1. 소스코드

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <unistd.h>  #include <dirent.h>  #include <fcntl.h>  #include <string.h>  #include <sys/stat.h>  #define DIRECTORY\_SIZE MAXNAMLEN  int main(int argc, char \*argv[]) {  struct dirent \*dentry;  struct stat statbuf;  char filename[DIRECTORY\_SIZE + 1];  DIR \*dirp;  if(argc < 2) {  fprintf(stderr, "usage: %s <directory>\n", argv[0]);  exit(1);  }  //argv[1] 디렉터리를 open하고 argv[1] 디렉터리로 이동  if((dirp = opendir(argv[1])) == NULL || chdir(argv[1]) == -1) {  fprintf(stderr, "opendir, chdir error for %s\n", argv[1]);  exit(1);  }  //디렉터리를 read한다.  while((dentry = readdir(dirp)) != NULL) {  //inode가 없을 경우 다음 항목으로 이동  if(dentry->d\_ino == 0)  continue;    //디렉터리/파일 이름 복사  memcpy(filename, dentry->d\_name, DIRECTORY\_SIZE);  //파일 정보 추출  if(stat(filename, &statbuf) == -1) {  fprintf(stderr, "stat error for %s\n", filename);  break;  }  //모드 정보 중 파일 정보만 추출하여(S\_IFMT 이용) 정규파일인지 비교  //정규파일이면 파일이름과 크기도 출력  if((statbuf.st\_mode & S\_IFMT) == S\_IFREG)  printf("%-14s %ld\n", filename, statbuf.st\_size);  else//정규파일이 아니면 이름만 출력  printf("%-14s\n", filename);  }  exit(0);  } |

2. 실행결과

ssu\_oslab.c와 ssu\_oslab\_exec는 정규 파일이므로 파일이름과 함께 파일크기도 출력

