

설계과제 1 명세 : Soongsil SINDEX

Linux System Programming, School of CSE, Soongsil University, Spring 2022

○ 개요

- ssu_index는 지정한 파일 또는 디렉토리의 이름을 탐색해서 파일 및 디렉토리의 속성을 비교하는 프로그램

○ 목표

- 유닉스/리눅스 컴퓨팅 환경에서 파일 속성, 디렉토리 속성을 비교하여 출력하는 프로그램을 작성하고 시스템 프로그램 설계 및 응용 능력을 향상시킴

○ 팀 구성

- 개인별 프로젝트

○ 보고서 제출 방법

- 설계과제는 “보고서.hwp”(개요, 상세설계 구현방법, 결과 및 소스코드와 실행결과가 함께 있는 워드 (hwp 또는 MS-Word) 파일)와 “소스코드”(makefile, obj, *.c, *.h 등 컴파일하고 실행하기 위한 모든 파일)를 포함시켜야 함
- 모든 설계과제 결과물은 “#P설계과제번호_학번_버전.zip”(예. #P2_20160000_v1.0.zip)형태로 (zip프로그램으로 압축하여) 파일 이름을 명명하고, classroom.google.com에 제출해야 함. 단, 서버 다운으로 이메일(2022.oslab1@gmail.com과 2022.oslab2@gmail.com)로 제출 시 이메일 제목은 “#P설계과제번호_학번_버전”이어야 하고, 내용은 없이 “#P설계과제번호_학번_버전.zip)만 첨부하면 됨
- 버전 이름은 대문자 V와 함께 integer를 1부터 incremental 증가시키면서 부여하면 됨. (예. V1, V2, V3 ...), 단 처음 제출 시는 버전 번호를 붙이지 않아도 되면 두 번째부터 V1를 붙여 제출하면 됨
- 압축파일 내 “보고서” 디렉토리와 “소스코드” 디렉토리 2개 만들어 제출해야 함
- 제출한 압축 파일을 풀었을 때 해당 디렉토리에서 컴파일 및 실행이 되어야 함(특정 디렉토리에서 실행해야 할 경우는 제외). 해당 디렉토리에서 컴파일이나 실행되지 않을 경우, 설계과제 제출 방법(파일명, 디렉토리명, 컴파일 시에 포함되어야 할 파일 등)을 따르지 않는 경우 해당 과제 배점의 50% 감점
✓ 단, 첫 번째 설계과제 제출 방법을 따르지 않을 경우 경고만 할 것이며 감점을 하지 않을 것임. 한 번 경고에도 불구하고 설계과제 제출 방법을 따르지 않으면 무조건 해당 과제 배점의 50% 감점
- 기타 내용은 강의계획서 참고

○ 제출 기한

- 3월 23일(수) 오후 11시 59분 59초 (서버 시간 기준 1시간까지 지연 허용)

○ 보고서 양식

- 보고서는 다음과 같은 양식으로 작성

- | |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ol style="list-style-type: none">1. 과제 개요 (3점) // 위 개요를 더 상세하게 작성2. 상세 설계 (5점) // 함수 기능별 흐름도(순서도) 반드시 포함3. 구현 방법 설명 (5점) // 함수 프로토타입 반드시 포함4. 실행 결과 (2점) // 테스트 프로그램의 실행 결과 캡처 및 분석5. 소스코드 (85점) // 주석, 컴파일 여부 (10점), 실행 여부 (75점) |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

○ ssu_index 프로그램 기본 사항

- ssu_index는 주어진 디렉토리 내 특정 파일(이하 정규 파일과 디렉토리 파일 모두 가능)과 이름 및 크기가 동일한 정규 파일(이하 정규 파일과 디렉토리 파일 모두 가능)을 찾고 비교하는 프로그램
- ssu_index의 첫 번째 인자는 찾고자 하는 파일이며 ssu_index의 두 번째 인자는 탐색을 시작할 디렉토리 경로
- ssu_index에 내장된 find 명령어를 통해 두 번째 인자부터 두 번째 인자의 전체 서브디렉토리까지 재귀적으로 첫 번째 인자를 탐색하여 첫 번째 인자의 이름과 크기 모두 동일한 파일을 찾아 해당 파일의 속성을 출력
 - ✓ 파일의 속성은 stat 구조체 이용
 - ✓ 첫 번째 인자로 지정된 파일이 디렉토리 파일인 경우 모든 하위 디렉토리 내 모든 파일(. .. 제외하고 서브디렉토리까지 포함) 크기를 합산하여 출력
- ssu_index에서 동일한 이름과 크기의 파일이 존재할 경우, 특정 파일의 인덱스(index) 번호를 입력하면 선택된 파일의 속성 및 내용을 비교(diff)
 - ✓ 이름과 크기가 동일한 파일(디렉토리) 중 사용자가 비교하고 싶은 파일의 인덱스 번호를 입력하면 두 파일의 차이를 출력

○ 설계 및 구현

- 가) 프롬프트 출력
 - ✓ 프롬프트 모양 : “학번)” 문자 출력 (ex) 20201234))
 - ✓ 프롬프트에서는 ssu_index 내장 명령어인 find, exit, help 명령어만 실행
 - ✓ 이외 명령어 실행 시 자동으로 help를 실행시킨 것과 동일한 결과가 출력되어야 함
 - ✓ 엔터 키만 입력 시 프롬프트 재출력
- 나) 내장명령어 1, find [FILENAME] [PATH]
 - ✓ 파일(디렉토리)이름과 디렉토리 경로 입력 시 입력한 디렉토리부터 탐색을 시작하여 이름과 크기가 동일한 모든 파일(디렉토리)을 리스트로 출력
 - ✓ 파일 리스트에서 특정 파일(디렉토리)을 선택하면 해당 파일(디렉토리)과 입력한 파일(디렉토리) 내용의 차이점을 출력
 - ✓ [FILENAME]
 - 현재 또는 특정 디렉토리 내에 존재하는 파일(디렉토리) 이름(절대경로와 상대경로 모두 가능해야 함)
 - FILENAME 입력이 없거나 존재하지 않는 파일(디렉토리)일 경우, 에러 처리 후 프롬프트 출력
 - ✓ [PATH]
 - 탐색을 시작할 디렉토리 경로(절대경로와 상대경로 모두 가능해야 함)
 - [PATH] 인자에 루트(“/”)를 입력할 경우, 파일 탐색 시 권한 문제가 발생할 수 있으므로 root 권한으로 실행해야 함
 - [PATH] 인자에 입력이 없거나 존재하지 않는 파일인 경우, 에러 처리 후 프롬프트 출력
 - ✓ find 명령어 실행 시 출력은 다음과 같음
 - 첫째 줄에는 파일 리스트의 순번과 파일 정보를 다음과 같은 순서대로 행에 출력
 - index(탐색한 파일의 순번), Mode(10bit, 파일 종류 및 접근 권한), Blocks(할당된 블록 수), Links(하드링크 수), UID(사용자 ID), GID(그룹 ID), Access(최종 접근 시간), Change(최종 상태 변경 시간), Modify(최종 수정 시간), Path(파일의 절대 경로)
 - 둘째 줄에는 입력한 [FILENAME]의 정보(index, Mode, Blocks, Links, UID, GID, Access, Change, Modify, Path) 출력, 단 index는 0으로 출력
 - 셋째 줄부터는 입력한 [FILENAME]과 동일한(이름과 크기가 동일한 파일 탐색 결과) 파일의 정보 (index, Mode, Blocks, Links, UID, GID, Access, Change, Modify, Path) 출력. 단, 디렉토리 간 탐색은 제약은 없음 (Depth First나 Breadth First 중 하나 구현하면 됨)
 - 파일 리스트는 파일의 절대 경로가 아스키 코드 소팅 순서에 맞게 출력
 - 파일 리스트 출력 후 “>>” 문자 출력하여 사용자의 입력을 대기
 - 탐색 결과 이름과 크기가 같은 파일이 없을 경우 “(None)” 출력 후 프롬프트 출력

예. 파일 탐색 결과 출력 (1) - 동일한 크기와 이름의 파일이 3개 나온 경우										
<pre>\$ sudo ./ssu_index 20221234> find a / Index Size Mode Blocks Links UID GID Access Change Modify Path 0 478 -rw-r--r-- 8 1 1001 1001 22-02-27 21:01 22-02-27 20:47 22-02-27 06:53 /home/OSLAB/P1/a 1 478 -rw-r--r-- 8 1 1001 1001 22-02-27 21:01 22-02-27 20:47 22-02-27 06:50 /home/OSLAB/P1/dir1/a 2 478 -rw-r--r-- 8 1 1001 1001 22-02-27 21:15 22-02-27 20:47 22-02-27 06:53 /home/OSLAB/P1/dir2/a 3 478 -rw-r--r-- 8 1 1001 1001 22-02-27 21:06 22-02-27 20:47 22-02-27 06:50 /home/OSLAB/P1/dir2/dir1/a >> █</pre>										

예. 파일 탐색 결과 출력 (2) - 동일한 크기와 이름의 파일이 없는 경우										
<pre>\$ sudo ./ssu_index 20221234> find aa / Index Size Mode Blocks Links UID GID Access Change Modify Path 0 0 -rw-rw-r-- 0 1 1001 1001 22-02-27 21:13 22-02-27 21:13 22-02-27 21:13 /home/OSLAB/P1/aa (None) 20221234></pre>										

✓ >> [INDEX] [OPTION ...]

- [INDEX]

- 파일 리스트에서 비교하고자 하는 파일의 순번(인덱스 번호)
- INDEX 입력이 없거나 존재하지 않을 경우, 에러 처리 후 다시 입력 대기
- 사용자가 선택한 [INDEX]에 해당하는 정규 파일과 index가 0인 정규 파일(즉, find 명령어의 첫 번째 인자 [FILENAME]에 해당하는 파일)의 내용을 비교하여 차이점 출력
 - 1. 사용자가 선택한 [INDEX]에 해당하는 파일과 index가 0인 파일의 내용이 차이가 없을 경우 아무 것도 출력하지 않음
 - 2. 사용자가 선택한 [INDEX]에 해당하는 파일과 index가 0인 파일의 내용이 차이가 있을 경우 차이점을 출력
 - 3. 정규 파일 비교 시 차이점 출력 방법
 - 정규 파일은 1023개 이하의 문자를 포함하는 파일만 가능
 - 두 파일을 라인 단위로 비교하여 차이점을 출력
 - 파일 끝에 개행이 없을 경우 “\NNo newline at end of file” 출력

	[FILENAME]과 [INDEX]의 차이		
	[INDEX]에 새로운 내용이 추가된 경우	[FILENAME]의 내용이 [INDEX]에서 삭제된 경우	[FILENAME]의 내용이 [INDEX]에서 변경된 경우
범위	<i>o1an1,(n2)</i>	<i>o1(o2)dn1</i>	<i>o1(o2)cn1,(n2)</i>
설명	[FILENAME]의 o1번째 라인 다음에 [INDEX]의 n1(~n2) 번째 라인의 내용이 추가됨	[FILENAME]의 o1(~o2)번째 라인의 내용이 [INDEX]에는 n1번째 라인 다음에 삭제됨	[FILENAME]의 o1(~o2)번째 라인의 내용이 [INDEX]의 n1(~n2)번째 라인의 내용으로 교체됨
출력 형태	> “[INDEX] 추가내용”	< “[FILENAME] 삭제내용”	> “[FILENAME] 변경내용” --- < “[INDEX] 변경내용”

예. 파일 비교 예시 (여기서 숫자는 라인 넘버, vi(m)에서 :set nu한 결과로 파일의 내용과는 무관)	
/home/OSLAB/P1/a 파일 내용	/home/OSLAB/P1/dir1/a 파일 내용

<pre> 1 This part of the 2 document has stayed the 3 same from version to 4 version. It shouldn't 5 be shown if it doesn't 6 change. Otherwise, that 7 would not be helping to 8 compress the size of the 9 changes. 10 11 This paragraph contains 12 text that is outdated. 13 It will be deleted in the 14 near future. 15 16 It is important to spell 17 check this dokument. On 18 the other hand, a 19 misspelled word isn't 20 the end of the world. 21 Nothing in the rest of 22 this paragraph needs to 23 be changed. Things can 24 be added after it. </pre>	<pre> 1 This is an important 2 notice! It should be 3 located at this document! 4 5 This part of the 6 document has stayed the 7 same from version to 8 version. It shouldn't 9 be shown if it doesn't 10 change. Otherwise, that 11 would not be helping to 12 compress anything. 13 14 It is important to spell 15 check this document. On 16 the other hand, a 17 misspelled word isn't 18 the end of the world. 19 Nothing in the rest of 20 this paragraph needs to 21 be changed. 22 23 This paragraph contains 24 important additions 25 to this document. </pre>
출력 결과 (1)	
<pre> 20221234> find a / Index Size Mode Blocks Links UID GID Access Change Modify Path 0 477 -rw-r--r-- 8 1 1001 1001 22-02-28 22:12 22-02-28 22:11 22-02-28 20:14 /home/OSLAB/P1/a 1 477 -rw-r--r-- 8 1 1001 1001 22-02-28 22:14 22-02-28 22:13 22-02-28 22:13 /home/OSLAB/P1/dir1/a 2 477 -rw-r--r-- 8 1 1001 1001 22-02-28 22:12 22-02-28 22:12 22-02-28 22:12 /home/OSLAB/P1/dir2/a 3 477 -rw-r--r-- 8 1 1001 1001 22-02-28 22:12 22-02-28 22:13 22-02-28 22:13 /home/OSLAB/P1/dir2/dir1/a >> 1 0a1,4 > This is an important > notice! It should be > located at this document! > 8,14c12 < compress the size of the < changes. < < This paragraph contains < text that is outdated. < It will be deleted in the < near future. --- > compress anything. 17c15 < check this dokument. On --- > check this document. On 23,24c21,25 < be changed. Things can < be added after it. \ No newline at end of file --- > be changed. > > This paragraph contains > important additions > to this document. \ No newline at end of file 20221234> █ </pre>	
/home/OSLAB/P1/a 파일 내용	/home/OSLAB/P1/dir1/a 파일 내용

<pre> 1 The Way that can be told of is not the eternal Way; 2 The name that can be named is the eternal name. 3 The Nameless is the origin of Heaven and Earth; 4 The Named is the mother of all things. 5 Therefore let there always be non-being, 6 so we may see their subtlety, 7 And let there always be being, 8 so we may see their outcome. 9 The two are the same, 10 But after they are produced, 11 they have different names. </pre>	<pre> 1 The Nameless is the origin of Heaven and Earth; 2 The named is the mother of all things. 3 4 Therefore let there always be non-being, 5 so we may see their subtlety, 6 And let there always be being, 7 so we may see their outcome. 8 The two are the same, 9 But after they are produced, 10 they have different names. 11 They both may be called deep and profound. 12 Deeper and more profound, 13 The door of all subtleties! </pre>
출력 결과 (2)	
<pre> 20221234> find a / Index Size Mode Blocks Links UID GID Access Change Modify Path 0 401 -rw-r--r-- 8 1 1001 1001 22-02-28 21:48 22-02-28 21:46 22-02-28 21:46 /home/OSLAB/P1/a 1 401 -rw-r--r-- 8 1 1001 1001 22-02-28 21:48 22-02-28 21:46 22-02-28 21:46 /home/OSLAB/P1/dir1/a 2 401 -rw-r--r-- 8 1 1001 1001 22-02-28 21:50 22-02-28 21:47 22-02-28 21:47 /home/OSLAB/P1/dir2/a 3 401 -rw-r--r-- 8 1 1001 1001 22-02-28 21:50 22-02-28 21:47 22-02-28 21:47 /home/OSLAB/P1/dir2/dir1/a >> 1 1,2d0 < The Way that can be told of is not the eternal Way; < The name that can be named is the eternal name. 4c2,3 < The Named is the mother of all things. --- > The named is the mother of all things. > 11c10,13 < they have different names. \ No newline at end of file --- > they have different names. > They both may be called deep and profound. > Deeper and more profound, > The door of all subtleties! \ No newline at end of file 20221234> █ </pre>	

✓ >> [INDEX] [OPTION ...]

- 사용자가 선택한 [INDEX]에 해당하는 디렉토리 파일과 index가 0인 디렉토리 (즉, find 명령의 첫 번째 [FILENAME]에 해당하는 디렉토리 파일)의 내용을 비교하여 차이점 출력

- 두 디렉토리의 하위 파일 또는 하위 디렉토리의 이름을 비교하여 차이점을 출력
 - ☞ 두 디렉토리의 하위 정규 파일들 중 이름이 동일하나,
 - (1) 내용이 다른 경우 “diff [현재 적용된 OPTION] [입력 하위 파일 경로] [대상 하위 파일 경로]” 출력 후 차이점 출력
 - (2) 파일 종류가 다른 경우 “File [입력 하위 파일 경로] is a [directory/regular file] while file [대상 하위 파일 경로] is a [directory/regular file]” 출력
 - ☞ 두 디렉토리의 하위 디렉토리 이름이 동일한 경우 “Common subdirectories : [입력 하위 디렉토리 경로] and [대상 하위 디렉토리 경로]” 출력
 - ☞ 단, [INDEX] [OPTION ...]에서 [OPTION ...]에 -r이 없을 경우 재귀적으로 하위 디렉토리 모두 비교하지 않고 해당 디렉토리 내 정규 파일과 디렉토리 파일의 단순 이름 비교만 수행
 - ☞ 단, [INDEX] [OPTION ...]에서 [OPTION ...]에 -r이 있을 경우 재귀적으로 하위 디렉토리 내 전체 정규 파일과 디렉토리 파일 비교 수행
- 정규 파일의 이름 또는 디렉토리 이름이 다른 것은 “Only in [디렉토리명]: [파일명]” 출력
- 정규 파일의 이름 또는 디렉토리 이름 비교는 아스키코드 순으로 진행하고 결과 출력
- 비교를 마치면 프롬프트 출력

- [OPTION]

- OPTION은 공백으로 구분하여 입력하며 다음 4개의 OPTION은 동시에 사용 가능 (예. q s i r 띄어서 옵션 사용, 순서 상관 없음)
- OPTION이 존재하지 않을 경우나 공백 없이 여러 OPTION이 동시에 사용된 경우, 예러 처리 후 다시 입력 대기
 - ☞ q : 두 파일 내용이 다른 경우 차이점 출력 없이 알림
 - “Files [입력 하위 파일 경로] and [대상 하위 파일 경로] differ” 출력
 - ☞ s : 두 파일 내용이 동일한 경우 알림
 - “Files [입력 하위 파일 경로] and [대상 하위 파일 경로] are identical” 출력
 - ☞ i : 대소문자 구분 없이 비교
 - ☞ r : 디렉토리 비교 시 하위 디렉토리를 전부 재귀적으로 탐색하여 모든 파일 비교

예. 디렉토리 비교 예시

현재 작업 디렉토리 구조에서 ./dir1과 ./dir2/dir1을 비교할 경우 (tree는 리눅스 명령어)

\$ tree

```
.
├── a
├── aa
├── dir1
│   ├── a
│   ├── b
│   ├── c
│   ├── d
│   └── e
│       └── g
├── dir2
│   ├── a
│   └── dir1
│       ├── a
│       ├── b
│       ├── d
│       ├── e
│       └── g
│           └── f
├── ssu_index
└── ssu_index.c
```

7 directories, 13 files

출력 결과

```
20221234> find dir1 /
Index Size Mode      Blocks Links UID  GID Access          Change          Modify          Path
0      527 drwxrwxr-x 8         4   1001 1001 22-02-27 20:47 22-02-27 20:47 22-02-27 18:31 /home/OSLAB/P1/dir1
1      527 drwxrwxr-x 8         4   1001 1001 22-02-27 21:09 22-02-27 21:09 22-02-27 21:09 /home/OSLAB/P1/dir2/dir1
>> 1
diff dir1/b dir2/dir1/b
2,3c2,3
< banana
< carrot
\ No newline at end of file
---
> BANANA
> cherry
\ No newline at end of file
Only in dir1: c
File dir1/d is a directory while file dir2/dir1/d is a regular file
Common subdirectories: dir1/e and dir2/dir1/e
Only in dir2/dir1: f
20221234> █
```

예. OPTION q 출력

```
20221234> find a /
Index Size Mode      Blocks Links UID  GID Access          Change          Modify          Path
0      478 -rw-r--r-- 8         1   1001 1001 22-02-27 21:01 22-02-27 20:47 22-02-27 06:53 /home/OSLAB/P1/a
1      478 -rw-r--r-- 8         1   1001 1001 22-02-27 21:01 22-02-27 20:47 22-02-27 06:50 /home/OSLAB/P1/dir1/a
2      478 -rw-r--r-- 8         1   1001 1001 22-02-27 18:14 22-02-27 20:47 22-02-27 06:53 /home/OSLAB/P1/dir2/a
3      478 -rw-r--r-- 8         1   1001 1001 22-02-27 21:06 22-02-27 20:47 22-02-27 06:50 /home/OSLAB/P1/dir2/dir1/a
>> 1 q
Files a and dir1/a differ
```

```

예. OPTION s 출력
20221234> find a /
Index Size Mode          Blocks Links UID  GID Access          Change          Modify          Path
0      478 -rw-r--r-- 8          1  1001 1001 22-02-27 21:01 22-02-27 20:47 22-02-27 06:53 /home/OSLAB/P1/a
1      478 -rw-r--r-- 8          1  1001 1001 22-02-27 21:01 22-02-27 20:47 22-02-27 06:50 /home/OSLAB/P1/dir1/a
2      478 -rw-r--r-- 8          1  1001 1001 22-02-27 18:14 22-02-27 20:47 22-02-27 06:53 /home/OSLAB/P1/dir2/a
3      478 -rw-r--r-- 8          1  1001 1001 22-02-27 21:06 22-02-27 20:47 22-02-27 06:50 /home/OSLAB/P1/dir2/dir1/a
>> 2 s
Files a and dir2/a are identical

```

```

예. OPTION i 출력
20221234> find dir1 /
Index Size Mode          Blocks Links UID  GID Access          Change          Modify          Path
0      527 drwxrwxr-x 8          4  1001 1001 22-02-27 20:47 22-02-27 20:47 22-02-27 18:31 /home/OSLAB/P1/dir1
1      527 drwxrwxr-x 8          4  1001 1001 22-02-27 21:09 22-02-27 21:09 22-02-27 21:09 /home/OSLAB/P1/dir2/dir1
>> 1 i
Only in dir1: c
File dir1/d is a directory while file dir2/dir1/d is a regular file
Common subdirectories: dir1/e and dir2/dir1/e
Only in dir2/dir1: f

```

```

예. OPTION r 출력
20221234> find dir1 /
Index Size Mode          Blocks Links UID  GID Access          Change          Modify          Path
0      527 drwxrwxr-x 8          4  1001 1001 22-02-27 20:47 22-02-27 20:47 22-02-27 18:31 /home/OSLAB/P1/dir1
1      527 drwxrwxr-x 8          4  1001 1001 22-02-27 21:09 22-02-27 21:09 22-02-27 21:09 /home/OSLAB/P1/dir2/dir1
>> 1 r
diff -r dir1/b dir2/dir1/b
2,3c2,3
< banana
< carrot
\ No newline at end of file
---
> BANANA
> cherry
\ No newline at end of file
Only in dir1: c
File dir1/d is a directory while file dir2/dir1/d is a regular file
diff -r dir1/e/g dir2/dir1/e/g
1,2c1
< kiwi
< melon
---
> mango
Only in dir2/dir1: f

```

- 다) 내장명령어 2, exit
 - ✓ 프로그램을 종료시키는 명령어
 - ✓ EXIT 명령어 실행 시 프로그램 실행 시간(gettimeofday() 사용) 출력 후 종료
 - ✓ EXIT 명령어 실행 시 명령어를 실행시키는 프로세스, ssu_index 프로세스 모두가 종료되어야 함

```

예. exit 명령어 출력 예시
20221234> exit
Prompt End
Runtime: 210:479159(sec:usec)

```

- 라) 내장명령어 3, help
 - ✓ 명령어 사용법을 출력하는 명령어

```

예. help 명령어 출력 예시
20221234> a
Usage:
> find [FILENAME] [PATH]
>> [INDEX] [OPTION ... ]
> help
> exit

[OPTION ... ]
q : report only when files differ
s : report when two files are the same
i : ignore case differences in file contents
r : recursively compare any subdirectories found

```

○ 과제 구현에 필요한 함수 (필수 아님)

- 1. scandir : 디렉토리에 존재하는 파일 및 디렉토리 전체 목록 조회하는 라이브러리 함수

```
#include <dirent.h>
int scandir(const char *dirp, struct dirent ***namelist, int (*filter)(const struct dirent *), int (*compar)(const struct dirent **, const struct dirent **));
```

-1 : 오류가 발생, 상세한 오류 내용은 errno에 설정
0 이상 : 정상적으로 처리, namelist에 저장된 struct dirent *의 개수가 return

- 2. realpath : 상대경로를 절대경로로 변환하는 라이브러리 함수

```
#include <stdlib.h>
char *realpath(const char *path, char *resolved_path);
```

NULL : 오류가 발생, 상세한 오류 내용은 errno 전역변수에 설정
NULL이 아닌 경우 : resolved_path가 NULL이 아니면, resolved_path를 return,
resolved_path가 NULL이면, malloc(3)으로 할당하여 real path를 저장한 후에 return

- 3. gettimeofday : 마이크로초 단위의 시간 측정 함수

```
#include <sys/time.h>
#include <unistd.h>
int gettimeofday(struct timeval *tv, struct timezone *tz);
```

-1 : 오류가 발생, 상세한 오류 내용은 errno에 설정
0 : 정상적으로 처리

○ 보고서 제출 시 유의 사항

- 보고서 제출 마감은 3월 23일 11:59 PM
- 지연 제출 시 감점 : 1일 지연 시 마다 30% 감점, 3일 지연 후부터는 미제출 처리
- 압축 오류, 파일 누락 관련 감점 등은 강의계획서 참고

○ 프로그램 구현 시 유의 사항 및 감점

- ssu_index의 모든 출력 내용에 대해서 출력 형식 미 준수 시 감점(공백 또는 탭은 제외) 50% 감점

○ 구현 배점

- 가) 프롬프트 출력 [5점]

- 나) 내장명령어 1, find [FILENAME] [PATH], 아래 기타1과 기타2를 제외한 기본 기능 [15점]
- 다) 내장명령어 2, exit [5점]
- 라) 내장명령어 3, help [5점]
- 기타1) find 내장 명령어, >> [INDEX] [OPTION ...]에서 사용자가 선택한 [INDEX]에 해당하는 디렉토리 파일과 index가 0인 디렉토리 (즉, find 명령의 첫 번째 [FILENAME]에 해당하는 디렉토리 파일)의 내용을 비교하여 차이점 출력하는 기능 [48점]
- 기타2) find 내장 명령어, >> [INDEX] [OPTION ...]에서 4가지 옵션 처리 [20점]
- 기타3) gettimeofday() 로 시간 측정 [2점]
- 추가기능) 추가로 학생이 스스로 옵션을 추가할 경우 옵션당 3점, 최대 10점까지 부여 가능. 옵션을 추가할 경우 내장명령어 help에도 설명이 들어가야 하며, 추가한 옵션 기능을 상세하게 설명하고, 예시를 넣어 테스트한 내용을 보고서에 포함시켜야 함

○ 필수 구현 사항

- 가) 프롬프트 출력
- 나) 내장명령어 1, find [FILENAME] [PATH], 아래 기타 1을 제외한 기본 기능
- 다) 내장명령어 2, exit
- 라) 내장명령어 3, help
- 기타1) find 내장 명령어, >> [INDEX] [OPTION ...]에서 사용자가 선택한 [INDEX]에 해당하는 디렉토리 파일과 index가 0인 디렉토리 (즉, find 명령의 첫 번째 [FILENAME]에 해당하는 디렉토리 파일)의 내용을 비교하여 차이점 출력하는 기능
- 기타3) gettimeofday() 로 시간 측정