

Xilinx Zynq FPGA, TI DSP, MCU 기반의 프로그래밍 및 회로 설계 전문가 과정

강사 – Innova Lee(이상훈)

gcccompil3r@gmail.com

학생 – 장성환

redmk1025@gmail.com

*술게임 만들기

여러명 접속하여 즐길 수 있는 게임

시간제한(3 초)은 존재하고, 횡수 제한은 없음. (시간제한이 걸릴시에 횡수가 +1 됨)

서버는 횡수를 카운트 하고 있어야함.

fork()분할로 여러 사람이 동시에 서비스가 가능해야 함.

숫자는 1~3333 까지 업다운게임

* 서버

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>
#include <sys/socket.h>
#include <time.h>
#include <fcntl.h>
#include <sys/wait.h>
#include <signal.h>
```

```
typedef struct sockaddr_in si;
typedef struct sockaddr *sap;
int ans_cnt;
#define BUF_SIZE 1024
#define chop(str) str[strlen(str)-1]=0x00;
```

```
void err_handler(char *msg){
    fputs(msg, stderr);
    fputc('\n', stderr);
    exit(1);
}
```

```
}

int make_game(){
    int tmp;
    tmp =rand()%3333+1;
    return tmp;
}

char* start_game(char *msg,int res){
    int data = atoi(msg);
    char *tmp[3]={"정답입니다!\n","입력한 값이 작습니다!\n","입력한 값이 큼니다!\n"};

    if(data == res){
        return tmp[0];
    }

    else if(data < res){
        ans_cnt++;
        return tmp[1];
    }

    else if(data > res){
        ans_cnt++;
        return tmp[2];
    }
}

void sig_handler(int signo){
    ans_cnt++;
    printf("late!~");
}
```

```
int main (int argc, char **argv){
    int i, str_len;
    int serv_sock, clnt_sock;
    pid_t pid[5];
    int res;
    int status;

    char msg[BUF_SIZE];

    si serv_addr, clnt_addr;
    socklen_t clnt_addr_size;
    srand(time(NULL));

    if(argc != 2){
        printf("use: %s <port>\n", argv[0]);
        exit(1);
    }

    serv_sock = socket(PF_INET, SOCK_STREAM, 0);

    if(serv_sock == -1)
        err_handler("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, (sap)&serv_addr, sizeof(serv_addr)) == -1)
        err_handler("bind() error");
    if(listen(serv_sock, 5) == -1)
        err_handler("listen() error");
```

```
clnt_addr_size = sizeof(clnt_addr);
res = make_game();

printf("res : %d\n",res);

for(i =0; i<5; i++){

    pid[i] = fork();

    if(pid[i]>0){
        waitpid(-1,&status,WNOHANG);
    }
    else if(pid[i] ==0){ //실제 작동구간

        clnt_sock = accept(serv_sock, (struct sockaddr *)&clnt_addr, &clnt_addr_size);
        signal(SIGALRM,sig_handler);

        if(clnt_sock == -1)
            err_handler("accept() error");

        else
            printf("Connected Client %d\n", i+1);

        while(1){
            char *tmp;
            int tmp_len;

            alarm(3);
            str_len = read(clnt_sock, msg, BUF_SIZE);
            alarm(0);

            tmp = start_game(msg,res);
```

```
        tmp_len = strlen(tmp);
        write(clnt_sock, tmp, tmp_len);
        if(strcmp(tmp,"정답입니다!\n")==0)
            break;
    }

    char *tmp;
    int tmp_len;
    sprintf(tmp,"당신의 점수는 %d 회 입니다.\n",ans_cnt+1);
    tmp_len = strlen(tmp);
    write(clnt_sock,tmp,tmp_len);
    close(clnt_sock);
}

}
close(serv_sock);

return 0;
}
```

* 클라이언트

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>
#include <sys/socket.h>
```

```
typedef struct sockaddr_in si;
typedef struct sockaddr * sap;
```

```
#define BUF_SIZE 1024
```

```
void err_handler(char *msg){
    fputs(msg, stderr);
    fputc('\n', stderr);
    exit(1);
}
```

```
int main(int argc, char **argv){
```

```
    int sock;
    int str_len;
    si serv_addr;
    char *m = "Input Message(q to quit): ";
    char msg[BUF_SIZE];
```

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

```
sock = socket(PF_INET, SOCK_STREAM, 0);

if(sock == -1)
    err_handler("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, (sap)&serv_addr, sizeof(serv_addr)) == -1)
    err_handler("connect() error");
else
    printf("Connected.....");

for(;;){
    fputs("Input msg(q to QUIT): ", stdout);
    fgets(msg, BUF_SIZE, stdin);

    if(!strcmp(msg, "q\n") || !strcmp(msg, "Q\n"))
        break;
    write(sock, msg, strlen(msg));
    str_len = read(sock, msg, BUF_SIZE-1);
    if(str_len == -1)
        err_handler("read() error!");
    msg[str_len] = 0;
    printf("msg from serv: %s\n", msg);
}
close(sock);

return 0;
}
```


* 출력

2 개의 클라이언트를 켜고 출력을 확인.

```
sunghwan@HWAN:~/Documents/linux/test$ ./testc 127.0.0.1 1597
Connected.....Input msg(q to QUIT): 123
msg from serv: 입력한 값이 작습니다!

Input msg(q to QUIT): 4456
msg from serv: 입력한 값이 큼니다!

Input msg(q to QUIT): 1047
msg from serv: 정답입니다!
당신의 점수는 4회 입니다.

Input msg(q to QUIT): 65
msg from serv:
Input msg(q to QUIT): 45
```

3 초 경과하여 점수가 3 점이 아닌 4 점이 출력

```
sunghwan@HWAN:~/Documents/linux/test$ ./testc 127.0.0.1 1597
Connected.....Input msg(q to QUIT): 456
msg from serv: 입력한 값이 작습니다!

Input msg(q to QUIT): 7897
msg from serv: 입력한 값이 큼니다!

Input msg(q to QUIT): 1047
msg from serv: 정답입니다!
당신의 점수는 5회 입니다.

Input msg(q to QUIT): 65
msg from serv:
Input msg(q to QUIT): 56
```

3 초 2 번 경과하여 3 점이 아닌 5 점이 출력