

2019100991 이선아 2차 과제

<서버측 프로그램 소스 코드>

// 2019100991 이선아

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>
#include <sys/socket.h>
#define BUF_SIZE 1024
void error_handling(char* score);

int main(int argc, char* argv[]) {
    int serv_sock, client_sock;
    char score[BUF_SIZE], grade;
    int tot, avg;
    int str_len;

    struct sockaddr_in serv_addr, client_addr;
    socklen_t client_addr_size;

    if(argc != 2) {
        printf("Usage : %s <PORT>Wn", argv[0]);
        exit(1);
    }

    serv_sock = socket(PF_INET, SOCK_STREAM, 0);
    if(serv_sock == -1)
        error_handling("socket error");

    memset(&serv_addr, 0, sizeof(serv_addr));
    serv_addr.sin_family = AF_INET;
    serv_addr.sin_addr.s_addr = htonl(INADDR_ANY);
    serv_addr.sin_port = htons(atoi(argv[1]));

    if(bind(serv_sock, (struct sockaddr*)&serv_addr, sizeof(serv_addr)) == -1)
        error_handling("bind error");

    if(listen(serv_sock, 5) == -1)
        error_handling("listen error");

    client_addr_size = sizeof(client_addr);

    client_sock = accept(serv_sock, (struct sockaddr*)&client_addr, &client_addr_size);
    if(client_sock == -1)
        error_handling("accept error");

    read(client_sock, score, BUF_SIZE);

    tot = score[0] + score[1] + score[2];
```

```

    avg = tot / 3;
    printf("총점 : %d\n", tot);
    printf("평균 : %d\n", avg);

    if(avg >= 90)
        grade = 'A';
    else if(avg >= 80)
        grade = 'B';
    else if(avg >= 70)
        grade = 'C';
    else if(avg >= 60)
        grade = 'D';
    else
        grade = 'F';

    write(client_sock, (char*) &grade, 1); // 클라이언트로 학점 전송

    close(client_sock);
    close(serv_sock);

    return 0;
}

void error_handling(char* score) {
    fputs(score, stderr);
    fputc('\n', stderr);
    exit(1);
}

```

<클라이언트측 프로그램 소스 코드>

// 2019100991 이선아

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>
#include <sys/socket.h>
#define BUF_SIZE 1024
void error_handling(char* grade);

int main(int argc, char* argv[]) {
    int sock;
    char score[BUF_SIZE];
    char grade;
    int str_len;

    struct sockaddr_in serv_addr;

    if(argc != 3) {
        printf("Usage : %s <IP> <PORT>\n", argv[0]);
        exit(1);
    }
}

```

```

}

sock = socket(PF_INET, SOCK_STREAM, 0);
if(sock == -1)
    error_handling("socket error");

memset(&serv_addr, 0, sizeof(serv_addr));
serv_addr.sin_family = AF_INET;
serv_addr.sin_addr.s_addr = inet_addr(argv[1]);
serv_addr.sin_port = htons(atoi(argv[2]));

if(connect(sock, (struct sockaddr*) &serv_addr, sizeof(serv_addr)) == -1)
    error_handling("connect error");
else {
    printf("연결하는 중입니다.....\n");
    printf(".....\n");
}

printf("학점 계산 프로그램입니다.\n");
// score 배열에 차례로 값 입력
printf("국어 점수를 입력하세요 : ");
scanf("%d", (int*) &score[0]);
printf("영어 점수를 입력하세요 : ");
scanf("%d", (int*) &score[1]);
printf("수학 점수를 입력하세요 : ");
scanf("%d", (int*) &score[2]);

write(sock, score, strlen(score));

read(sock, &grade, 1); // 서버로부터 학점 읽기

printf("당신의 학점은 %c 입니다.\n", grade); // 출력

close(sock);

return 0;
}

void error_handling(char* grade) {
    fputs(grade, stderr);
    fputc('\n', stderr);
    exit(1);
}

```

<주요 실행결과 이미지>

```
[seona@seona:~$ gcc server.c -o server ]  
[seona@seona:~$ gcc client.c -o client ]
```

-> 서버측

```
[seona@seona:~$ ./server 9190 ]  
총 점 : 220  
평균 : 73
```

-> 클라이언트측

```
[seona@seona:~$ ./client 127.0.0.1 9190 ]  
연결 하는 중 입니다 .....  
.....  
학 점 계 산 프 로 그 램 입 니 다 .  
[국 어 점 수 를 입 력 하 세 요 : 70 ]  
[영 어 점 수 를 입 력 하 세 요 : 90 ]  
[수 학 점 수 를 입 력 하 세 요 : 60 ]  
당 신 의 학 점 은 C 입 니 다 .
```