|  |
| --- |
|  |
| 시스템 프로그래밍 |
|  |
| PrintDir |

**1분반**

2010136017

김덕균

시스템 프로그래밍

PrintDir

# Printdir.c 소스 및 주석

#include <unistd.h>

#include <stdio.h>

#include <dirent.h>

#include <string.h>

#include <sys/stat.h>

#include <stdlib.h>

void printdir(char \*dir, int depth) {

DIR \*dp;

struct dirent \*entry;

struct stat statbuf;

if ((dp = opendir(dir)) == NULL) {

fprintf(stderr, "cannot open directory : %s\n", dir);

return;

}

//현 프로세스의 작업 디렉토리를 dir로 변경

chdir(dir);

//현 디렉토리의 내용이 끝날때까지 반복

while ((entry = readdir(dp)) != NULL) {

//읽은 엔트리의 상태를 확인하는 펑션

lstat(entry->d\_name, &statbuf);

//히든 폴더 혹은 파일이거나 상위폴더를 가르키는 파일은 무시

if(entry->d\_name[0]=='.'){ continue; }

//디렉토리인지 확인 파일이면 출력

if (S\_ISDIR(statbuf.st\_mode)) {

//파일명 + / 로 사용자가 디렉토리 구분을 하게 해주고

//depth로 칸을 띄어 구분을 지어준다

printf("%\*s%s / \n", depth, "", entry->d\_name);

//디렉토리면 하위 내용이 있기에 해당 펑션을 재호출하여 재귀처리

printdir(entry->d\_name, depth + 4);

}

else {

printf("%\*s%s\n", depth, "", entry->d\_name);

}

}

chdir("..");

closedir(dp);

}

int main() {

printf("Directory scan of ./:\n");

printdir("./", 0);

printf("done.\n");

exit(0);

}

# 실행 화면

