

시스템프로그래밍



과 제 명 : Proxy 2-3

교 수 님 : 최상호 교수님

강의시간 : 월요일

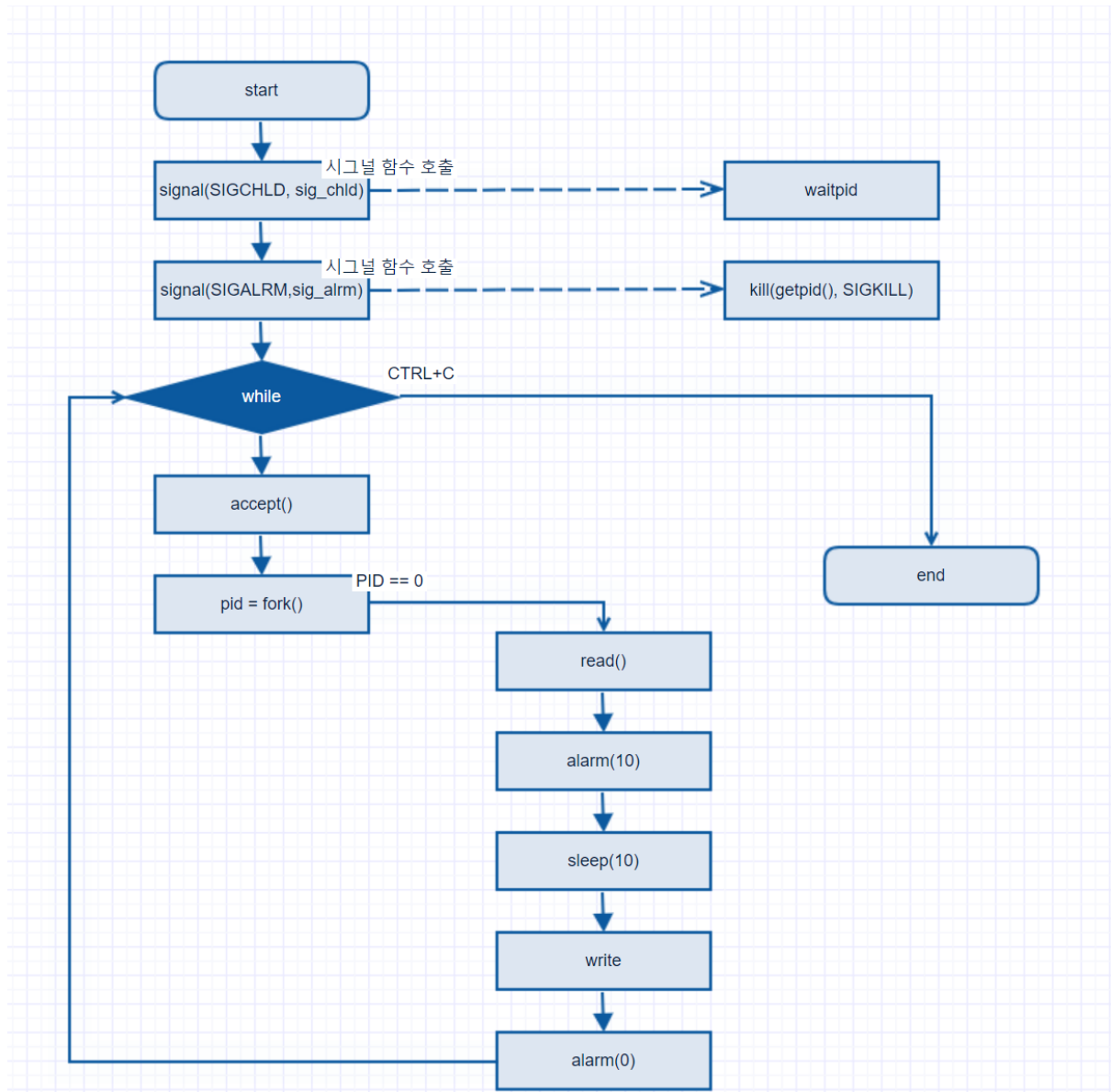
학 번 : 2019202100

이 름 : 하주영

Introduction

Signal() 함수를 활용하여 SIGCHLD, SIGALRM 과 같은 시그널을 처리한다. SIGCHLD는 자식 프로세스가 종료되면 호출되는 시그널로 디폴트 값은 무시이다. 하지만 이번 과제에서는 시그널함수를 사용하여 그 안에 waitpid로 자식프로세스의 종료를 기다리는 함수를 추가한다. SIGALRM은 alarm() 함수의 설정한 sec 경과후에 실행되는 시그널로 이번 과제에서는 10초동안 response를 받지 못하면 자식 프로세스를 종료할 수 있도록 시그널 함수를 작성한다.

Flow Chart



Pseudo Code

```
/*시그널 핸들러*/
```

```
handler()
```

```
    while(waitpid(-1, &status, WNOHANG);
```

```
sig_alrm()
```

```
    printf("=== no response ===");
```

```
    alarm(0);
```

```
    kill(getpid(), SIGKILL);
```

```
/*main 함수*/
```

```
signal(SIGCHLD, (void *)sig_chld);
```

```
signal(SIGALRM, (void *)sig_alrm);
```

```
while
```

```
    client_fd = accept(socket_fd, &client_addr, &len);
```

```
    pid = fork();
```

```
    if(pid == 0) /*자식 프로세스*/
```

```
        read(client_fd, buf, BUFFSIZE);
```

```
        alarm(10);
```

```
        #1-2 hit/miss 판별하고 로그파일에 기록
```

```
        sleep(10);
```

```
        write(client_fd, response_header, strlen(response_header));
```

```
        write(client_fd, response_message, strlen(response_message));
```

```
        alarm(0);
```

```
endwhile
```

Result Screen

```
kw2019202100@ubuntu:~/2-3$ ./proxy_cache
[192.168.80.128:20709] client was connected.
=====
Request from [192.168.80.128 : 20709]
GET http://www.naver.com/ HTTP/1.1
Host: www.naver.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:88.0) Gecko/20100101 Firefox/88.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1

=====
===== No Response =====
^C
```

네트워크 연결된 상태로 실행파일(proxy_cache)를 실행하고 web browser에 url을 입력한 다음 네트워크 연결을 끊으면 request는 전달되지만 response는 전달되지 않는다.

따라서 request에 해당되는 부분이 출력이 되고 No Response 문구가 출력된다.

Consideration

Signal alarm은 설정한 시간(초)이 지난 후 시그널 핸들러 수를 호출하는 시그널이다. 이번 과제요구사항은 10초안 HTTP response를 받지 못할 경우 화면에 "응답 없음"메시지를 출력하는 것이다. 이 때 주의사항이 있다.

지난과제인 ul을 입력하면 url을 request 형태로 서버에 전달하고 서버는 이 url을 처리하여 웹 브라우저에 response를 전달하는 프로그램에 '네트워크가 도중에 끊어졌을 때' 라는 상황을 가정하여 시그널 함수를 추가하는 것이다.

하지만 request부터 response까지의 동작이 너무 빠르기 때문에 sleep을 설정하여 임의로 alarm이 실행되도록 설정하였다.

해결방법은 지연시간을 늘리기 위해 대역폭을 줄이면 된다.

*대역폭: 대역폭은 시간에 따른 데이터 전송 속도를 결정합니다. 대역폭은 초당 전송할 수 있는 데이터의 양입니다.

Reference

<https://www.thefastcode.com/ko-krw/article/how-latency-can-make-even-fast-internet-connections-feel-slow>