]!=' ') || dataRecv[i]==',' ){
                                no_of_words+=1;
                        }


                        i++;
                }
                // Now open in write mode and write
                // string taken from user.
                printf("\nData After Processing :: \nNo of characters : %d \nNo of words : %d \nNo
of lines : %d\n",i,no_of_words+1,no_of_lines+1);
                sprintf(str1, "No of chars : %d \nNo of words : %d \nNo of lines : %d",
i,no_of_words+1,no_of_lines+1);
                fputs(str1,fptr);
```
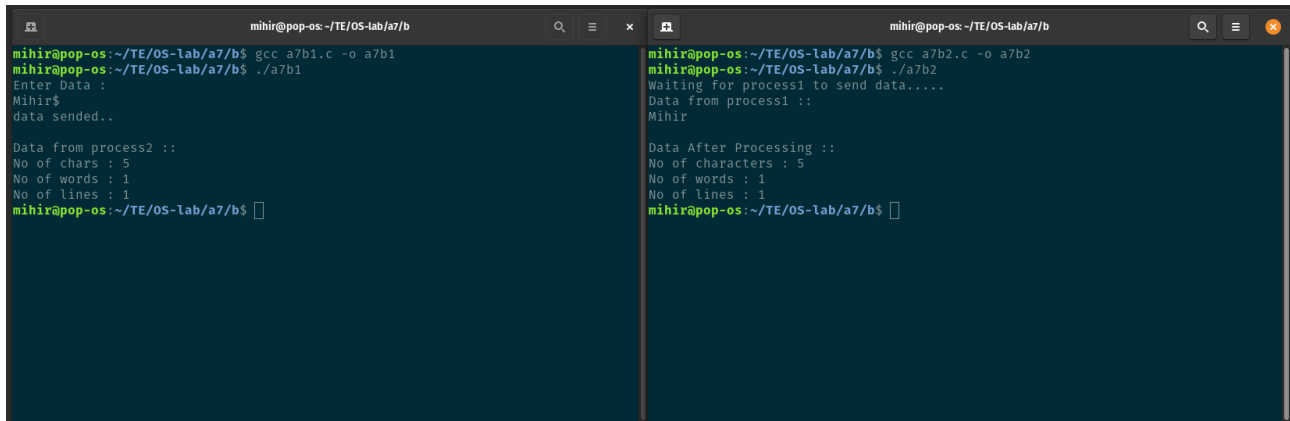
```
            fd1 = open(myfifo,O_WRONLY);
            write(fd1, str1, size);
            close(fd1);

        return 0;
}
```