

# 시스템 호출 프로그래밍 (파일 정보 관련)

안양대학교 컴퓨터공학과 하은용

파일 정보 시스템 호출 예제

# 내용

2

- 파일 정보관련 시스템 호출
- stat(), fstat(), lstat()

# 파일 정보관련 시스템 호출 함수

3

## □ 함수 프로토타입

```
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>
```

```
int stat ( const char *path , struct stats *buf );
int fstat ( int filedes , struct stats *buf );
int lstat ( const char *path , struct stats *buf );
```

# 구조체 *stat* 정의

4

struct stat

```
{  
mode_t      st_mode    // 파일종류 및 모드(접근권한)  
dev_t       st_dev     // 파일 시스템 장치 번호: major()+minor()  
ino_t       st_ino     // inode 번호  
dev_t       st_rdev     // 특수 파일인 경우 장치 번호  
n_link      st_nlink   // 하드 링크 개수  
uid_t       st_uid     // 소유자의 사용자 ID  
gid_t       st_gid     // 소유자의 그룹 ID  
off_t       st_size    // 바이트 단위 크기  
time_t      st_atim    // 최종 파일 접근 시간  
time_t      st_mtim    // 마지막으로 파일의 내용이 수정된 시간  
time_t      st_ctim    // 마지막으로 파일의 상태가 변경된 시간  
blksize_t   st_blksize // 최적의 I/O 블록 크기  
blkcnt_t    st_blocks  // 파일 저장을 위해 할당된 블록 개수  
}
```

# 파일 종류 및 검사 매크로

5

## □ 변수 st\_mode : 파일 종류

파일 종류	st_mode 값	8진수 값	파일종류 검사 매크로
파일타입마스크	S_IFMT	0170000	
소켓	S_IFSOCK	0140000	S_ISSOCK(m)
심볼릭링크	S_IFLNK	0120000	S_ISLNK(m)
일반 파일	S_IFREG	0100000	S_ISREG(m)
블록 장치	S_IFBLK	0060000	S_ISBLK(m)
디렉터리	S_IFDIR	0040000	S_ISDIR(m)
문자 장치	S_IFCHR	0020000	S_ISCHR(m)
FIFO 파일	S_IFIFO	0010000	S_ISFIF(m)

# 파일 접근 권한 1

6

## □ 변수 st\_mode : 파일 접근 권한

st_mode값	8진수 값	설명
S_ISUID	0004000	set-user-ID 비트
S_ISGID	0002000	set-group-ID 비트
S_ISVTX	0001000	sticky 비트
S_IRWXU	00700	소유자(User)의 접근 권한 검사 마스크
S_IRUSR	00400	소유자 읽기 권한
S_IWUSR	00200	소유자 쓰기 권한
S_IXUSR	00100	소유자 실행 권한

# 파일 접근 권한 2

7

## □ 변수 st\_mode : 파일 접근 권한

S_IRWXG	00070	그룹(Group)의 접근 권한 검사 마스크
S_IRGRP	00040	그룹 읽기 권한
S_IWGRP	00020	그룹 쓰기 권한
S_IXGRP	00010	그룹 실행 권한
S_IRWXO	00007	다른 사용자(Other)의 접근 권한 검사 마스크
S_IROTH	00004	다른 사용자 읽기 권한
S_IWOTH	00002	다른 사용자 쓰기 권한
S_IXOTH	00001	다른 사용자 실행 권한

# 파일 정보 : *stat.c* 프로그램 1

8

## □ 프로그램 : stat.c

```
#include      <stdio.h>
#include      <sys/types.h>
#include      <sys/stat.h>

int main (int argc, char *argv[])
{
    struct stat info;

    if ( stat(argv[1], &info) != -1 )
    {
        printf("파일명   : %s\n", argv[1]);
        printf("파일시스템 : %d\n", info.st_dev);
        printf("장치번호  : %d\n", info.st_rdev);
        printf("inode 번호 : %d\n", info.st_ino);
    }
}
```



# 파일 정보 : *stat.c* 프로그램 2

9

```
printf("파일 종류와 모드 : %d\n", info.st_mode);
printf("하드링크 수 : %d\n", info.st_nlink);
printf("소유자 ID : %d\n", info.st_uid);
printf("그룹 ID : %d\n", info.st_gid);
printf("파일 크기(바이트단위) : %d\n", info.st_size);
printf("블록 크기 : %d\n", info.st_blksize);
printf("할당 블록 수 : %d\n", info.st_blocks);
printf("마지막 접근시간 : %d\n", info.st_atime);
printf("마지막 상태 변경시간 : %d\n", info.st_ctime);
printf("마지막 내용 수정시간 : %d\n", info.st_mtime);
}
else
    fprintf(stderr, "stat() 오류\n");
return 0;
}
```

# 파일 종류 : *stat\_type.c* 프로그램 1

10

## □ 프로그램 : *stat\_type.c*

```
#include <sys/types.h>
#include <sys/stat.h>
#include <time.h>
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    struct stat sb;

    if (stat(argv[1], &sb) == -1) {
        perror("stat");
        exit(EXIT_FAILURE);
    }
}
```

# 파일 종류 : *stat\_type.c* 프로그램 2

11

```
printf("파일 종류: ");  
switch ( sb.st_mode & S_IFMT ) {  
case S_IFBLK:    printf("block device\n"); break;  
case S_IFCHR:    printf("character device\n"); break;  
case S_IFDIR:    printf("directory\n"); break;  
case S_IFIFO:    printf("FIFO/pipe\n"); break;  
case S_IFLNK:    printf("symlink\n"); break;  
case S_IFREG:    printf("regular file\n"); break;  
case S_IFSOCK:   printf("socket\n"); break;  
default:        printf("unknown?\n"); break;  
}  
exit(EXIT_SUCCESS);  
}
```

# 파일 권한 : *stat\_mode.c* 프로그램 1

12

## □ 프로그램 : stat\_mode.c

```
#include <sys/types.h>
#include <sys/stat.h>
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    struct stat fileStat;

    if( stat(argv[1], &fileStat) < 0)
        exit(EXIT_FAILURE);
```

# 파일 권한 : *stat\_mode.c* 프로그램 2

13

```
printf("File Permissions: \t");  
printf( (S_ISDIR(fileStat.st_mode)) ? "d" : "-");  
printf( (fileStat.st_mode & S_IRUSR) ? "r" : "-");  
printf( (fileStat.st_mode & S_IWUSR) ? "w" : "-");  
printf( (fileStat.st_mode & S_IXUSR) ? "x" : "-");  
printf( (fileStat.st_mode & S_IRGRP) ? "r" : "-");  
printf( (fileStat.st_mode & S_IWGRP) ? "w" : "-");  
printf( (fileStat.st_mode & S_IXGRP) ? "x" : "-");  
printf( (fileStat.st_mode & S_IROTH) ? "r" : "-");  
printf( (fileStat.st_mode & S_IWOTH) ? "w" : "-");  
printf( (fileStat.st_mode & S_IXOTH) ? "x" : "-");  
printf("\n\n");  
exit(EXIT_SUCCESS);
```

```
}
```

# 요약

14

- 파일 정보 관련 시스템 호출
- `stat()`, `fstat()`, `lstat()`
- `struct stat {}`
- 파일 종류, 접근권한 프로그래밍