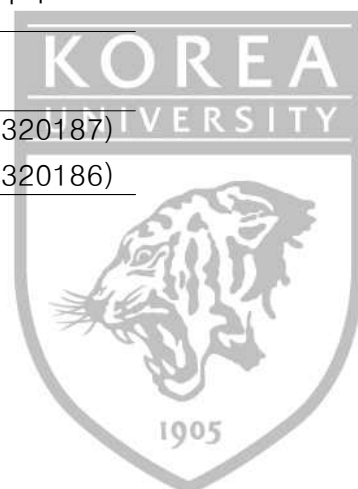


시스템 프로그래밍

2차과제 보고서

- pthread를 이용한
client-side socket programming -

제출일	2020.11.06.	학과	컴퓨터학과
과목	시스템 프로그래밍	팀	4
Freeday	0일 사용	팀원	홍성윤(2016320187) 윤상준(2016320186)



○ 조원 정보

- 2016320187 홍성윤(조장)
- 2016320186 윤상준(조원)

○ 작성한 소스코드에 대한 설명

- pthread를 이용한 패킷 수신 프로그램을 작성했습니다. fgets함수와 strtok 함수를 이용해 사용자로부터 정보를 입력받아 저장합니다. 입력한 포트 개수가 10보다 크거나, 같은 포트 번호를 5개보다 많이 입력하면 다시 입력을 받도록 했습니다. port_num 배열을 동적으로 할당해서 포트 번호를 저장하도록 했습니다.
- for 문과 pthread_create 함수를 이용해서 입력한 포트 번호의 개수만큼 thread를 생성했습니다. pthread_create 함수는 미리 선언한 client_threads배열과 client_creator 함수 포인터, client_creator 함수의 인자로 사용할 포트번호를 인자로 받습니다. 부모 thread가 자식 thread가 완료될 때까지 기다리게 하기 위해서 pthread_join을 호출합니다.
- client_creator 함수는 client가 소켓을 생성하고 연결하는 함수입니다. 먼저 socket 함수로 소켓을 생성합니다. 그 후, client가 접속할 서버의 address 정보를 server_addr 구조체에 담습니다. 이때 입력받은 포트 번호 정보도 저장합니다. connect 함수를 호출해 서버와 connect를 시도합니다. connect 가 성공적으로 되면, client와 서버는 서로 data 전송을 할 준비가 됩니다.
- server로부터 받은 data를 기록하기 위해 <포트번호-fd>.txt 파일을 생성합니다. read함수를 통해 버퍼에 정보를 읽어옵니다. timeval 구조체와 gettimeofday 함수를 통해 현재 시간에 대한 정보를 출력할 수 있게 했습니다. atsign_counting 함수를 통해 '@' 개수를 확인할 수 있게 하고 개수가 5개 이상이면 at_checking 함수를 호출해 '@@@@@' 까지만 출력하고 while문을 빠져나오게 했습니다. open 한 txt파일을 닫고 소켓도 닫으면 통신이 종료됩니다.
- 통신이 잘 되었는지 확인하기 위해 확인 메시지를 출력합니다.



○ pthread를 사용하는 이유

- 각각의 소켓이 parallel하게 통신을 하게 해주기 위해서 pthread를 사용합니다. 이를 통해, 각 포트에 대한 통신이 동시에 진행되고, 불필요한 코드 중복을 방지할 수 있습니다.

○ 과제 수행 시 어려웠던 점과 해결 과정

- 네트워크 주소

: connect 함수 호출에 필요한 서버의 address 정보 중 네트워크 주소를 찾는데 어려움이 있었습니다. 처음에는 배경지식이 부족해 네트워크 주소에 임의의 주소를 입력해서 connection이 계속해서 실패했습니다. ifconfig 명령어로 확인한 네트워크 주소를 입력하지 않았다는 것을 알게 되었고 connection을 성공적으로 할 수 있었습니다.

- strtok

: 사용자에게 입력 받은 문자열 정보를 포트 개수와 포트 번호로 저장하기 위해 strtok 함수를 사용하는 과정에서 어려움이 있었습니다. 원하는 대로 strtok가 진행되지 않았고, 조사를 해본 결과 strtok 함수는 자체적으로 문자열을 읽던 위치를 기억하기 때문에 strtok(NULL, " ")을 통해서 잘 문자열이 잘라지는 다는 사실을 알게 되었습니다.

- time 정보

: time 정보를 위한 timeval 구조체를 pointer로 선언함으로 인해 segmentation fault가 발생했습니다. timeval 구조체를 pointer를 사용하지 않고 접근해서 이 문제를 해결했습니다.

