**TCP/IP 기말 대체 과제**

**202044030 배은빈**

**[ 쓰레드 기반 TCP/IP 네트워크 프로그래밍 소스 ]**

**쓰레드**

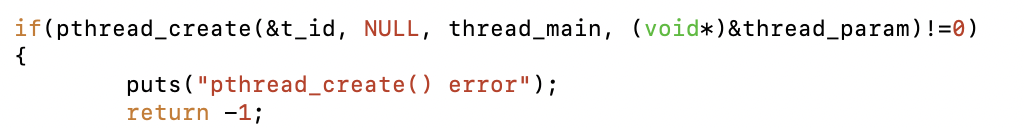
**🡪 어떤 프로그램에서 프로세스가 실행되는 흐름의 단위**

**Ex) main 함수**

****

**thread\_main 함수**

**🡪 쓰레드의 main 함수이며 이를 가리켜 쓰레드 함수라고 함**

****

**pthread\_create**

**🡪 현재 실행되고 있는 프로세스에서 새로운 스레드를 생성하는 함수**

**&t\_id : 성공적으로 함수가 호출되면 thread ID가 저장 됨**

**NULL : 스레드의 특성을 정의함, 기본적으로 NULL을 지정**

**thread\_main : 어떤 로직을 할지 함수 포인터를 매개변수로 받음**

**(void\*)&thread\_param : start\_routine에 전달될 인자를 말하며,**

**start\_routine에서 이 인자를 변환하여 사용함**

**🡪 성공 시 thread 초기화함과 동시에 0을 반환**

**실패 시 thread 인자가 설정되지 않고 에러 값을 반환**

**//성공적으로 호출이 되면 0을 반환하고 실패하면 에러 숫자 반환함**

**텍스트이(가) 표시된 사진

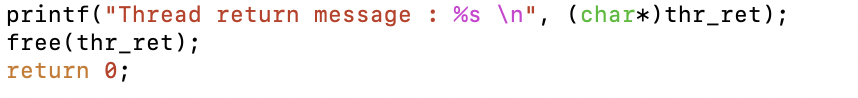
자동 생성된 설명**

**pthread\_join**

**🡪 main이나 부모 thread에서 자식 thread가 종료할 때까지 대기하는 함수**

**t\_id : 생성한 thread의 id**

**&thr\_ret : 해당 thread가 종료되면 return받을 변수이며 리턴 받을 값이 없으면. Null을 넣으면 됨**

****

**//thr\_ret의 메모리 해제**

**free( ) 함수**

**🡪 thr\_ret에 해당되는 메모리를 해제하는 함수**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**malloc( ) 함수**

**🡪 size 바이트의 메모리를 할당하여 반환하는 함수**

**sleep( ) 함수**

**🡪 프로세스의 종료를 막기 위해서 sleep 함수 호출을 통해서**

**1초간 main 함수의 실행 지연시키고 있음**

**[뮤텍스(mutex) 기반의 동기화 코드]**

**뮤텍스(mutex)**

**🡪 각 스레드가 동시에 동일한 주소에 접근하는 현상을 예방하기 위해**

**한 스레드가 주소에 접근했을 때 다른 스레드의 주소 접근을 막는 것**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**//thread ID를 저장할 배열**

**//mutex 생성 함수**

**pthread\_mutex\_init**

**🡪 mutex를 동적으로 초기화하는 함수**

**&mutex : 생성 시 – 뮤텍스의 참조 값 저장을 위한 변수의 주소 값 전달**

**소멸 시 – 소멸하고자 하는 뮤텍스의 참조 값을 저장하고 있는**

**변수의 주소 값 전달**

**NULL : 생성하는 뮤텍스의 특성정보를 담고 있는 변수의 주소 값 전달**

**별도 특성 지정 안하면 NULL 전달**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**i가 짝수이면 thread\_inc 함수 호출하며 새로운 스레드 생성**

**홀수이면 thread\_des 함수 호출하며 새로운 스레드 생성**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**//각각의 thread가 끝날때까지 주프로세스가 기다림**

**//mutex 소멸 함수**

**pthread\_mutex\_destroy**

**🡪 사용이 끝난 mutex 변수를 해제하는 함수**

**🡪 malloc으로 동적으로 선언한 mutex 변수는 free()를 호출하기 전에 꼭 먼저 호출해야 함**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**//임계영역 벗어났으므로 다른 쓰레드 접근 허용**

**//Mutex를 이용하고 다른 쓰레드 접근 막음**

**//짝수 스레드가 실행 할 함수**

**pthread\_mutex\_lock , pthread\_mutex\_unlock**

**🡪 mutex를 이용하여 임계구역을 진입할 때 그 코드 구역을 잠그고 다시 임계구역이 끝날 때 다시 풀어 다음 스레드가 진입할 수 있도록 함**

**//홀수 스레드가 실행 할 함수**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**//임계영역 벗어났으므로 다른 쓰레드**

**접근 허용**

**//mutex를 이용하고 다른 쓰레드 접근 막음**

**[세마포어 기반 동기화 코드]**

**세마포어(semaphore)**

**🡪 임계영역에 접근할 쓰레드의 수를 제한하는 방법**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**//세마포어 2번 종료**

**//세마포어 1번 종료**

**//id\_t2이 종료 될 때까지 기다림**

**//id\_t1이 종료 될 때까지 기다림**

**//accu 함수 호출하는 id\_t2인 쓰레드 실행**

**//read 함수 호출하는 id\_t1인 쓰레드 실행**

**//semm\_two 세마포어 생성**

**//semm\_one 세마포어 생성**

**0으로 초기화 했으므로 키를 갖고 있지 않기 때문에 임계영역에 접근이 불가능함**

**1로 초기화 했으므로 키를 갖고 있기 때문에 임계영역에 접근이 가능함**

**sem\_init**

**🡪 첫번째 인자 : 세마포어 객체를 초기화 할 세마포어 받음**

**두번째 인자 : 0을 주지 않으면 항상 에러코드 반환**

**세번째 인자 : 세마포어를 몇으로 초기화 할 지 의미함**

**sem\_destroy**

**🡪세마포어와 관련된 것들을 소멸시킴**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**//세마포어 2번의 값을 0으로 막고 num을 쓰레드 2번의 임계영역에 넣음**

**//num을 읽고 난 후 세마포어 1번의 값을 1로 해서 막는것을 해제함**

**//num값을 읽고 난 후 세마포어 1번을 막는 것을 해제함**

**//값을 읽어들이는 함수**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**//sum에 값이 정상적으로 담긴 후에 세마포어 2번을 막는 것을 해제함**

**//세마포어 1번을 막아서 num값이 잘 넘어오는지 확인함**

**sem\_wait**

**🡪 세마포어를 하나 감소시키는 역할을 함**

**sem\_post**

**🡪 세마포어 값을 증가시키는 역할을 함**