운영체제 3주차 과제

2016156007

소프트웨어학과

김영민

1.

#include<sys/types.h>

#include<stdio.h>

#include<unistd.h>

int main() {

pid t pid;//프로세스 아이디

pid = fork(); //자식 프로세스 생성

if (pid < 0) {// pid 값은 0보다 큰 정수이므로 0보다 작을경우 에러

fprintf(stderr, "Fork Failed");

return 1;

}

else if (pid == 0) { //자식 프로세스

execlp("/bin/ls", "ls", NULL);//시스템 호출

}

else {//부모 프로세스 ,pid 가 0 보다 큰경우

wait(NULL);// 자식 프로세스가 완료될 때 까지 부모 프로세스 대기

printf("child complete"); //자식 프로세스 종료

}

return 0;

}}

2.

#include<stdio.h>

#include<windows.h>

int main(VOID)

{

STARTUPINFO si;

PROCESS\_INFORMATION pi;

//메모리를 할당

ZeroMemory(&si, sizeof(si));

si.cb = sizeof(si);

ZeroMemory(&pi, sizeof(pi));

//자식 프로세스 생성

if (!CreateProcess(NULL, //audfuddj fkdls tkdyd

"C:\\WINDOWS\\system32\\mspaint.exe", // 명령어 라인

NULL, // 프로세스를 상속하지 말 것

NULL, //스레드 핸들을 상속하지 말 것

FALSE, //핸들 상속 디제이블

0, // 생성 플래그 없음

NULL,// 부모 환경 블록 사용

NULL, //부모 프로세스가 존재하는 디렉터리 사용

&si,

&pi))

{

fprintf(stderr, "Create Process Failed");

return -1;

}

//부모 프로세스가 자식 프로세스가 끝나기를 기다림

WaitForSingleObject(pi.hProcess, INFINITE);

printf("Child Complete");

//핸들 닫기

CloseHandle(pi.hProcess);

CloseHandle(pi.hThread);

}