

```

/* fserver.c */
#define DEFAULT_PROTOCOL 0
#define MAXLINE 100
int readLine(int fd, char* str);
/* 파일 서버 프로그램 */
int main (int argc, char* argv[])
{
    int sfd, cfd, port, clientlen;
    FILE *fp;
    char inmsg[MAXLINE], outmsg[MAXLINE];
    struct sockaddr_in serveraddr, clientaddr;
    struct hostent *hp;
    char *haddrp;
    signal(SIGCHLD, SIG_IGN);
    if (argc != 2) {

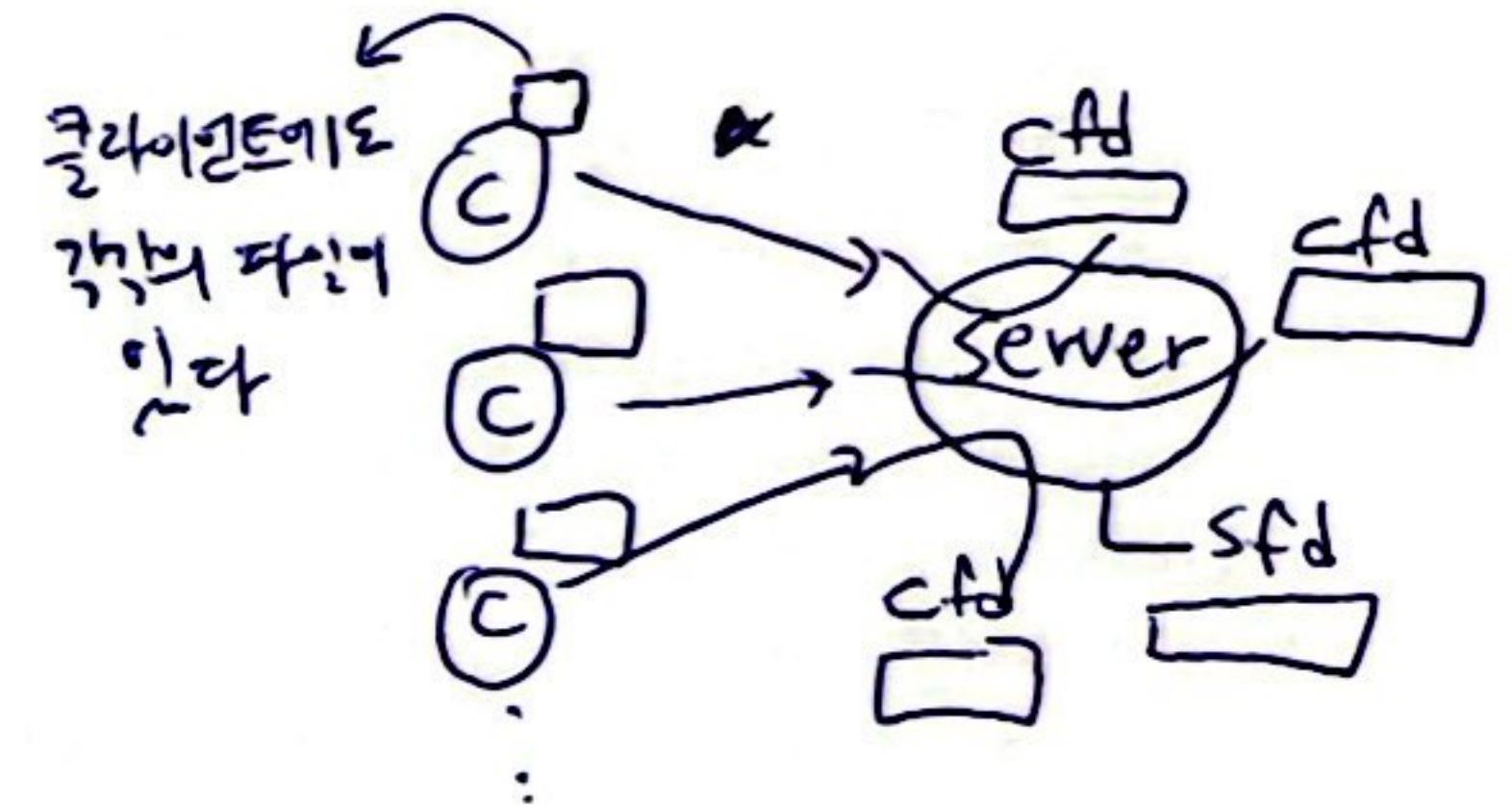
```

```

        fprintf(stderr, "사용법: %s <port>\n", argv[0]);
        exit(0);
    }
    port = atoi(argv[1]);
    sfd = socket(AF_INET, SOCK_STREAM, DEFAULT_PROTOCOL);
    bzero((char *) &serveraddr, sizeof(serveraddr));
    serveraddr.sin_family = AF_INET;
    serveraddr.sin_addr.s_addr = htonl(INADDR_ANY);
    serveraddr.sin_port = htons((unsigned short)port);
    bind(sfd, (struct sockaddr *) &serveraddr, sizeof(serveraddr));
    listen(sfd, 5);
    while (1) {
        clientlen = sizeof(clientaddr);
        cfd = accept(sfd, (struct sockaddr *) &clientaddr, &clientlen); // 연결 요청 수락
        /* 클라이언트의 도메인 이름과 IP 주소 결정 */
        //hp = gethostbyaddr((char *) &clientaddr.sin_addr.s_addr,
        //                    sizeof(clientaddr.sin_addr.s_addr), AF_INET);
        haddrp = inet_ntoa(clientaddr.sin_addr);
        printf("서버: %s (%d)에 연결됨\n", haddrp, clientaddr.sin_port);
        if (fork() == 0) {
            readLine(cfd, inmsg); /* 소켓에서 파일 이름을 읽는다 */
            if (!strcmp(inmsg, "*"))
                break;
            fp = fopen(inmsg, "r");
            if (fp == NULL) {
                write(cfd, "해당 파일 없음", 10);
            } else {
                /* 파일에서 한 줄씩 읽어 소켓을 통해 보낸다 */
                while(fgets(outmsg, MAXLINE, fp) != NULL)
                    write(cfd, outmsg, strlen(outmsg)+1);
            }
            close(cfd);
            exit(0);
        } else close(cfd);
    }
}
int readLine(int fd, char* str)
{
    int n;
    do {
        n = read(fd, str, 1);
    } while (n > 0 && *str++ != '\0');
    return(n > 0);
}

```

소켓은 연결 파일은 리턴함.
클라이언트가 사용하는 파일.



클라이언트 → 서버 보내는 내용은 SF

서버 → 클라이언트 보내는 내용은 C

채팅하고 싶으면 sfd에 보내는 내용

모든 cfd에 보내면 됨

```

}
int readLine(int fd, char* str)
{
    int n;
    do {
        n = read(fd, str, 1);
    } while (n > 0 && *str++ != '\0');
    return(n > 0);
}

```


/* fclient.c */

2200992

최성현

```
#define DEFAULT_PROTOCOL 0
#define MAXLINE 100
int readLine(int fd, char* str);
/* 파일 클라이언트 프로그램 */
int main (int argc, char* argv[])
{
    int cfd, port, result;
    char *host, inmsg[MAXLINE], outmsg[MAXLINE];
    struct sockaddr_in serverAddr;
    struct hostent *hp;
    if (argc != 3) {
        fprintf(stderr, "사용법 : %s <host> <port>\n", argv[0]);
        exit(0);
    }
    host = argv[1];
    port = atoi(argv[2]);
    cfd = socket(AF_INET, SOCK_STREAM, DEFAULT_PROTOCOL);
    /* 서버의 IP 주소와 포트 번호를 채운다. */
    if ((hp = gethostbyname(host)) == NULL)
        perror("gethostbyname error"); // 호스트 찾기 오류
    bzero((char *) &serverAddr, sizeof(serverAddr));
    serverAddr.sin_family = AF_INET;
    bcopy((char *)hp->h_addr_list[0],
        (char *)&serverAddr.sin_addr.s_addr, hp->h_length);
    serverAddr.sin_port = htons(port);
    do { /* 연결 요청 */ 요청하면 서버가 accept함
        result = connect(cfd, (struct sockaddr *) &serverAddr, sizeof(serverAddr));
        if (result == -1) sleep(1);
    } while (result == -1);
    printf("다운로드할 파일 이름 입력:");
    scanf("%s", inmsg);
    write(cfd, inmsg, strlen(inmsg)+1);
    /* 소켓으로부터 파일 내용 읽어서 프린트 */
    while (readLine(cfd, outmsg))
        printf("%s", outmsg);
    close(cfd);
    exit(0);
}

int readLine(int fd, char* str)
{
    int n;
    do {
        n = read(fd, str, 1);
    } while (n > 0 && *str++ != '\0');
    return(n > 0);
}
```