



2-3-2

금3,4

과목명	시스템프로그래밍
담당교수	김태석 교수님
학과	컴퓨터정보공학부
학년	3학년
학번	2019202009
이름	서여지
제출일	21.05.15 (토)

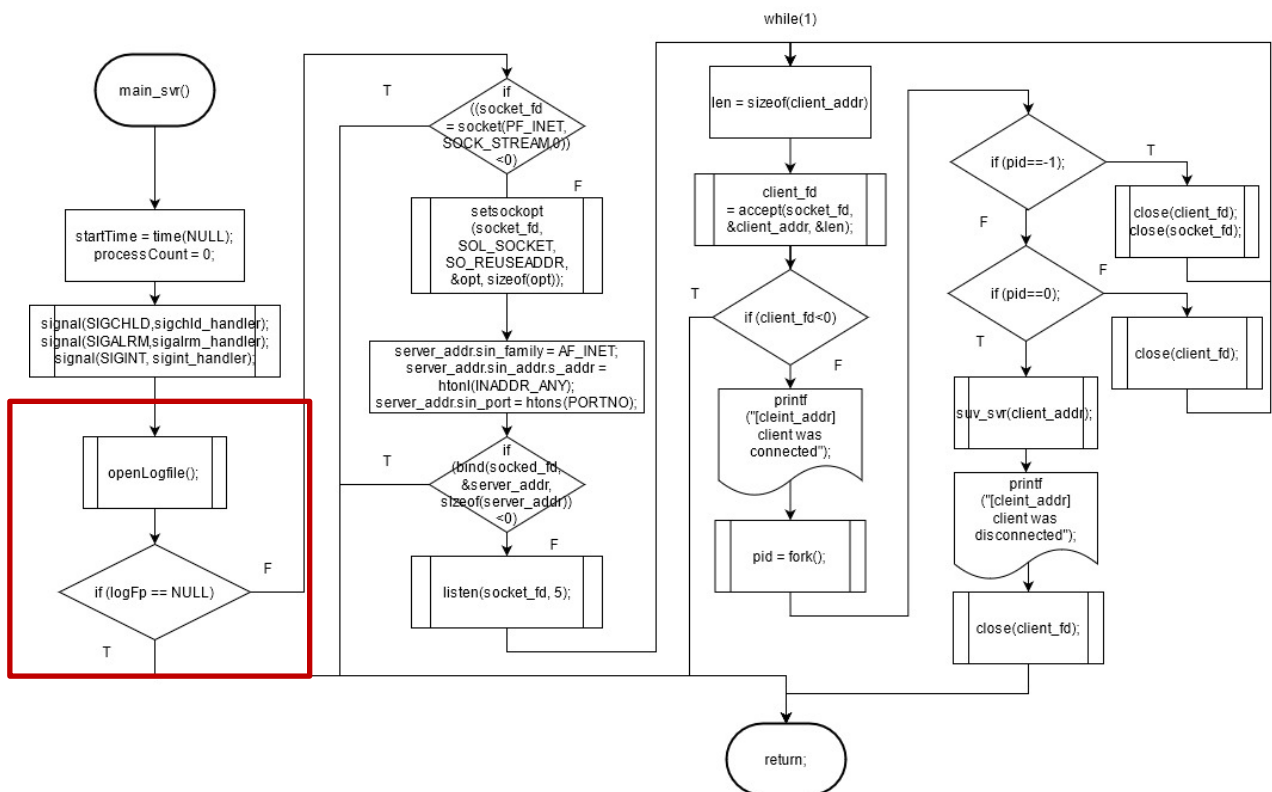
1. Introduction //과제 소개 - 5줄 내외(background 제외)

logfile을 생성하여 작성하는 기능을 추가한다. 웹서버에서 받은 모든 response정보를 캐시에 저장하지만, logfile에는 웹서버의 주소만을 기록한다. 전체 프로그램을 종료할 때 logfile을 기록하고 종료한다. 2-3 과제를 수행하며 웹서버에서 정보를 받는 기능을 대부분 구현했으므로 해당 기능의 오류수정과, logfile을 작성하는 부분과 SIGINT 핸들러를 추가로 작성하였다. logfile의 작성은 1-3과제에서 구현한 부분을 이용하였다.

2. Flow Chart //코드 작성 순서도

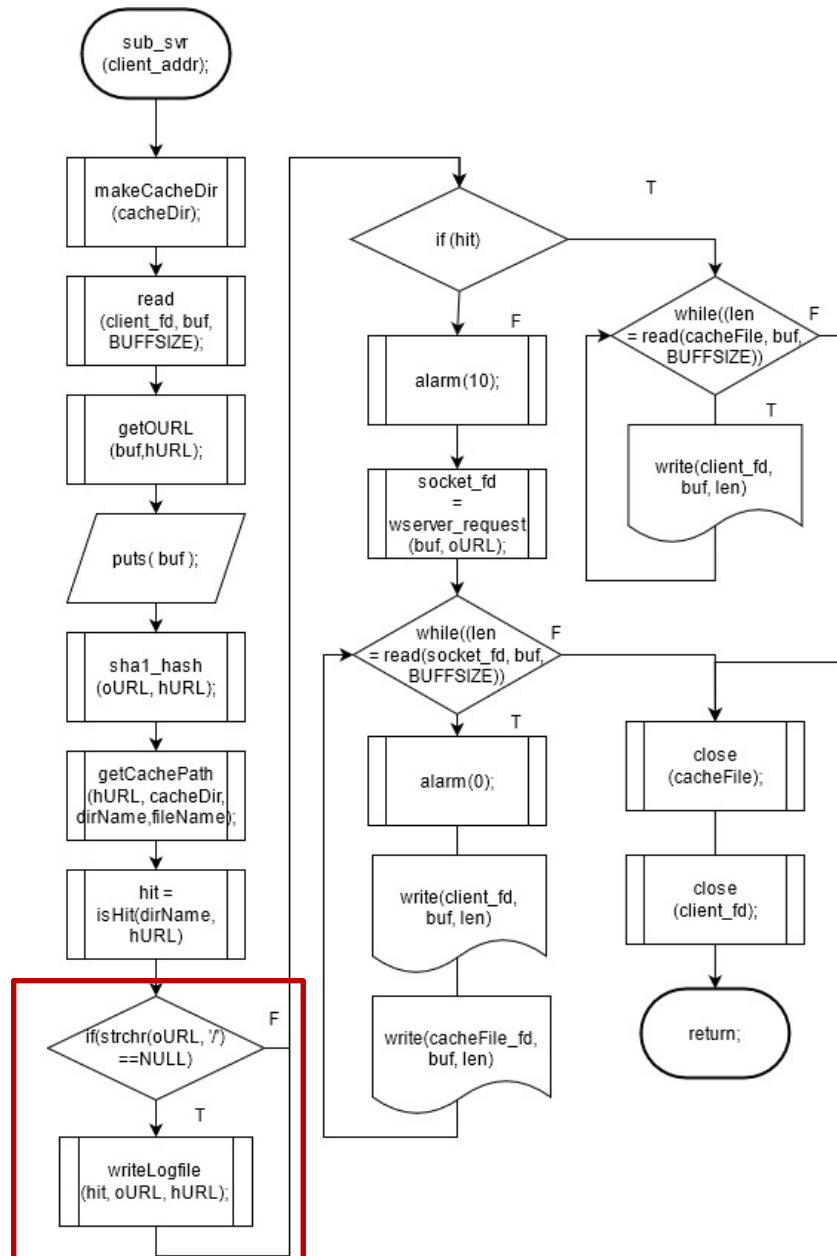
(1) main_svr

이전 프로젝트의 내용에서 SIGINT의 handler를 지정하는 내용과 logfile을 생성하는 부분이 추가되었다.



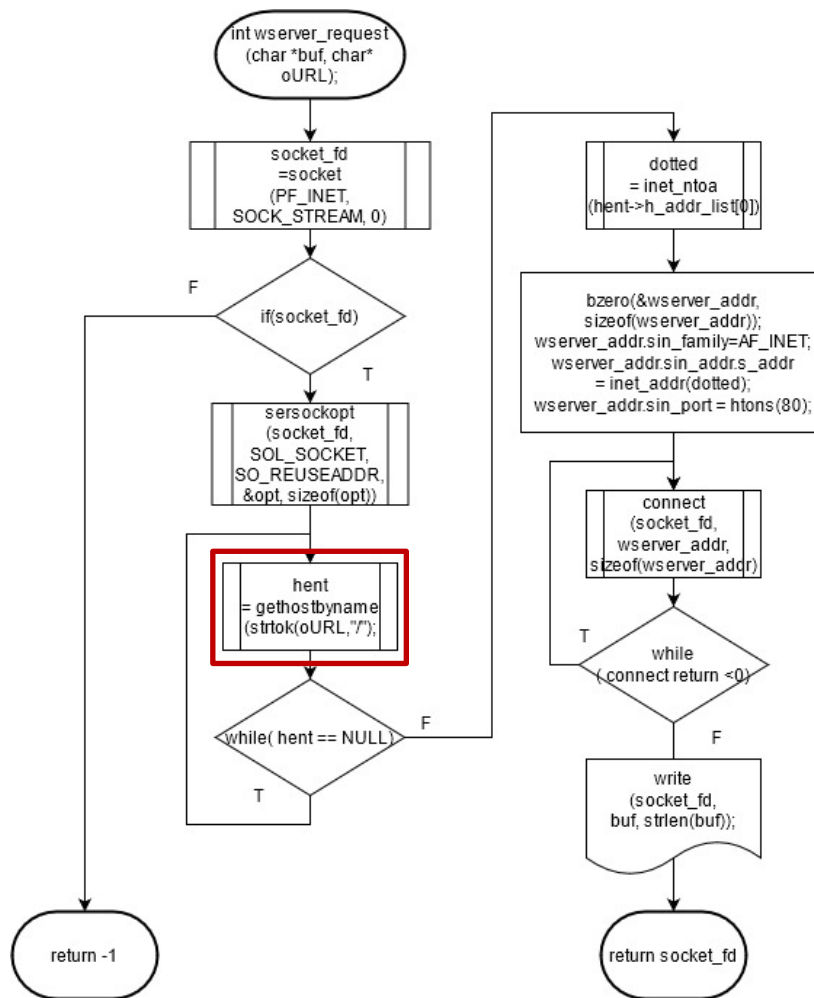
(2) sub_svr

client에 보내는 response message의 터미널 출력을 삭제하고 logfile을 작성하는 부분을 추가하였다.



(3) wserver_request

인자로 입력된 URL에 대하여 '/'을 토큰으로하는 strtok를 실행하여 웹서버의 주소를 찾는다.



3. Pseudo code //알고리즘

```

void main_svr(void) {
    startTime = time(NULL);
    processCount = 0;
    signal(SIGCHLD, sigchld_handler);
    signal(SIGALRM, sigalrm_handler);
    signal(SIGINT, sigint_handler);

    openLogfile();
    if(logfile open error) return;    //can't open logfile

    socket(PF_INET, SOCK_STREAM);
    if (socket error ) return;    // can't open socket
    setsockopt(socket_fd, SOL_SOCKET, SO_REUSEADDR, &opt, sizeof(opt));

    server_addr = 0;
    server_addr.sin_family = AF_INET;
    server_addr.sin_addr.s_addr = htonl(INADDR_ANY);
    server_addr.sin_port = htons(PORTNO);
  
```

```

bind(socket_fd, &server_addr, sizeof(server_addr))
if( bind error ) return;    // can't bind address

listen(socket_fd, 5);

while(true){
    client_addr = 0;
    len = sizeof(client_addr);
    client_fd = accept(socket_fd, &server_addr, &len)
    if(client_fd < 0)    return;    // accept err
    write ("[client addr] client was connected/n");
    processCount+ + ;
    pid = fork();
    if (fork err) {
        close(client_fd);
        close (socket_fd);
        continue;
    }
    if (sub process) {
        sub_svr(client_addr);
        write("[client addr] client was disconnectedWn");
        close(client_Fd);
        return;
    }
    close(client_fd);    //main process
}
close(socket_fd);
return;
}

void sub_svr(struct sockaddr_in  client_addr) {
    makeCacheDir(cacheDir);

    read(client_fd, buf, BUFSIZE);
    strcpy(tmp, buf);
    getOURL(tmp, oURL);

    sha1_hash(oURL, hURL);
    getCachePath(hURL, cacheDir, dirName, fileName);

    hit = isHit(dirName, hURL);
    if(oURL is webserver address) writeLogfile(hit, oURL, hURL);

    if(hit) {
        cacheFile = open(fileName, "r");
        while((read(cacheFile, buf, BUFSIZE))>0)
            write(client_fd, buf, len);
        close(cacheFile);
    }
    else {
        alarm(10);
        socket_fd = wserver_requset(buf, oURL);
        cacheFile = open(fileName, "w");
    }
}

```

```

        while((len = read(socket_fd, buf, sizeof(buf))){
            alarm(0);
            write(client_fd, buf, len);
            write(cacheFile, buf, len);
            buf = 0;
        }
        close(socket_fd);
        close(cacheFile);
    }
    close(client_fd);
    return;
}

int wserver_requset(char* buf, char* oURL){
    socket_fd = socket(PF_INET, SOCK_STREAM);
    if (socket error ) return;    // can't open socket
    setsockopt(socket_fd, SOL_SOCKET, SO_REUSEADDR, &opt, sizeof(opt));

    hent = gethostbyname(oURL);
    while (gethostbyname error) retry gethostbyname(strtok(oURL, "/")) after 1 sec
    dotted = inet_ntoa(hent->h_addr_list[0]);

    wserver_addr = 0;
    wserver_addr.sin_family = AF_INET;
    wsrever_addr.sin_addr.s_addr = dotted
    wsrever_addr.sin_port = htons(80);

    connect(socket_fd, &wserver_addr, sizeof(wserver_addr));
    while(connect error) retry connect after 1 sec

    write(socket_fd, buf, strlen(buf));
    return socket_fd;
}

```

4. 결과화면 //수행한 내용을 캡처 및 설명

firefox에서 ip설정을 변경한 이후 임의의 URL을 입력하여 miss인 경우 동작을 확인한다. 이후 동일한 URL을 다시 입력하여 hit인 경우의 동작을 확인한다. 생성된 cache와 logfile의 내용을 확인한다.

```

kw2019202009@ubuntu:~/Documents/0514/proxy23$ ./proxy_cache
[-2107922240: 24724] client was connected.

oURL: africanmoss.myspecies.info
=====
Request form [192.168.91.130 : 24724]
GET http://africanmoss.myspecies.info/ HTTP/1.1
Host: africanmoss.myspecies.info
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
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1

X◇
=====
**MISS**
connect: 157.140.2.32
[-2107922240: 25748] client was connected.

oURL: africanmoss.myspecies.info/sites/africanmoss.myspecies.info/57TkZCESl8zp_00VduSG6wH0S0__GhwE4qu0hVadFwC_xt42ge_nqKc1E-M0uzyo8YZAKaYUQg.css
=====
Request form [192.168.91.130 : 25748]
GET http://africanmoss.myspecies.info/sites/africanmoss.myspecies.info/57TkZCESl8zp_00VduSG6wH0S0__GhwE4qu0hVadFwC_xt42ge_nqKc1E-M0uzyo8YZAKaYUQg.css HTTP/1.1
Host: africanmoss.myspecies.info
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
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1

```

<처음 url을 입력한 결과>

```

oURL: africanmoss.myspecies.info/sites/africanmoss.myspecies.info/Ruwenzori_Vegetation.jpg?itok=RRD4AXyU
=====
Request form [192.168.91.130 : 8853]
GET http://africanmoss.myspecies.info/sites/africanmoss.myspecies.info/Ruwenzori_Vegetation.jpg?itok=RRD4AXyU HTTP/1.1
Host: africanmoss.myspecies.info
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:88.0) Gecko/20100101 Firefox/88.0
Accept: image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive
Referer: http://africanmoss.myspecies.info/
Cookie: has_js=1
Range: bytes=5097-
If-Range: "195a0-52a780b2576c0"

=====
**HIT**
[-2107922240: 8853] client was disconnected.
[-2107922240: 9365] client was connected.

oURL: africanmoss.myspecies.info/sites/africanmoss.myspecies.info/Seychelles.JPG?itok=4PpB1t0B
=====
Request form [192.168.91.130 : 9365]
GET http://africanmoss.myspecies.info/sites/africanmoss.myspecies.info/Seychelles.JPG?itok=4PpB1t0B HTTP/1.1
Host: africanmoss.myspecies.info
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:88.0) Gecko/20100101 Firefox/88.0
Accept: image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive
Referer: http://africanmoss.myspecies.info/
Cookie: has_js=1
Range: bytes=5097-
If-Range: "195a0-52a780b2576c0"

```

<동일한 url을 다시 입력한 결과>


```

[-2107922240: 9365] client was disconnected.
^Ckw2019202009@ubuntu:~/Documents/0514/proxy23$ cat ~/logfile/logfile.txt
[MISS] africanmoss.myspecies.info - [2021/04/14, 12:21:16]
[HIT] 1ef/9fcfaa1a3ee97db0b46534e817139d4e84b90 - [2021/04/14, 12:22:03]
[HIT] africanmoss.myspecies.info
**SERVER** [Terminated] run time: 75 sec, #sub process: 65
kw2019202009@ubuntu:~/Documents/0514/proxy23$

```

<logfile 결과>

5. 결론 및 고찰

저번에 작성한 과제에 logfile출력을 추가하였다. 1-3 과제에서 작성한 내용을 이용하여 큰 어려움 없이 작성할 수 있었다. 지난 2-3 구현에서 웹페이지의 여러 내용을 웹서버로부터 받아오는 과정에서 url에 별도의 처리를 하지 않아 각 파일의 이름이 gethostbyname함수의 인자로 전달되어 프로그램이 정상적으로 동작하지 않는 문제가 있었다. 이것을 strtok을 이용하여 수정하였고, 웹페이지가 정상적으로 출력되는 것을 확인할 수 있었다.

프로그램 전체가 종료되었을 때 마지막으로 logfile에 내용을 적어야 하는데, sub process가 SIGINT에 의해 종료되는 경우에도 동일한 내용이 반복적으로 출력되는 문제가 있었다. main process의 pid를 전역변수로 저장하고, sigint_handler에서 getpid를 이용하여 얻은 값을 서로 비교하여 일치하는 경우에만 logfile을 작성하도록 설계하여 문제를 해결할 수 있었다.