

# 네트워크 프로그래밍 과제 #03

## ☞ 주의 사항

아래의 조건(소켓 타입, 구조체 및 cmd type)을 따르지 않고 구현한 경우, 0점 처리함  
각 소스 파일에 학번, 이름(영문 가능) 주석이 없는 경우, 파일당 -1점 감점  
각 기능에 대한 소스 코드만 있고, 동작이 되지 않는 경우에는 아래의 점수를 받을 수 없음  
화면 출력 과정에서 쓰레기 값이 출력되거나, 출력 내용이 맞지 않으면 항목당 -5점 감점함

### 1. TCP 데이터 송수신(SEQ, ACK)을 이용한 파일 전송 프로그램 (20점)

- 파일제출: hw3\_server.c, hw3\_client.c

#### ■ 공통 사항

- ✓ 클라이언트, 서버의 read/write 버퍼 크기는 **1024 바이트로 고정**
- ✓ 파일 입출력 함수는 저수준(read, write)함수 또는 고수준(fread, fwrite) 함수 모두 사용 가능함
- ✓ 서버의 파일 크기와 클라이언트의 파일 크기는 동일해야 되며, 파일 이름이 화면에 입력한 내용과 동일해야 됨 (버퍼 사용 주의)
- ✓ 클라이언트와 서버는 서로 다른 폴더에서 동작을 시킴

#### ■ 클라이언트, 서버 공용 데이터 구조

```
#define BUF_SIZE 1024
#define SEQ_START 0
typedef struct {
    int seq;          // SEQ number
    int ack;          // ACK number
    int buf_len;       // File read/write bytes
    char buf[BUF_SIZE]; // 파일 이름 또는 파일 내용 전송
}Packet;
```

#### ■ 클라이언트 (2점)

- ✓ 클라이언트는 사용자로부터 전송 받을 파일의 이름을 입력 받고, 서버에 해당 파일 이름을 전송함 (buf[BUF\_SIZE]에 파일 이름을 저장하여 전송)
- ✓ 클라이언트는 서버에서 전송한 파일 수신이 완료되면 소켓을 종료

#### ■ 서버 (3점)

- ✓ 서버는 클라이언트에게 전송 받은 파일이름을 확인해서 파일이 있는 경우, 해당 파일을 전송함
- ✓ 파일이 서버에 없는 경우, "**File Not Found**"를 클라이언트에게 전송하고 소켓 종료

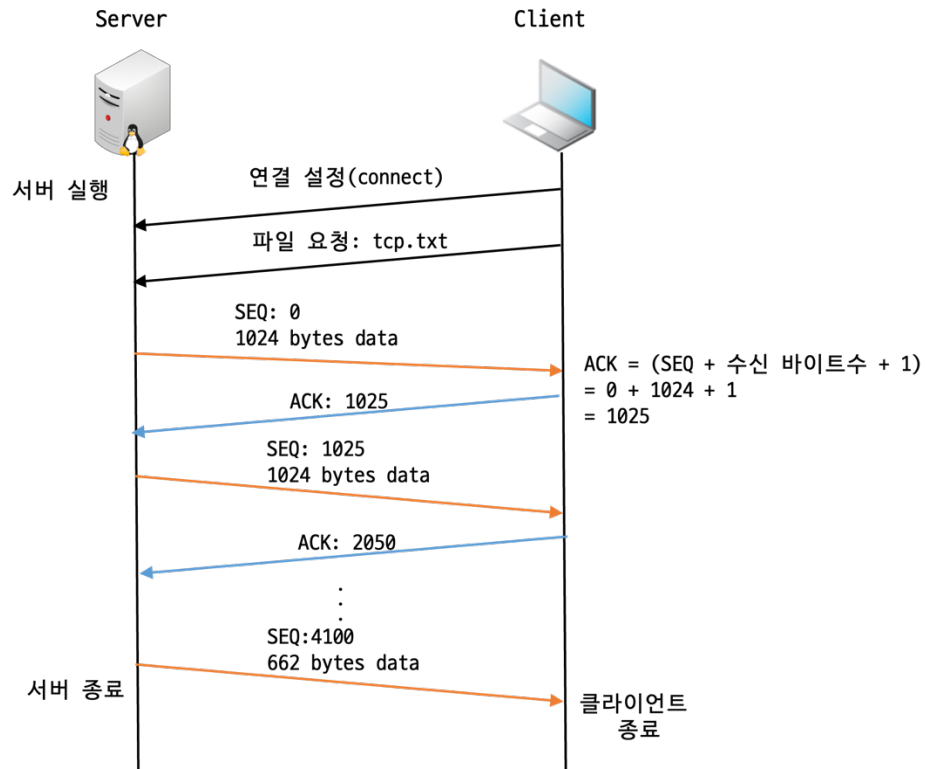
#### ■ TCP 데이터 송수신 과정 출력 (15점)

- ✓ TCP 데이터 송수신 과정(SEQ, ACK 내용)을 반드시 화면 출력해야 됨: **8점**

- 송수신 과정에서 SEQ, ACK 출력이 없거나 SEQ, ACK의 출력은 있지만 번호가 정확하지 않는 경우: 해당 점수 없음
- ✓ 서버의 시작 SEQ번호: 0
- ✓ 서버의 파일 크기와 클라이언트가 수신한 파일 크기가 동일해야 됨: 7점
  - 파일 크기가 같지 않은 경우, 해당 점수 없음

#### ■ 동작 과정

- 클라이언트가 tcp.txt 파일을 서버에게 요청한 경우



#### 실행 결과 #01: 클라이언트가 요청한 파일이 없는 경우

[서버] : 서버와 클라이언트는 서로 다른 폴더에서 실행 시킴

서버	클라이언트
<pre>\$ ./hw3_server 9190</pre> <hr/> <p>File Transmission Server</p> <hr/> <p>aaa.txt File Not Found Server socket close</p>	<pre>client\$ ./hw3_client 127.0.0.1 9190</pre> <p>Input file name: aaa.txt [Client] request aaa.txt</p> <p>File Not Found Client socket close</p>

## 실행 결과 #02: 클라이언트가 tcp.txt 파일을 요청한 경우

서버	클라이언트
<pre>\$ ./hw3_server 9190 ----- File Transmission Server ----- [Server] sending tcp.txt  [Server] Tx SEQ: 0, 1024 byte data [Server] Rx ACK: 1025  [Server] Tx SEQ: 1025, 1024 byte data [Server] Rx ACK: 2050  [Server] Tx SEQ: 2050, 1024 byte data [Server] Rx ACK: 3075  [Server] Tx SEQ: 3075, 1024 byte data [Server] Rx ACK: 4100  [Server] Tx SEQ: 4100, 662 byte data tcp.txt sent (4758 Bytes) Server socket close</pre>	<pre>client\$ ./hw3_client 127.0.0.1 9191 Input file name: tcp.txt [Client] request tcp.txt  [Client] Rx SEQ: 0, len: 1024 bytes [Client] Tx ACK: 1025  [Client] Rx SEQ: 1025, len: 1024 bytes [Client] Tx ACK: 2050  [Client] Rx SEQ: 2050, len: 1024 bytes [Client] Tx ACK: 3075  [Client] Rx SEQ: 3075, len: 1024 bytes [Client] Tx ACK: 4100  [Client] Rx SEQ: 4100, len: 662 bytes tcp.txt received (4758 Bytes) Client socket close</pre>

## 클라이언트에서 파일 수신 및 파일 크기 확인

<pre>\$ ll total 44 drwxrwxr-x 2 netprog netprog 4096 10월 13 15:50 ./ drwxr-xr-x 3 netprog netprog 4096 10월 13 15:50 ../ -rw-rw-r-- 1 netprog netprog 3001 10월 13 15:50 abc.txt -rwxrwxr-x 1 netprog netprog 16936 10월 13 15:50 client* -rw-r--r-- 1 netprog netprog 3823 10월 13 15:51 hw3_client.c -rw-rw-r-- 1 netprog netprog 4758 10월 13 15:52 tcp.txt</pre>
--