

2023년 1학기 시스템프로그래밍 & 시스템 프로그래밍 실습

Assignment 2-3

System Software Laboratory
College of Software and Convergence
Kwangwoon Univ.

Assignment 2-3

다중 접속과 접근 제어를 지원하는 웹 서버 프로그램 작성

- 이전 과제의 웹 서버(Assignment 2-2)의 요구조건을 만족해야 함

다중 접속 지원

- 클라이언트와 소켓 연결된 이후의 작업을 새로운 프로세스에서 수행하여 동시에 다수의 클라이언트가 접속할 수 있도록 지원
- 클라이언트와 접속 연결 및 해제 시 마다 클라이언트들의 정보를 아래와 같이 출력 (①)
- 10초에 한 번씩 연결되었던 클라이언트의 정보를 일괄 출력 (②)
- 총 연결 횟수 ("Number of request(s)")는 서버 프로그램 실행 이후의 누적 값 유지
- 하단의 접속 기록은 최대 10개의 최근 기록만 출력(최신 시간 순서대로 정렬)

```
===== New Client =====
IP : 127.0.0.1
Port : 18123
=====

===== Disconnected Client =====
IP : 127.0.0.1
Port : 18123
=====

===== New Client =====
IP : 127.0.0.1
Port : 18127
=====

===== Disconnected Client =====
IP : 127.0.0.1
Port : 18127
=====

===== Connection History =====
Number of request(s) : 2
No.   IP       PID      PORT    TIME
1     127.0.0.1  2878     18127   Sat May 11 17:59:08 2019
2     127.0.0.1  2876     18123   Sat May 11 17:59:08 2019
```

ex) History 출력 부분

- No.: 서버 실행 후 몇 번째 request인지
- 최근 접속 기록이 먼저 출력

* NO. : 해당 클라이언트가 몇 번째로 서버에 연결된 것인지

* IP : 클라이언트의 IP 주소

* Port : 해당 클라이언트를 위한 클라이언트의 Port 번호

* PID : 해당 클라이언트를 위한 프로세스의 ID

* Time : 서버-클라이언트가 연결된 시간

Assignment 2-3 (cont'd)

- 접근 제어

- 미리 허용한 IP를 가진 사용자만 서버에 접속할 수 있도록 허가
 - 현재 연결을 요청한 클라이언트의 IP를 기준으로 판단
- 접근 가능한 클라이언트의 목록을 파일("accessible.usr")로 유지
 - Dotted-decimal 형태의 IP(ex. 128.134.52.62)를 명시
 - 10진수 수 대신에 *가 들어가 있을 경우도 처리

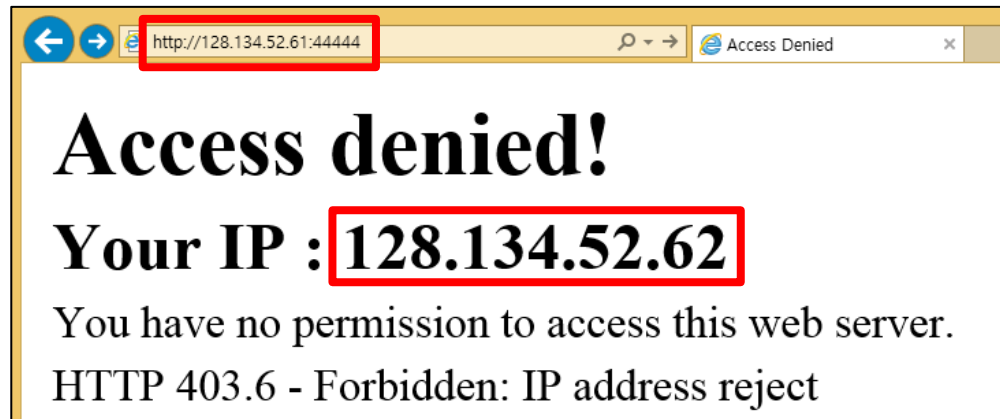
- 예시

```
kks@sslab-splab-ubuntu:~$ cat accessible.usr
128.134.52.61
192.168.*.1
128.*.*.62
kks@sslab-splab-ubuntu:~$
```

Assignment 2-3 (cont'd)

- 허가되지 않은 IP에서 접속할 경우, 에러 메시지가 포함된 페이지 출력

```
kks@ssslab-splab-ubuntu:~$ cat accessible usr
128.134.52.61
kks@ssslab-splab-ubuntu:~$
```



- 허가 되지 않은 IP(e.g. 128.134.52.62)가 웹 서버(e.g. 128.134.52.61:44444)에 접근하고자 할 때 접근 제한 메시지를 출력
- 해당하는 문자열을 전부 출력해야 함
 - 단, 디자인(글자 크기 등)은 채점 요소가 아님

Assignment 2-3 (cont'd)

- **Code Requirements**

- 이전 과제 부분에 문제가 있는 경우 감점
- 출력 형식에 맞지 않으면 감점

- **Makefile Requirements**

- 실행 파일이 “adv_server”로 생성되도록 Makefile 작성
- “\$ make” 를 통해 실행파일이 생성되지 않는 경우, 0점

Report Requirements

- **Ubuntu 16.04.5 Desktop 64bits 환경에서 채점**
- **Copy 발견 시 0점 처리**
- **보고서 구성**
 - **보고서 표지**
 - 수업 명, 과제 이름, 담당 교수님, 학번, 이름 필히 명시
 - 과제 이름 → assignment2-3
 - **과제 내용**
 - Introduction
 - 과제 소개 - 4줄 이상(background 제외) 작성
 - Flow chart (appendix 참고)
 - Pseudo code (appendix 참고)
 - 결과화면
 - 수행한 내용을 캡처 및 설명
 - 고찰
 - 과제를 수행 결과 분석 및 해석 해결한 부분, 문제점 발생에 대한 요약
 - Reference
 - 과제를 수행하면서 참고한 내용을 구체적으로 기록
 - 강의자료만 이용한 경우 생략 가능

Report Requirements

▪ Softcopy Upload

- 제출 파일
 - 보고서 + 소스파일 하나의 압축 파일로 압축하여 제출(tar.gz)
 - 보고서(.pdf 파일 변환)
 - 소스코드
 - 학번_adv_server.c
 - Makefile
 - 실행파일 명: adv_server
 - 소스 코드, 실행파일명 다르게 작성 시 감점
 - 소스파일 comment 작성 없을 시 감점
- Tar 압축 및 해제 방법
 - 압축 시 → tar -zcvf [압축 파일명].tar.gz[폴더 명]
 - 해제 시 → tar -zxvf 파일명.tar.gz
- 보고서 및 압축 파일 명 양식
- **Web2_3_수강분류코드_학번** 으로 작성

수강요일	이론1 월5수6	이론2 월2수1	실습1 금12	실습2 금56	실습3 금78
수강분류 코드	A	B	C	D	E

- 예시-이론 월5 수6 수강하는 학생인 경우
 - 보고서: Web2_3_A_20231234567.pdf
 - 압축 파일 명: Web2_3_A_20231234567.tar.gz

Report Requirements

- 실습 수업을 수강하는 학생인 경우

- 실습 과목에 과제를 제출(.tar.gz)
- 이론 과목에 간단한 .txt 파일로 제출

 실습수업때제출했습니다. 2022-08-29 오후 3:58 텍스트 문서 OKB

- 이론 과목에 .txt 파일 미 제출 시 감점

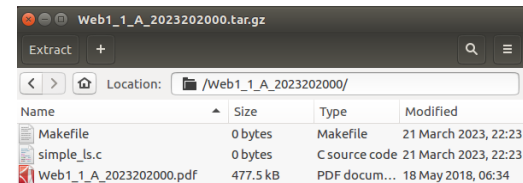
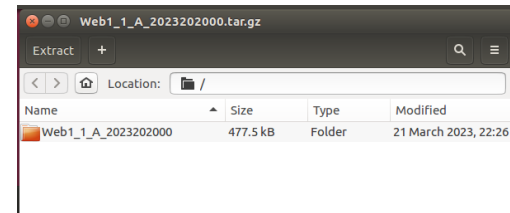
- 과제 제출

- KLAS – 강의 과제 제출
- 2023년 5월 10일 수요일 23:59까지 제출
 - 딜레이 받지 않음
 - 제출 마감 시간 내 미제출시 해당 과제 0점 처리
 - 교내 서버 문제 발생 시, 메일로 과제 제출 허용

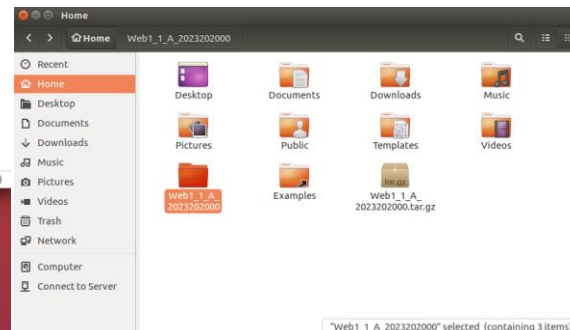
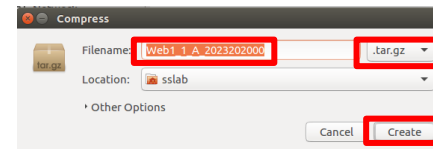
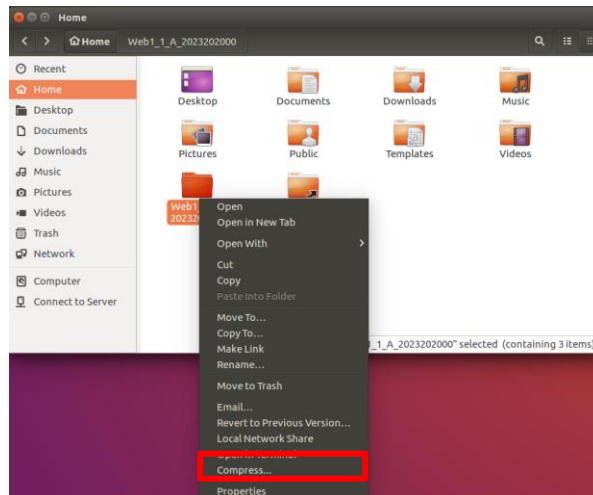
과제 제출 양식

- 소스파일, 보고서, Makefile만 하나의 tar.gz 로 압축, 아래 두 가지 방법 중 택 1
- 압축 명령어 사용

```
sslslab@ubuntu: ~/Web1_1_A_2023202000
sslslab@ubuntu:~$ ls
Desktop  Downloads      Music  Public  Videos
Documents examples.desktop Pictures Templates Web1_1_A_2023202000
sslslab@ubuntu:~$ cd Web1_1_A_2023202000/
sslslab@ubuntu:~/Web1_1_A_2023202000$ ls
Makefile simple_ls.c Web1_1_A_2023202000.pdf
sslslab@ubuntu:~/Web1_1_A_2023202000$
sslslab@ubuntu:~$ tar -zcvf Web1_1_A_2023202000.tar.gz Web1_1_A_2023202000/
Web1_1_A_2023202000/
Web1_1_A_2023202000/Makefile
Web1_1_A_2023202000/simple_ls.c
Web1_1_A_2023202000/Web1_1_A_2023202000.pdf
sslslab@ubuntu:~$ ls
Desktop  examples.desktop  Public  Web1_1_A_2023202000
Documents Music           Templates Web1_1_A_2023202000.tar.gz
Downloads Pictures         Videos
```



- 디렉토리에서 직접 압축



2023년 1학기 시스템프로그래밍실습

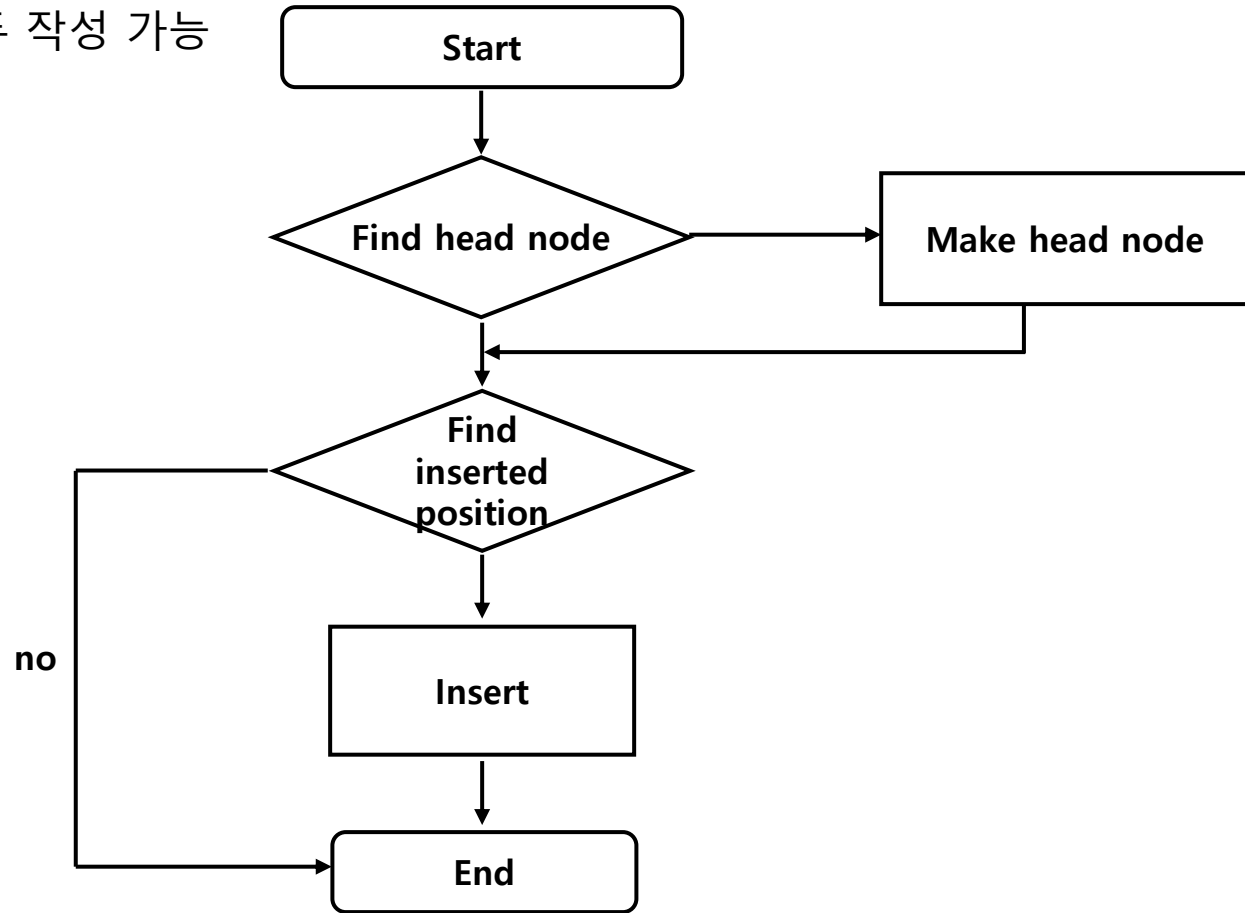
Appendix

System Software Laboratory
College of Software and Convergence
Kwangwoon Univ.

보고서 작성 요령 (1/2)

■ Algorithm – Flow Chart (Each function)

- E.g.
- 국문, 영문 모두 작성 가능



보고서 작성 요령 (2/2)

■ Algorithm – Pseudo Code

- E.g.
- 국문, 영문 모두 작성 가능

```
FixHeap(Node *root, Key k)
{
    Node vacant, largerChild;
    vacant = root;
    while( vacant is not leaf ) {
        largerChild = the child of vacant with the larger key;
        if( k < largerChild's Key ) {
            copy largerChild's key to vacant;
            vacant = largerChild;
        }
        else exit loop;
    }
}
```

Set grade counter to one

While grade counter is less than or equal to ten

 Input the next grade

 Add the grade into the total

Set the class average to the total divided by ten

Print the class average

Comment 작성 요령 (1/3)

■ File Head Comment

- E.g.
- 국문, 영문 모두 작성 가능

```
////////////////////////////////////  
// File Name      : Main.c                      //  
// Date           : 2023/03/01                  //  
// Os             : Ubuntu 16.04 LTS 64bits      //  
// Author         : Hong Gil Dong               //  
// Student ID     : 2023123456                  //  
// ----- //  
// Title : System Programming Assignment #1-1 (proxy server) //  
// Description : ...                            //  
////////////////////////////////////
```

Comment 작성 요령 (2/3)

■ Function Head Comment

- E.g.
- 국문, 영문 모두 작성 가능

```
////////////////////////////////////  
// InsertNode                                                    //  
// =====                                                    //  
// Input: Node* -> Insert Node,                                  //  
//           Node* -> Column node before insert node          //  
//           Node* -> Row node before insert node              //  
//           (Input parameter Description)                       //  
// Output: int   - 1 success                                     //  
//           0 fail                                             //  
//           (Out parameter Description)                         //  
// Purpose: Inserting node                                       //  
////////////////////////////////////
```

Comment 작성 요령 (3/3)

■ In-line Comment

- e.g.
- 국문, 영문 모두 작성 가능

```
if( pRowPos->pNextRow != pRowPos ) {  
    pTemp->pNextRow = pRowPos->pNextRow;           // pTemp set next row  
    if( !( pRowPos->pNextRow->bHead ) ){  
        pRowPos->pNextRow->NodeItem.pPrevRow = pTemp;  
    } // end of if  
} // end of if  
else {  
    pTemp->pNextRow = pRowPos;                       // pTemp set next row  
} // end of else  
pTemp->NodeItem.pPrevRow = pRowPos;                 // pTemp set previous row  
pRowPos->pNextRow = pTemp;
```