

# 설계과제 1 실행예제모음 : SSU-Backup

## 1. ssu\_backup

- ssu\_backup 프로그램은 명령어를 인자로 입력받아 해당 기능을 실행(백업디렉토리 접근을 위해 root 권한으로 실행해야 함)

예제 1-2. ssu\_backup 실행결과(help를 통해 명령어 목록을 출력한 모습)

```
% sudo ./ssu_backup help
Usage:
  > backup <PATH> [OPTION]...
    <none> : backup file if <PATH> is file
    -d : backup files in directory if <PATH> is directory
    -r : backup files in directory recursive if <PATH> is directory
    -y : backup file although already backedup
  > remove <PATH> [OPTION]...
    <none> : remove backedup file if <PATH> is file
    -d : remove backedup files in directory if <PATH> is directory
    -r : remove backedup files in directory recursive if <PATH> is directory
    -a : remove all backedup files
  > recover <PATH> [OPTION]...
    <none> : recover backedup file if <PATH> is file
    -d : recover backedup files in directory if <PATH> is directory
    -r : recover backedup files in directory recursive if <PATH> is directory
    -l : recover latest backedup file
    -n <NEW_PATH> : recover backedup file with new path
  > list [PATH]
    >> rm <INDEX> [OPTION]...
    >> rc <INDEX> [OPTION]...
    >> vi(m) <INDEX>
  > help [COMMAND]
```

- 첫 번째 인자를 입력하지 않을 경우(명령어를 입력하지 않을 경우) “ERROR: wrong input.” 출력 후 다음 줄에 “**실행한 프로그램 경로(argv[0]) help : show commands for program**” 출력 후 프로그램 종료

예제 1-1. ssu\_backup 실행결과(인자 입력하지 않을 시)

```
% sudo ./ssu_backup
ERROR: wrong input.
./ssu_backup help : show commands for program
```

- 첫 번째 인자로 잘못된 명령어를 입력할 경우 “ERROR: invalid command -- ‘**잘못된 명령어(argv[1])**’” 출력 후 다음 줄에 “**실행한 프로그램 경로(argv[0]) help : show commands for program**” 출력 후 프로그램 종료(0점)

예제 1-3. ssu\_backup 실행결과(잘못된 명령어 입력 시)

```
% sudo ./ssu_backup helpa
ERROR: invalid command -- 'helpa'
./ssu_backup help : show commands for program
```

## 2. 내장명령어 1. backup

- 첫 번째 인자 <PATH>를 백업 디렉토리에 백업함. 백업 진행 시 백업 시간(YYMMDDHHMMSS)을 이름으로 하는 버전 디렉토리를 백업 디렉토리 내에 만들고 버전 디렉토리 내에 백업한 파일, 디렉토리를 복사하여 붙여넣음. 명령어 실행 후 백업한 파일이 없다면 버전 디렉토리를 만들지 않음
- 첫 번째 인자 <PATH>의 최하위 디렉토리를 기준으로 버전 디렉토리와 연동하여 백업을 진행함 (ex. /home/oslab/a/b.txt를 백업할 시 새로운 버전 디렉토리(/home/backup/백업시간)은 /home/oslab/a/와 연동)
- 백업 성공 시 원본 경로와 백업 경로, 백업 시간을 (“원본 경로” backedup to “백업 경로”)의 형태로 출력하고 백업 로그(/home/backup/ssubak.log)에 (백업시간 : “원본 경로” backedup to “백업 경로”)의 형태로 append하여 저장
- 이미 내용이 같은 백업파일이 존재할 경우 (“원본 경로” already backedup to “백업 경로”)의 형태로 출력하고 따로 로그파일에는 해당 내용을 작성하지 않음. 내용이 같은 백업파일이 여러개일 경우 가장 최근에 백업한 파일의 경로를 출력

그림 2-1-1. /home/oslab/ 디렉토리 트리 구조	그림 2-2-1. 백업 디렉토리(/home/backup/) 트리 구조
<pre>/home/oslab/ ├ a/ │ └ a.txt │ └ b/ │   └ c.txt └ c.txt └ a.txt └ b/</pre>	<pre>/home/backup/ └ ssubak.log</pre>

예제 2-1. add 내장명령어 실행(그림 1의 상황에서 현재 작업 디렉토리(pwd)가 “/home/oslab/” 일 때)

```
% sudo ./ssu_backup backup ./a.txt
“/home/oslab/a.txt” backuped to “/home/backup/240225163125/a.txt”

% sudo ./ssu_backup backup ./a/b/c.txt
“/home/oslab/a/b/c.txt” backuped to “/home/backup/240225163137/c.txt”

//절대 경로와 상대경로 모두 입력 가능 ./a.txt == /home/oslab/a.txt
% sudo ./ssu_backup backup /home/oslab/a.txt
“/home/oslab/a.txt” already backuped to “/home/backup/240225163125/a.txt”

% echo “hello world” > ./a.txt

% sudo ./ssu_backup backup ./a.txt
‘/home/oslab/a.txt’ backuped to ‘/home/backup/240225163207/a.txt’

//경로 입력 없을 경우
% sudo ./ssu_backup backup
ERROR: missing operand <PATH>

//경로 없을 시 예외처리
% sudo ./ssu_backup backup ./a/b/a.txt
ERROR: ‘/home/oslab/a/b/a.txt’ is not exist

//경로가 디렉토리일 경우 예외처리
% sudo ./ssu_backup backup ./a/b
ERROR: ‘/home/oslab/a/b’ is directory.
- use ‘-d’ option or input file path.

//경로가 사용자 디렉토리를 벗어난 경우 예외처리
% sudo ./ssu_backup backup /root
ERROR: path must be in user directory
- ‘/root’ is not in user directory

//경로가 백업디렉토리일 경우 예외처리 == 위와 같음
% sudo ./ssu_backup backup /home/backup
ERROR: path must be in user directory
- ‘/home/backup’ is not in user directory
```

그림 2-3. 백업 디렉토리(/home/backup/) 트리 구조	그림 2-4. 백업 로그(ssubak.log) 파일 내용
<pre>/home/backup/ ├ 240225163125/ │ └ a.txt ├ 240225163137/ │ └ c.txt └ 240225163207/   └ a.txt   └ ssubak.log</pre>	<pre>240225163125 : “/home/oslab/a.txt” backuped to “/home/backup/240225163125/a.txt” 240225163137 : “/home/oslab/a/b/c.txt” backuped to “/home/backup/240225163137/c.txt” 240225163207 : “/home/oslab/a.txt” backuped to “/home/backup/240225163207/a.txt”</pre>

- ‘-d’ 옵션 입력시에는 <PATH>가 디렉토리일 시 해당 경로 아래 있는 모든 파일들에 대해 백업을 진행함.

그림 2-5. /home/oslab/ 디렉토리 트리 구조	그림 2-6. 백업 디렉토리(/home/backup/) 트리 구조
<pre>/home/oslab/ ├ a/ │ ├ a.txt │ ├ b/ │ │ └ c.txt │ └ c.txt └ a.txt └ b/</pre>	<pre>/home/backup/ └ ssubak.log</pre>

예. add 내장명령어 실행(그림 1의 상황에서 현재 작업 디렉토리(pwd)가 “/home/oslab/” 일 때)

```
% sudo ./ssu_backup backup ./a -d
"/home/oslab/a/a.txt" backedup to "/home/backup/240225163125/a.txt"
"/home/oslab/a/c.txt" backedup to "/home/backup/240225163125/c.txt"
```

//파일 경로 입력시 예외처리

그림 2-7. 백업 디렉토리(/home/backup/) 트리 구조	그림 2-8. 백업 로그(ssubak.log) 파일 내용
<pre>/home/backup/ └ 240225163125/   ├ a.txt   └ c.txt   └ ssubak.log</pre>	<pre>240225163125 : "/home/oslab/a/a.txt" backedup to "/home/backup/240225163125/a.txt" 240225163125 : "/home/oslab/a/c.txt" backedup to "/home/backup/240225163125/c.txt"</pre>

- ‘-r’ 옵션 입력시에는 <PATH>가 디렉토리일 시 해당 경로 아래 있는 모든 파일들에 대해 백업을 진행함. 이 때, 서브디렉토리 내에 모든 파일들에 대해서도 재귀적으로 탐색하여 백업을 진행함

그림 2-9. /home/oslab/ 디렉토리 트리 구조	그림 2-10. 백업 디렉토리(/home/backup/) 트리 구조
<pre>/home/oslab/ ├ a/ │ ├ a.txt │ ├ b/ │ │ └ c.txt │ └ c.txt └ a.txt └ b/</pre>	<pre>/home/backup/ └ ssubak.log</pre>

예. add 내장명령어 실행(그림 1의 상황에서 현재 작업 디렉토리(pwd)가 “/home/oslab/” 일 때)

```
% sudo ./ssu_backup backup ./a -r
"/home/oslab/a/a.txt" backedup to "/home/backup/240225163125/a.txt"
"/home/oslab/a/c.txt" backedup to "/home/backup/240225163125/c.txt"
"/home/oslab/a/b/c.txt" backedup to "/home/backup/240225163125/b/c.txt"
```

//파일 경로 입력시 예외처리

그림 2-11. 백업 디렉토리(/home/backup/) 트리 구조	그림 2-12. 백업 로그(ssubak.log) 파일 내용
<pre>/home/backup/ └ 240225163125/   ├ a.txt   ├ b/   │ └ c.txt   └ c.txt   └ ssubak.log</pre>	<pre>240225163125 : "/home/oslab/a/a.txt" backedup to "/home/backup/240225163125/a.txt" 240225163125 : "/home/oslab/a/c.txt" backedup to "/home/backup/240225163125/c.txt" 240225163125 : "/home/oslab/a/b/c.txt" backedup to "/home/backup/240225163125/b/c.txt"</pre>

- ‘-y’ 옵션 입력시에는 <PATH>에 대해 경로와 내용이 같은 백업파일이 존재해도 백업을 진행함

그림 2-13. /home/oslab/ 디렉토리 트리 구조	그림 2-14. 백업 디렉토리(/home/backup/) 트리 구조
<pre>/home/oslab/ ├ a/ │ ├ a.txt │ ├ b/ │ │ └ c.txt │ └ c.txt └ a.txt └ b/</pre>	<pre>/home/backup/ └ ssubak.log</pre>

예. add 내장명령어 실행(그림 1의 상황에서 현재 작업 디렉토리(pwd)가 "/home/oslab/" 일 때)

```
% sudo ./ssu_backup backup ./a.txt  
"/home/oslab/a.txt" backedup to "/home/backup/240225163125/a.txt"  
  
% sudo ./ssu_backup backup ./a.txt  
"/home/oslab/a.txt" already backedup to "/home/backup/240225163125/a.txt"  
  
% sudo ./ssu_backup backup ./a.txt -y  
"/home/oslab/a.txt" backedup to "/home/backup/240225163148/a.txt"
```

그림 2-15. 백업 디렉토리(/home/backup/) 트리 구조

```
/home/backup/  
└ 240225163125/  
  └ a.txt  
└ 240225163148/  
  └ a.txt  
└ ssubak.log
```

그림 2-16. 백업 로그(ssubak.log) 파일 내용

```
240225163125 : "/home/oslab/a/a.txt" backedup to "/home/backup/240225163125/a.txt"  
240225163148 : "/home/oslab/a/a.txt" backedup to "/home/backup/240225163148/a.txt"
```

### 3. 내장명령어 2. remove

- 첫 번째 인자 <PATH>에 대해 백업 디렉토리에 해당 경로에 대한 백업 파일을 삭제함. <PATH>에 대한 백업 파일이 여러 개일 경우 백업 파일에 대한 정보들을 번호를 매겨 출력. 목록 맨 위에는 (backup files of “원본 경로”)를 출력하고 다음 줄부터 번호를 매겨 백업 파일에 대한 백업 시간, 파일 사이즈를 출력함. 이 때 목록에 대한 출력 순서는 백업 시간에 대해 오름차순으로 정렬.
- 사용자 입력에 대해 번호를 입력하면 해당 번호에 맞는 백업 파일에 대해 삭제를 진행. 사용자 입력에 대해 0번 입력 시 백업 파일에 대한 삭제를 진행하지 않고 탈출
- 삭제 성공 시 (“백업 경로” removed by “원본 경로”)의 형태로 출력하고 백업 로그(/home/backup/ssubak.log)에 (삭제시간 : “백업 경로” removed by “원본 경로”)의 형태로 append하여 저장
- 명령어 실행 후 해당 버전 디렉토리 안에 아무 파일이 없다면 해당 버전 디렉토리 삭제

#### 예제 3-1-1. remove 내장명령어 실행(현재 작업 디렉토리(pwd)가 “/home/oslab/” 일 때)

```
% sudo ./ssu_backup remove ./a.txt
backup files of /home/oslab/a.txt
0. exit
1. 240225163125      14bytes
2. 240225163207      11bytes
)) 0

% sudo ./ssu_backup remove ./a.txt
backup files of /home/oslab/a.txt
0. exit
1. 240225163125      14bytes
2. 240225163207      11bytes
)) 1
"/home/backup/240225163125/a.txt" removed by "/home/oslab/a.txt"

% sudo ./ssu_backup remove ./a.txt
"/home/backup/240225163207/a.txt" removed by "/home/oslab/a.txt"
```

//사용자 흄 디렉토리에 있지만 버전 디렉토리에 있는 파일일 경우(지워진 파일일 경우)

```
% sudo ./ssu_backup remove ./a/c.txt
"/home/backup/240225163137/c.txt" removed by "/home/oslab/a/c.txt"
```

#### 예제 3-1-2. 백업 로그(ssubak.log) 파일 내용

```
...
240225163125 : "/home/oslab/a.txt" backuped to "/home/backup/240225163125/a.txt"
240225163137 : "/home/oslab/a/c.txt" backuped to "/home/backup/240225163137/c.txt"
240225163207 : "/home/oslab/a.txt" backuped to "/home/backup/240225163207/a.txt"
...
240225163302 : "/home/backup/240225163125/a.txt" removed by "/home/oslab/a.txt"
240225163314 : "/home/backup/240225163207/a.txt" removed by "/home/oslab/a.txt"
240225163343 : "/home/backup/240225163137/c.txt" removed by "/home/oslab/a/c.txt"
```

그림 3-1-1. /home/oslab/ 디렉토리 트리 구조	그림 3-1-2. (전)백업 디렉토리(/home/backup/) 트리 구조
/home/oslab/ └ a/     └ a.txt     └ b/       └ c.txt └ a.txt └ b/	/home/backup/ └ 240225163125/     └ a.txt     └ 240225163137/       └ c.txt     └ 240225163207/       └ a.txt └ ssubak.log
그림 3-1-3. (후)백업 디렉토리(/home/backup/) 트리 구조	/home/backup/ └ ssubak.log

- ‘-d’ 옵션 입력 시에는 <PATH>가 디렉토리일 시 해당 경로 아래 있는 모든 파일들의 백업 파일에 대해 삭제를 진행함.
- 백업 파일이 2개 이상인 파일들에 대해 모두 목록을 출력하여 사용자 입력을 기다림

예제 3-2-1. remove 내장명령어 실행(현재 작업 디렉토리(pwd)가 “/home/oslab/” 일 때)

```
% sudo ./ssu_backup remove /home/oslab/a -d
“/home/backup/240225163125/a.txt” removed by “/home/oslab/a/a.txt”
backup files of /home/oslab/a/c.txt
0. exit
1. 240225163125      14bytes
2. 240225163207      11bytes
>> 2
“/home/backup/240225163207/c.txt” removed by “/home/oslab/a/c.txt”
```

예제 3-2-2. 백업 로그(ssubak.log) 파일 내용

```
...
240225163125 : “/home/oslab/a/a.txt” backed up to “/home/backup/240225163125/a.txt”
240225163125 : “/home/oslab/a/c.txt” backed up to “/home/backup/240225163125/c.txt”
240225163125 : “/home/oslab/a/b/c.txt” backed up to “/home/backup/240225163125/b/c.txt”
240225163207 : “/home/oslab/a/c.txt” backed up to “/home/backup/240225163207/c.txt”
...
240225163302 : “/home/backup/240225163125/a.txt” removed by “/home/oslab/a/a.txt”
240225163302 : “/home/backup/240225163207/c.txt” removed by “/home/oslab/a/c.txt”
```

그림 3-2-1. /home/oslab/ 디렉토리 트리 구조

```
/home/oslab/
├ a/
│ └ a.txt
│   └ b/
│     └ c.txt
└ a.txt
  └ b/
```

그림 3-2-2. (전)백업 디렉토리(/home/backup/) 트리 구조

```
/home/backup/
├ 240225163125/
│ └ a.txt
│   └ b/
│     └ c.txt
└ 240225163207/
  └ c.txt
  └ ssubak.log
```

그림 3-2-3. (후)백업 디렉토리(/home/backup/) 트리 구조

```
/home/backup/
├ 240225163125/
│ └ b/
│   └ c.txt
└ c.txt
└ ssubak.log
```

- ‘-r’ 옵션 입력 시에는 <PATH>가 디렉토리일 시 해당 경로 아래 있는 모든 파일들의 백업 파일에 대해 삭제를 진행함. 이 때, 서브 디렉토리 내에 모든 파일들에 대해서도 재귀적으로 탐색하여 백업을 진행함. 진행 순서는 파일을 우선으로 하고 그 다음 서브디렉토리를 재귀적으로 탐색함(BFS)
- 백업 파일이 2개 이상인 파일들에 대해 모두 목록을 출력하여 사용자 입력을 기다림
- ‘-r’ 옵션과 ‘-d’ 옵션을 동시에 입력 시에는 ‘-r’ 옵션을 적용시켜 재귀적으로 백업 파일에 대한 삭제를 진행함

**예제 3-3-1. remove 내장명령어 실행(현재 작업 디렉토리(pwd)가 “/home/oslab/” 일 때)**

```
% sudo ./ssu_backup remove /home/oslab/a -r
“/home/backup/240225163125/a.txt” removed by “/home/oslab/a/a.txt”
backup files of /home/oslab/a/c.txt
0. exit
1. 240225163125      14bytes
2. 240225163207      11bytes
>> 2
“/home/backup/240225163207/c.txt” removed by “/home/oslab/a/c.txt”
“/home/backup/240225163125/b/c.txt” removed by “/home/oslab/a/b/c.txt”
```

**예제 3-3-2. 백업 로그(ssubak.log) 파일 내용**

```
...
240225163125 : “/home/oslab/a/a.txt” backed up to “/home/backup/240225163125/a.txt”
240225163125 : “/home/oslab/a/c.txt” backed up to “/home/backup/240225163125/c.txt”
240225163137 : “/home/oslab/a/b/c.txt” backed up to “/home/backup/240225163125/b/c.txt”
240225163207 : “/home/oslab/a/c.txt” backed up to “/home/backup/240225163207/c.txt”
...
240225163302 : “/home/backup/240225163125/a.txt” removed by “/home/oslab/a/a.txt”
240225163302 : “/home/backup/240225163207/c.txt” removed by “/home/oslab/a/c.txt”
240225163302 : “/home/backup/240225163125/b/c.txt” removed by “/home/oslab/a/b/c.txt”
```

**그림 3-3-1. /home/oslab/ 디렉토리 트리 구조**

```
/home/oslab/
├ a/
│ └ a.txt
│ └ b/
│   └ c.txt
└ a.txt
└ b/
```

**그림 3-3-2. (전)백업 디렉토리(/home/backup/) 트리 구조**

```
/home/backup/
└ 240225163125/
  └ a.txt
  └ b/
    └ c.txt
  └ c.txt
  └ 240225163207/
    └ c.txt
    └ ssubak.log
```

**그림 3-2-3. (후)백업 디렉토리(/home/backup/) 트리 구조**

```
/home/backup/
└ 240225163125/
  └ c.txt
  └ ssubak.log
```

- ‘-a’ 옵션 입력 시에는 백업본이 2개 이상인 파일들에 대해 목록을 출력하지 않고 모든 백업 파일을 삭제함.

**예제 3-4-1. remove 내장명령어 실행(현재 작업 디렉토리(pwd)가 “/home/oslab/” 일 때)**

```
% sudo ./ssu_backup remove /home/oslab/a -r -a
“/home/backup/240225163125/a.txt” removed by “/home/oslab/a/a.txt”
“/home/backup/240225163125/c.txt” removed by “/home/oslab/a/c.txt”
“/home/backup/240225163207/c.txt” removed by “/home/oslab/a/c.txt”
“/home/backup/240225163125/b/c.txt” removed by “/home/oslab/a/b/c.txt”
```

**예제 3-4-2. 백업 로그(ssubak.log) 파일 내용**

```
...
240225163125 : “/home/oslab/a/a.txt” backed up to “/home/backup/240225163125/a.txt”
240225163125 : “/home/oslab/a/c.txt” backed up to “/home/backup/240225163125/c.txt”
240225163125 : “/home/oslab/a/b/c.txt” backed up to “/home/backup/240225163125/b/c.txt”
240225163207 : “/home/oslab/a/c.txt” backed up to “/home/backup/240225163207/c.txt”
...
240225163302 : “/home/backup/240225163125/a.txt” removed by “/home/oslab/a/a.txt”
240225163302 : “/home/backup/240225163125/c.txt” removed by “/home/oslab/a/c.txt”
240225163302 : “/home/backup/240225163207/c.txt” removed by “/home/oslab/a/c.txt”
240225163302 : “/home/backup/240225163125/b/c.txt” removed by “/home/oslab/a/b/c.txt”
```

그림 3-4-1. /home/oslab/ 디렉토리 트리 구조	그림 3-4-2. (전)백업 디렉토리(/home/backup/) 트리 구조
/home/oslab/ └ a/     └ a.txt     └ b/       └ c.txt └ a.txt └ b/	/home/backup/ └ 240225163125/     └ a.txt     └ b/       └ c.txt       └ c.txt └ 240225163207/     └ c.txt └ ssubak.log
그림 3-2-3. (후)백업 디렉토리(/home/backup/) 트리 구조	
	/home/backup/ └ ssubak.log

#### 4. 내장명령어 3. recover

- 첫 번째 인자 <PATH>에 대해 백업 디렉토리에 해당 경로에 대한 백업 파일을 원본 경로에 복구하고(원본 경로에 파일이 있을 경우 덮어씌움) 백업 파일을 삭제함. <PATH>에 대한 백업 파일이 여러개일 경우 백업 파일에 대한 정보들을 번호를 매겨 출력. 목록 맨 위에는 (backup files of “원본 경로”)를 출력하고 다음 줄부터 번호를 매겨 백업 파일에 대한 백업 시간, 파일 사이즈를 출력함. 이 때 목록에 대한 출력 순서는 백업 시간에 대해 오름차순으로 정렬
- 사용자 입력에 대해 번호를 입력하면 해당 번호에 맞는 백업 파일에 대해 복구를 진행. 사용자 입력에 대해 0번 입력 시 백업 파일에 대한 복구를 진행하지 않고 탈출
- 복구 성공 시 (“백업 경로” recovered to “원본 경로”)의 형태로 출력하고 백업 로그 (/home/backup/ssubak.log)에 (복구시간 : “백업 경로” recovered to “원본 경로”)의 형태로 append하여 저장
- 명령어 실행 후 해당 버전 디렉토리 안에 아무 파일이 없다면 해당 버전 디렉토리 삭제

예제 4-1-1. recover 내장명령어 실행(현재 작업 디렉토리(pwd)가 “/home/oslab/” 일 때)

```
% sudo ./ssu_backup backup ./a.txt
“/home/oslab/a.txt” backedup to “/home/backup/240225163207/a.txt”

% sudo ./ssu_backup recover ./a.txt
backup files of /home/oslab/a.txt
0. exit
1. 240225163125      14bytes
2. 240225163207      11bytes
>> 0

% sudo ./ssu_backup recover ./a.txt
backup files of /home/oslab/a.txt
0. exit
1. 240225163125      14bytes
2. 240225163207      11bytes
>> 2
“/home/backup/240225163207/a.txt” not changed with “/home/oslab/a.txt”

% sudo ./ssu_backup recover ./a.txt
backup files of /home/oslab/a.txt
0. exit
1. 240225163125      14bytes
2. 240225163207      11bytes
>> 1
“/home/backup/240225163125/a.txt” recovered to “/home/oslab/a.txt”

% sudo ./ssu_backup recover ./a.txt
“/home/backup/240225163207/a.txt” recovered to “/home/oslab/a.txt”
```

//사용자 홈 디렉토리에 없지만 버전 디렉토리에 있는 파일일 경우(지워진 파일일 경우)

```
% sudo ./ssu_backup recover ./a/c.txt
“/home/backup/240225163137/c.txt” recovered to “/home/oslab/a/c.txt”
```

예제 4-1-2. 백업 로그(ssubak.log) 파일 내용

```
...
240225163125 : “/home/oslab/a.txt” backedup to “/home/backup/240225163125/a.txt”
240225163137 : “/home/oslab/a/c.txt” backedup to “/home/backup/240225163137/c.txt”
...
240225163207 : “/home/oslab/a.txt” backedup to “/home/backup/240225163207/a.txt”
240225163302 : “/home/backup/240225163207/a.txt” recovered to “/home/oslab/a.txt”
240225163314 : “/home/backup/240225163125/a.txt” recovered to “/home/oslab/a.txt”
240225163343 : “/home/backup/240225163137/c.txt” recovered to “/home/oslab/a/c.txt”
```

그림 4-1-1. (전)/home/oslab/ 디렉토리 트리 구조

```
/home/oslab/
├ a/
│ └ a.txt
│ └ b/
│   └ c.txt
└ a.txt
└ b/
```

그림 4-1-3. (후)/home/oslab/ 디렉토리 트리 구조

```
/home/oslab/
├ a/
│ └ a.txt
│ └ b/
│   └ c.txt
└ a.txt
└ b/
```

그림 4-1-2. (전)백업 디렉토리(/home/backup/) 트리 구조

```
/home/backup/
└ 240225163125/
  └ a.txt
└ 240225163137/
  └ c.txt
└ 240225163207/
  └ a.txt
└ ssubak.log
```

그림 4-1-4. (후)백업 디렉토리(/home/backup/) 트리 구조

```
/home/backup/
└ ssubak.log
```

- ‘-d’ 옵션 입력 시에는 <PATH>가 디렉토리일 시 해당 경로 아래 있는 모든 파일들의 백업 파일에 대해 복구를 진행함.
- 백업 파일이 2개 이상인 파일들에 대해 모두 목록을 출력하여 사용자 입력을 기다림

예제 4-2-1. recover 내장명령어 실행(현재 작업 디렉토리(pwd)가 “/home/oslab/” 일 때)

```
% sudo ./ssu_backup recover /home/oslab/a -d
"/home/backup/240225163125/a.txt" recovered to "/home/oslab/a/a.txt"
backup files of /home/oslab/a/c.txt
0. exit
1. 240225163125      14bytes
2. 240225163207      11bytes
)) 2
"/home/backup/240225163207/c.txt" recovered to "/home/oslab/a/c.txt"
```

예제 4-2-2. 백업 로그(ssubak.log) 파일 내용

```
...
240225163125 : "/home/oslab/a/a.txt" backed up to "/home/backup/240225163125/a.txt"
240225163125 : "/home/oslab/a/c.txt" backed up to "/home/backup/240225163125/c.txt"
240225163125 : "/home/oslab/a/b/c.txt" backed up to "/home/backup/240225163125/b/c.txt"
240225163207 : "/home/oslab/a/c.txt" backed up to "/home/backup/240225163207/c.txt"
...
240225163302 : "/home/backup/240225163125/a.txt" recovered to "/home/oslab/a/a.txt"
240225163302 : "/home/backup/240225163207/c.txt" recovered to "/home/oslab/a/c.txt"
```

그림 4-2-1. (전)/home/oslab/ 디렉토리 트리 구조

```
/home/oslab/
├ a/
│ ├ a.txt
│ └ b/
│   └ c.txt
└ a.txt
└ b/
```

그림 4-2-3. (후)/home/oslab/ 디렉토리 트리 구조

```
/home/oslab/
├ a/
│ ├ a.txt
│ ├ b/
│ │ └ c.txt
│ └ c.txt
└ a.txt
└ b/
```

그림 4-2-2. (전)백업 디렉토리(/home/backup/) 트리 구조

```
/home/backup/
└ 240225163125/
  ├ a.txt
  ├ b/
  │ └ c.txt
  └ c.txt
└ 240225163207/
  └ c.txt
└ ssubak.log
```

그림 4-2-4. (후)백업 디렉토리(/home/backup/) 트리 구조

```
/home/backup/
└ 240225163125/
  ├ b/
  │ └ c.txt
  └ c.txt
└ ssubak.log
```

- ‘-r’ 옵션 입력 시에는 <PATH>가 디렉토리일 시 해당 경로 아래 있는 모든 파일들의 백업 파일에 대해 복구를 진행함. 이 때, 서브디렉토리 내에 모든 파일들에 대해서도 재귀적으로 탐색하여 백업을 진행함. 진행 순서는 파일을 우선으로 하고 그 다음 서브디렉토리를 재귀적으로 탐색함(BFS)
- 백업 파일이 2개 이상인 파일들에 대해 모두 목록을 출력하여 사용자 입력을 기다림
- ‘-r’ 옵션과 ‘-d’ 옵션을 동시에 입력 시에는 ‘-r’ 옵션을 적용시켜 재귀적으로 백업 파일에 대한 복구를 진행함

예제 4-3-1. recover 내장명령어 실행(현재 작업 디렉토리(pwd)가 “/home/oslab/” 일 때)

```
% sudo ./ssu_backup recover /home/oslab/a -r
“/home/backup/240225163125/a.txt” recovered to “/home/oslab/a/a.txt”
backup files of /home/oslab/a/c.txt
0. exit
1. 240225163125      14bytes
2. 240225163207      11bytes
>> 2
“/home/backup/240225163207/c.txt” recovered to “/home/oslab/a/c.txt”
“/home/backup/240225163125/b/c.txt” recovered to “/home/oslab/a/b/c.txt”
```

예제 4-3-2. 백업 로그(ssubak.log) 파일 내용

```
...
240225163125 : “/home/oslab/a/a.txt” backed up to “/home/backup/240225163125/a.txt”
240225163125 : “/home/oslab/a/c.txt” backed up to “/home/backup/240225163125/c.txt”
240225163125 : “/home/oslab/a/b/c.txt” backed up to “/home/backup/240225163125/b/c.txt”
240225163207 : “/home/oslab/a/c.txt” backed up to “/home/backup/240225163207/c.txt”
...
240225163302 : “/home/backup/240225163125/a.txt” recovered to “/home/oslab/a/a.txt”
240225163302 : “/home/backup/240225163207/c.txt” recovered to “/home/oslab/a/c.txt”
240225163302 : “/home/backup/240225163125/b/c.txt” recovered to “/home/oslab/a/b/c.txt”
```

그림 4-3-1. (전)/home/oslab/ 디렉토리 트리 구조

```
/home/oslab/
├ a/
│ ├ a.txt
│ └ b/
│   └ c.txt
└ a.txt
└ b/
```

그림 4-3-3. (후)/home/oslab/ 디렉토리 트리 구조

```
/home/oslab/
├ a/
│ ├ a.txt
│ ├ b/
│   └ c.txt
│ └ c.txt
└ a.txt
└ b/
```

그림 4-3-2. (전)백업 디렉토리(/home/backup/) 트리 구조

```
/home/backup/
├ 240225163125/
│ ├ a.txt
│ ├ b/
│   └ c.txt
│ └ c.txt
└ 240225163207/
  └ c.txt
└ ssubak.log
```

그림 4-3-4. (후)백업 디렉토리(/home/backup/) 트리 구조

```
/home/backup/
├ 240225163125/
  └ c.txt
└ ssubak.log
```

- ‘-l’ 옵션 입력 시에는 백업 파일이 2개 이상인 파일들에 대해 리스트를 출력하지 않고 가장 최근에 백업한 백업 파일을 복구함.

예제 4-4-1. recover 내장명령어 실행(현재 작업 디렉토리(pwd)가 “/home/oslab/” 일 때)

```
% sudo ./ssu_backup recover ./a.txt
backup files of /home/oslab/a.txt
0. exit
1. 240225163125      14bytes
2. 240225163207      11bytes
>> 0
```

```
% sudo ./ssu_backup recover ./a.txt -l
“/home/backup/240225163207/a.txt” recovered to “/home/oslab/a.txt”
```

예제 4-4-2. 백업 로그(ssubak.log) 파일 내용

```
...
240225163125 : “/home/oslab/a.txt” backed up to “/home/backup/240225163125/a.txt”
240225163137 : “/home/oslab/a/c.txt” backed up to “/home/backup/240225163137/c.txt”
240225163207 : “/home/oslab/a.txt” backed up to “/home/backup/240225163207/a.txt”
...
240225163314 : “/home/backup/240225163125/a.txt” recovered to “/home/oslab/a.txt”
```

그림 4-4-1. (전) /home/oslab/ 디렉토리 트리 구조	그림 4-4-2. (전) 백업 디렉토리 (/home/backup/) 트리 구조
/home/oslab/ └ a/     └ a.txt     └ b/       └ c.txt └ a.txt └ b/	/home/backup/ └ 240225163125/     └ a.txt     └ 240225163137/       └ c.txt └ 240225163207/ └ a.txt └ ssubak.log
그림 4-4-3. (후) /home/oslab/ 디렉토리 트리 구조	그림 4-4-4. (후) 백업 디렉토리 (/home/backup/) 트리 구조
/home/oslab/ └ a/     └ a.txt     └ b/       └ c.txt └ a.txt └ b/	/home/backup/ └ 240225163125/     └ a.txt     └ 240225163137/       └ c.txt └ ssubak.log

- ‘-n’ 옵션 입력 시에는 해당 옵션 뒤에 <NEW\_PATH>을 필수로 입력받아, 백업 파일에 대해 새로운 경로를 지정하여 복구를 진행함. 만약 해당 경로가 없을 경우 디렉토리를 생성하여 경로를 설정함

예제 4-5-1. recover 내장명령어 실행(현재 작업 디렉토리(pwd)가 “/home/oslab/” 일 때)
% sudo ./ssu_backup recover /home/oslab/a -r -n ./new_d/1 “/home/backup/240225163125/a.txt” recovered to “/home/oslab/new_d/1/a.txt” backup files of /home/oslab/a/c.txt 0. exit 1. 240225163125        14bytes 2. 240225163207        11bytes )) 2 “/home/backup/240225163207/c.txt” recovered to “/home/oslab/new_d/1/c.txt” “/home/backup/240225163125/b/c.txt” recovered to “/home/oslab/new_d/1/b/c.txt”
예제 4-5-2. 백업 로그(ssubak.log) 파일 내용
... 240225163125 : “/home/oslab/a/a.txt” backed up to “/home/backup/240225163125/a.txt” 240225163125 : “/home/oslab/a/c.txt” backed up to “/home/backup/240225163125/c.txt” 240225163125 : “/home/oslab/a/b/c.txt” backed up to “/home/backup/240225163125/b/c.txt” 240225163207 : “/home/oslab/a/c.txt” backed up to “/home/backup/240225163207/c.txt” ... 240225163302 : “/home/backup/240225163125/a.txt” recovered to “/home/oslab/new_d/1/a.txt” 240225163302 : “/home/backup/240225163207/c.txt” recovered to “/home/oslab/new_d/1/c.txt” 240225163302 : “/home/backup/240225163125/b/c.txt” recovered to “/home/oslab/new_d/1/b/c.txt”

그림 4-5-1. (전) /home/oslab/ 디렉토리 트리 구조	그림 4-5-2. (전) 백업 디렉토리 (/home/backup/) 트리 구조
/home/oslab/ └ a/     └ a.txt     └ b/       └ c.txt └ a.txt └ b/	/home/backup/ └ 240225163125/     └ a.txt     └ b/       └ c.txt └ c.txt └ 240225163207/     └ c.txt └ ssubak.log
그림 4-5-3. (후) /home/oslab/ 디렉토리 트리 구조	그림 4-5-4. (후) 백업 디렉토리 (/home/backup/) 트리 구조
/home/oslab/ └ a/     └ a.txt     └ b/       └ c.txt └ a.txt └ b/ └ new_d/ └ 1/ └ a.txt └ b/ └ c.txt └ c.txt	/home/backup/ └ 240225163125/     └ c.txt └ ssubak.log

## 5. 내장명령어 4. list

- 첫 번째 인자 <PATH>에 대해 백업 파일들의 원본 경로를 기준으로 트리 형태로 출력함. <PATH> 생략 시 백업 디렉토리 내 모든 백업 파일들을 기준으로 함.
- 백업 파일들의 트리를 출력할 때는 백업 파일들의 원본 경로들을 디렉토리 구조의 형태로 출력하며 위에서부터 번호를 매기며 한줄에는 원본 파일 및 디렉토리 이름, 파일일 경우 최근 백업 시간, 파일 사이즈를 출력함. 이때 트리에 대한 출력 순서는 원본 파일 및 디렉토리 이름에 대해 오름차순으로 정렬함
- 목록을 출력한 후에는 사용자 입력을 기다리며 입력에 따라 명령을 실행함
- rm <INDEX> [OPTION]... 입력 시 해당 <INDEX>에 대한 경로를 remove 명령어와 같이 처리함
- rc <INDEX> [OPTION]... 입력 시 해당 <INDEX>에 대한 경로를 recover 명령어와 같이 처리함
- vi(m) <INDEX> 입력 시 해당 <INDEX>에 대한 원본 파일에 대해 수정함
- exit 입력 시 프로그램 종료

예제 5-1. list 내장명령어 실행(현재 작업 디렉토리(pwd)가 "/home/oslab/" 일 때)

```
% sudo ./ssu_backup list a.txt
0. a.txt          240225163148 1,025bytes
)) exit

% sudo ./ssu_backup list a.txt
0. a.txt          240225163148 1,025bytes
)) vim 0
...
% sudo ./ssu_backup list
0. /home/oslab
1.   ↗ a/
2.     ↗ a.txt      240225163125 32bytes
3.     ↗ b/
4.       ↗ c.txt    240225163125 14bytes
5.     ↗ c.txt    240225163207 11bytes
6.   ↗ a.txt      240225163148 1,025bytes
)) rm 5 -a
"/home/backup/240225163125/c.txt" removed by "/home/oslab/a/c.txt"
"/home/backup/240225163207/c.txt" removed by "/home/oslab/a/c.txt"

% sudo ./ssu_backup list ./a
0. a/
1.   ↗ a.txt      240225163125 32bytes
2.   ↗ b/
3.     ↗ c.txt    240225163125 14bytes
)) rc 0 -r -n new_d
"/home/backup/240225163125/a.txt" recovered to "/home/oslab/new_d/a.txt"
"/home/backup/240225163125/b/c.txt" recovered to "/home/oslab/new_d/b/c.txt"

% sudo ./ssu_backup list
no backup file(s) in backup directory
```

예제 5-2. 백업 로그(ssubak.log) 파일 내용

```
...
240225163125 : "/home/oslab/a/a.txt" backed up to "/home/backup/240225163125/a.txt"
240225163125 : "/home/oslab/a/c.txt" backed up to "/home/backup/240225163125/c.txt"
240225163125 : "/home/oslab/a/b/c.txt" backed up to "/home/backup/240225163125/b/c.txt"
240225163137 : "/home/oslab/a.txt" backed up to "/home/backup/240225163137/a.txt"
240225163148 : "/home/oslab/a.txt" backed up to "/home/backup/240225163148/a.txt"
240225163207 : "/home/oslab/a/c.txt" backed up to "/home/backup/240225163207/c.txt"
...
240225163302 : "/home/backup/240225163125/c.txt" removed by "/home/oslab/a/c.txt"
240225163302 : "/home/backup/240225163207/c.txt" removed by "/home/oslab/a/c.txt"
240225163322 : "/home/backup/240225163125/a.txt" recovered to "/home/oslab/new_d/a.txt"
240225163322 : "/home/backup/240225163125/b/c.txt" recovered to "/home/oslab/new_d/b/c.txt"
```

그림 5-1. (전)/home/oslab/ 디렉토리 트리 구조	그림 5-2. (전)백업 디렉토리(/home/backup/) 트리 구조
/home/oslab/ └ a/     └ a.txt     └ b/       └ c.txt └ b/	/home/backup/ └ 240225163125/     └ a.txt     └ b/       └ c.txt       └ c.txt └ 240225163137/       └ a.txt └ 240225163148/       └ a.txt └ 240225163207/       └ c.txt └ ssubak.log
그림 5-3. (후)/home/oslab/ 디렉토리 트리 구조	그림 5-4. (후)백업 디렉토리(/home/backup/) 트리 구조
/home/oslab/ └ a/     └ a.txt     └ b/       └ c.txt └ a.txt └ b/ └ new_d/     └ a.txt     └ b/ └ c.txt	/home/backup/ └ 240225163137/       └ a.txt └ 240225163148/       └ a.txt └ ssubak.log

## 6. 내장명령어 5. help

### 1) Usage : help [COMMAND]

- 프로그램 내장명령어에 대한 설명(Usage) 출력

### 2) 실행결과

- 첫 번째 인자에 내장 명령어 입력시 해당 명령어에 대한 설명(Usage)을 출력. 첫 번째 인자는 생략 가능하며 생략 시 프로그램 내 모든 내장명령어에 대한 설명(Usage)을 출력

예. help 내장명령어 실행

```
% sudo ./ssu_backup help
Usage:
  > backup <PATH> [OPTION]... : backup file if <PATH> is file
    -d : backup files in directory if <PATH> is directory
    -r : backup files in directory recursive if <PATH> is directory
    -y : backup file although already backedup
  > remove <PATH> [OPTION]... : remove backedup file if <PATH> is file
    -d : remove backedup files in directory if <PATH> is directory
    -r : remove backedup files in directory recursive if <PATH> is directory
    -a : remove all backedup files
  > recover <PATH> [OPTION]... : recover backedup file if <PATH> is file
    -d : recover backedup files in directory if <PATH> is directory
    -r : recover backedup files in directory recursive if <PATH> is directory
    -l : recover latest backedup file
    -n <NEW_PATH> : recover backedup file with new path
  > list [<PATH>] : show backup list by directory structure
    >> rm <INDEX> [OPTION]... : remove backedup files of [INDEX] with [OPTION]
    >> rc <INDEX> [OPTION]... : recover backedup files of [INDEX] with [OPTION]
    >> vi(m) <INDEX> : edit original file of [INDEX]
    >> exit : exit program
  > help [COMMAND] : show commands for program

% sudo ./ssu_backup help recover
Usage: recover <PATH> [OPTION]... : recover backedup file if <PATH> is file
  -d : recover backedup files in directory if <PATH> is directory
  -r : recover backedup files in directory recursive if <PATH> is directory
  -l : recover latest backedup file
  -n <NEW_PATH> : recover backedup file with new path
```

오류에 대한 예외처리에 대해서는 자유롭게 구현 가능