

# Practice 1

컴퓨터프로그래밍1

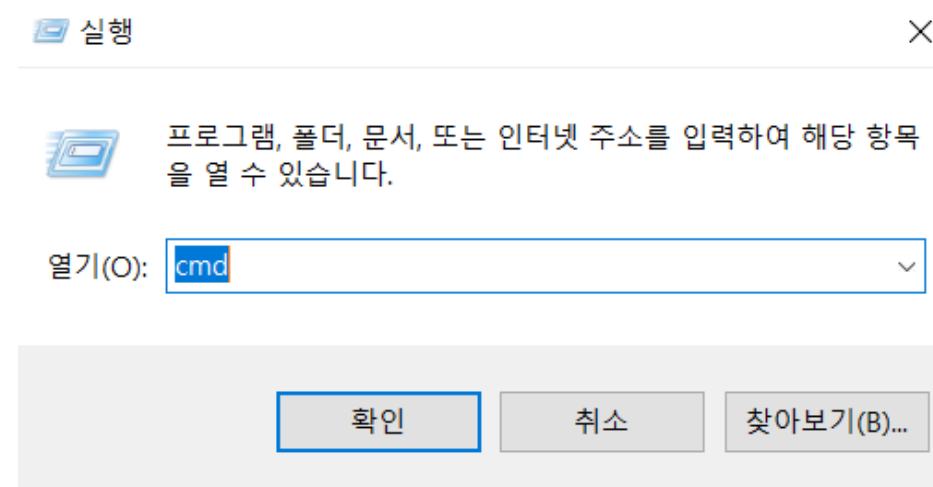
3월 16일

# Linux & VIM

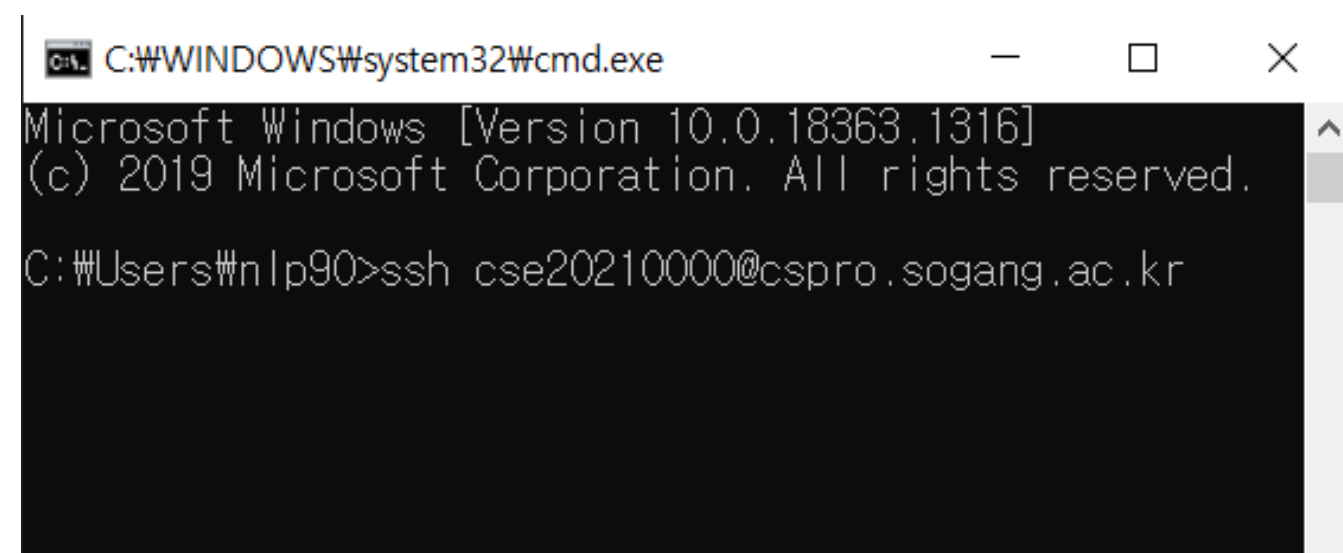
# Linux OS 환경 접속



1. 윈도우 실행 창 실행 [윈도우키] + [r]
2. cmd 검색하여 실행



3. ssh [cse학번]@cspro.sogang.ac.kr 입력



4. 비밀번호 입력 (초기 비밀번호는 cse학번)



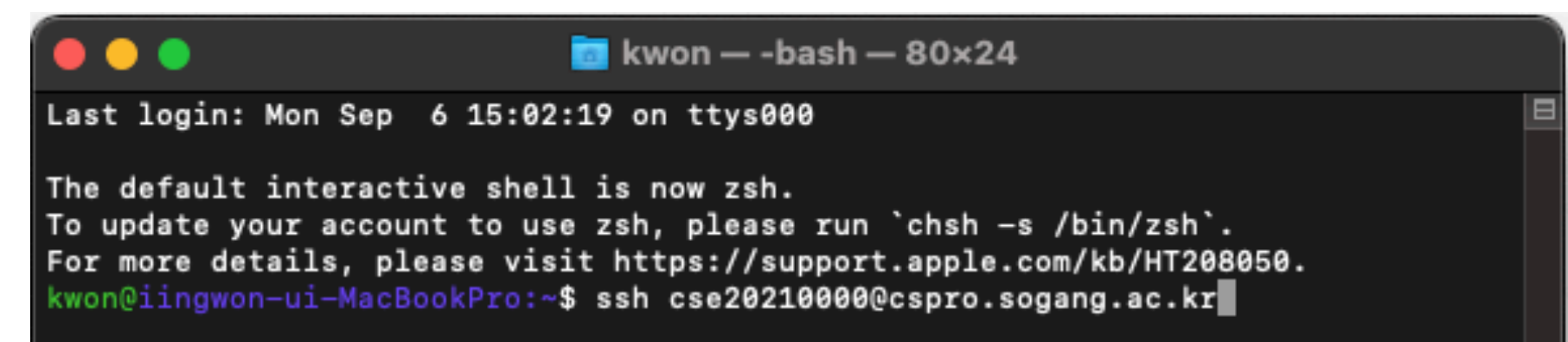
1. spotlight 실행 [cmd] + [space]



2. terminal 검색하여 실행



3. ssh [cse학번]@cspro.sogang.ac.kr 입력



4. 비밀번호 입력 (초기 비밀번호는 cse학번)

# Linux OS 환경 접속

- 로그인 성공

```
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-193-generic x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/advantage

246 packages can be updated.
114 updates are security updates.

New release '18.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Sun Sep  5 23:21:45 2021 from 163.239.199.207
gr120200190@cspro:~$
```

- 로그인 실패

```
[kwon@iingwon-ui-MacBookPro:~$ ssh gr120200190@cspro.sogang.ac.kr
[gr120200190@cspro.sogang.ac.kr's password:
Permission denied, please try again.
[gr120200190@cspro.sogang.ac.kr's password:
Permission denied, please try again.
[gr120200190@cspro.sogang.ac.kr's password:
gr120200190@cspro.sogang.ac.kr: Permission denied (publickey,password).
kwon@iingwon-ui-MacBookPro:~$
```

# Linux 명령어 실습

- passwd
  - 자신의 비밀번호를 변경 시켜주는 명령어

passwd

```
gr120080186@cspiro ~ $ passwd
Changing password for gr120080186
(current) UNIX password:
New UNIX password:
Retype new UNIX password:
passwd: password updated successfully
gr120080186@cspiro ~ $
```

현재 패스워드를 입력(1번)

새로운 패스워드를 입력(2번)

- who
  - 현재 접속한 사용자의 정보 확인

who

```
gr120080186@cspiro1 ~ $ who
gr120080 pts/1          Jan  2 12:21 (mclab21.sogang.ac.kr)
gr120080186@cspiro1 ~ $
```

# Linux 명령어 실습

- cat
  - 파일의 내용을 보는 명령어 (catenate의 약자)

`cat [filename]` 명시한 file의 내용을 출력한다.

```
gr120080186@cspro1 ~ $ cat
```

```
Assembly  
Assembly  
Operating System  
Operating System
```

cat 명령어만 수행하면, 자신이 입력한 내용을 바로 연달아 화면에 출력해준다.

```
gr120080186@cspro1 ~ $ cat > catExample
```

cat 다음에 ">"를 사용하면 입력한 내용을 ">" 다음에 명시한 catExample에 저장한다.

```
Database
```

```
Data Structures
```

```
gr120080186@cspro1 ~ $ ls
```

입력이 끝나면 Ctrl + D를 눌러 내용을 저장한다.

```
catExample test1
```

```
gr120080186@cspro1 ~ $ cat catExample
```

cat 다음에 열어볼 파일명을 쓰면, 파일의 저장된 내용을 확인할 수 있다.

```
Database
```

```
Data Structures
```

```
gr120080186@cspro1 ~ $ cat < catExample > Redirection
```

```
gr120080186@cspro1 ~ $ ls
```

```
Redirection catExample test1
```

기호 "<" 다음의 catExample의 내용을 받아서, 기호 ">" 다음의 Redirection 파일에 저장한다.

## 리다이렉션(Redirection) 기호

>, < 는 리다이렉션 기호라고 합니다. 이 문자를 사용해서 입력(<)과 출력(>)을 할 수 있습니다.



# Linux 명령어 실습

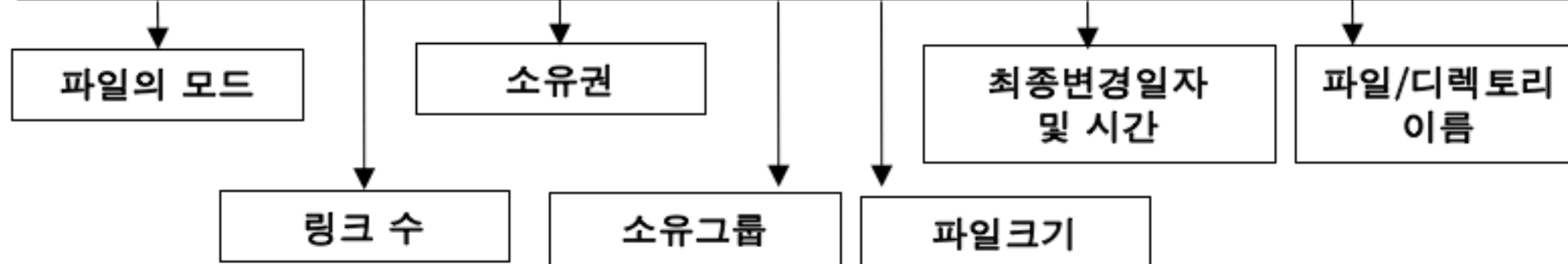
- ls
  - 현재 디렉토리의 파일리스트 보기 명령어 (list의 약자)

**ls** 파일 이름 및 디렉토리 이름만 출력

```
gr120080186@cspro1 ~ $ ls
test1 test2
```

**ls -l** 파일의 모드, 소유권, 파일크기, 최종변경 일자 및 시간을 출력

```
gr120080186@cspro1 ~ $ ls -l
total 0
drwxr-xr-x 2 gr120080186 501 48 Nov 14 13:05 test1
drwxr-xr-x 2 gr120080186 501 48 Nov 14 13:05 test2
```



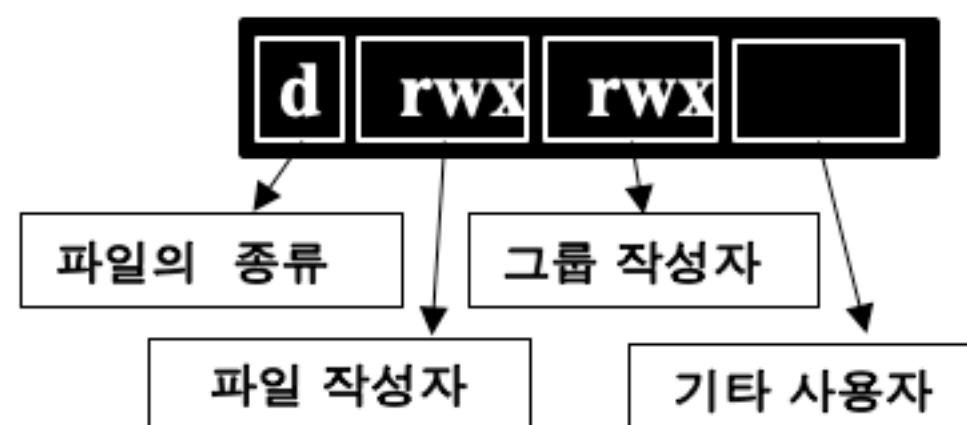
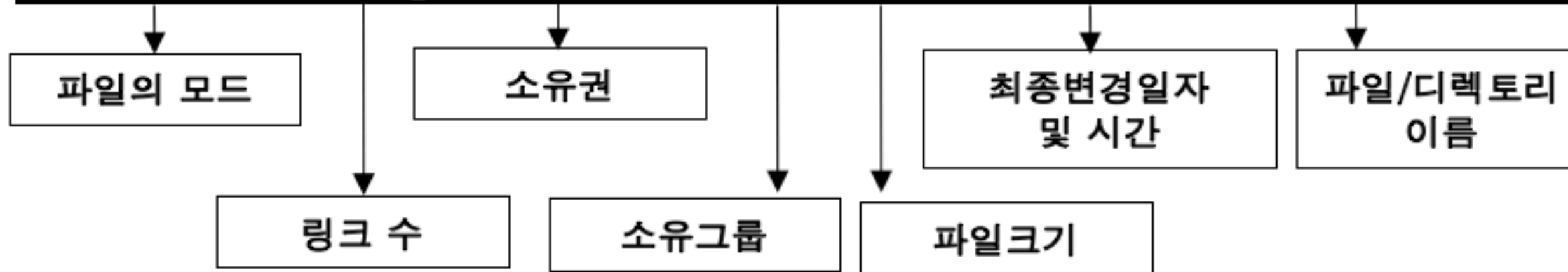
**ls -a** 숨겨진 파일(.(dot)으로 시작하는 파일)들까지 출력

```
gr120080186@cspro1 ~ $ ls -a
.  .bash_history  .bash_profile  .gdbinit  .viminfo  test1
.. .bash_logout  .bashrc        .ssh       .xsession  test2
```

# Linux 명령어 실습

- chmod
  - 파일의 권한을 변경하는 명령어
    - ls -l 명령을 사용하여 리눅스 파일들을 열람하면 많은 정보를 확인 할 수 있음
    - 이 정보를 통해, 이 파일이 누구의 소유인지 알 수 있음

```
gr120080186@cspro1 ~ $ ls -l
total 0
drwxr-xr-x  2 gr120080186 501 48 Nov 14 13:05 test1
drwxr-xr-x  2 gr120080186 501 48 Nov 14 13:05 test2
```



맨 처음 문자 `d`는 디렉토리를 의미하고, 만약 이 위치에 `-`이 표시되어 있다면 파일이라는 의미이다.  
그 다음 문자는 차례로 파일의 작성자, 작성자가 속한 그룹, 그 외의 기타 사용자에게 대한 읽기(`r`), 쓰기(`w`), 실행(`x`)의 허가 유무를 나타낸다.  
각 기호(`rwx`)가 표시되어 있다면 읽기, 쓰기, 실행에 대한 권리를 허용한다는 것이고, `rwx` 기호 대신 이 자리에 `-`가 표시되어 있다면 읽기, 쓰기, 실행의 각 권한이 거부되어 있다는 의미이다.



# Linux 명령어 실습

- mkdir

- 디렉토리를 생성하는 명령어 (make directory의 약자)

**mkdir** [*directory\_name*]

directory\_name으로 디렉토리를 만든다.

```
gr120080186@cspro1 ~ $ mkdir test1
gr120080186@cspro1 ~ $ ls
test1
```

```
gr120080186@cspro1 ~ $ mkdir test2
gr120080186@cspro1 ~ $ ls
test1 test2
```

- rmdir

- 디렉토리를 삭제하는 명령어

**rmdir** [*directory\_name*]

디렉토리를 삭제한다.

```
gr120080186@cspro1 ~ $ rmdir test1
gr120080186@cspro1 ~ $ ls
test2
```

```
gr120080186@cspro1 ~ $ rmdir test2
rmdir: `test2': Directory not empty
```

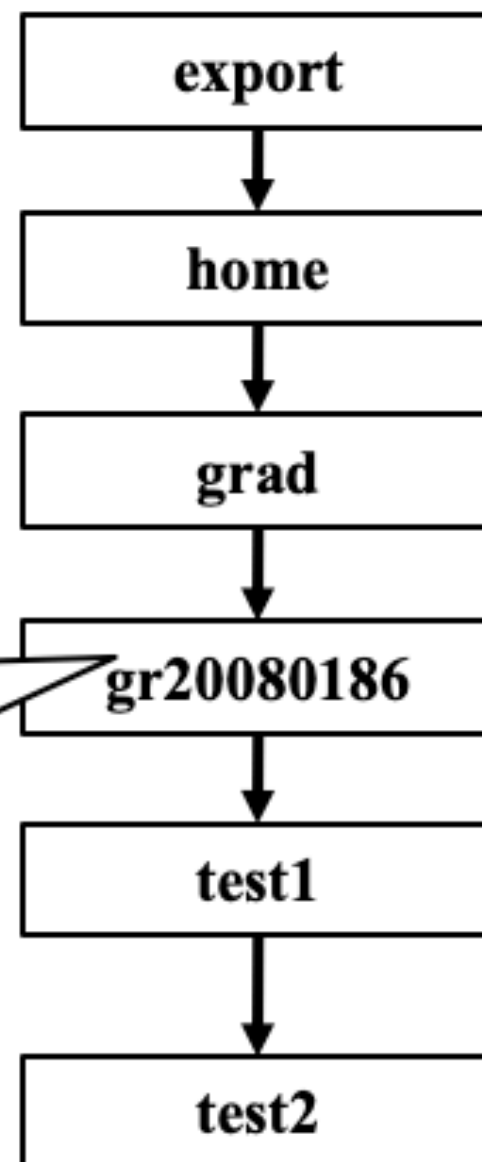
디렉토리 test2안에 파일이나 다른 디렉토리가 존재한다면, rmdir로 삭제할 수 없다.

# Linux 명령어 실습

- cd
  - 다른 디렉토리로 이동하는 명령어 (change directory의 약자)

`cd [test1] / cd [..] / cd`

“cd test1”, “cd ..”, “cd” 모두 다른 기능을 수행한다.



```
login as: gr120080186
Using keyboard-interactive authentication.
Password:
Last login: Thu Nov 13 22:29:48 2008 from mclab21.
gr120080186@cspro1 ~ $ pwd
/export/home/grad/gr120080186
```

로그인 후의  
위치가  
홈 디렉토리

```
gr120080186@cspro1 ~ $ cd test1
gr120080186@cspro1 ~/test1 $ pwd
/export/home/grad/gr120080186/test1
```

test1디렉토리  
이동  
cd test1

```
gr120080186@cspro1 ~/test1 $ cd test2
gr120080186@cspro1 ~/test1/test2 $ pwd
/export/home/grad/gr120080186/test1/test2
```

test2디렉토리  
이동

```
gr120080186@cspro1 ~/test1/test2 $ cd ..
gr120080186@cspro1 ~/test1 $ pwd
/export/home/grad/gr120080186/test1
```

상위디렉토리  
로 가는 명령어  
cd ..

```
gr120080186@cspro1 ~/test1 $ cd
gr120080186@cspro1 ~ $ pwd
/export/home/grad/gr120080186
```

홈디렉토리로  
가는 명령어  
cd

로그인 후의 위치  
(사용자의  
홈 디렉토리)

# Linux 명령어 실습

- pwd
  - 현재 자신이 속한 디렉토리의 위치를 알려주는 명령어 (present working directory의 약자)

pwd

```
gr120080186@cspro1 ~ $ pwd
/export/home/grad/gr120080186
gr120080186@cspro1 ~ $ ls
test2
gr120080186@cspro1 ~ $ cd test2
gr120080186@cspro1 ~/test2 $ pwd
/export/home/grad/gr120080186/test2
gr120080186@cspro1 ~/test2 $ mkdir test3
gr120080186@cspro1 ~/test2 $ ls
test3
gr120080186@cspro1 ~/test2 $ cd test3
gr120080186@cspro1 ~/test2/test3 $ pwd
/export/home/grad/gr120080186/test2/test3
gr120080186@cspro1 ~/test2/test3 $
```

pwd를 이용하여 현재 내 디렉토리를 확인해보자.

내 디렉토리 밑에 있는 test2 디렉토리로 이동하자.  
"cd"에 대해서는 다음장에서 배워보자

test2 디렉토리 밑에 test3 디렉토리를 만  
들어보자.

"cd test2" 실행 후에, 내 현재 위치가  
"test2" 인지 확인해보자.

# Linux 명령어 실습

- rm
  - 파일 삭제 명령어 (remove의 약자)
  - 테스트 파일을 만든 후 실습

```
gr120080186@cspro1 ~ $ ls
test2
gr120080186@cspro1 ~ $ cat > testfile
Sogang Univ.
Computer Science
C language
gr120080186@cspro1 ~ $ ls
test2  testfile
gr120080186@cspro1 ~ $ cat testfile
Sogang Univ.
Computer Science
C language
```

입력이 끝나면 Ctrl + D를 눌러 내용을 저장한다.

testfile의 내용을 확인한다.  
Cat명령어에 대해서는 뒤에서 자세히 다루도록 하겠다.

**rm [filename]** filename을 삭제

```
gr120080186@cspro1 ~ $ rm testfile
gr120080186@cspro1 ~ $ ls
test2
gr120080186@cspro1 ~ $
```

*Linux에는 윈도우 OS에 있는 휴지통이 없다. 즉, 한번 삭제된 파일은 복원할 수 없다.  
그러므로, rm 명령어를 사용할 때는 주의를 기울여야 한다.*



# Linux 명령어 실습

- cp
  - 파일을 복제하는 명령어 (copy의 약자)

```
gr120080186@cspro1 ~ $ cat > testfile1
Sogang Univ.
Computer Science
gr120080186@cspro1 ~ $
test1 testfile1
```

입력이 끝나면 Ctrl + D를 눌러 내용을 저장한다.

**cp [filename1] [filename2]** filename1을 filename2로 복사

```
gr120080186@cspro1 ~ $ cp testfile1 testfile2
gr120080186@cspro1 ~ $ ls
test1 testfile1 testfile2
gr120080186@cspro1 ~ $ cat testfile2
Sogang Univ.
Computer Science
```

```
gr120080186@cspro1 ~ $ cp testfile1 test1/testfile2
gr120080186@cspro1 ~ $ cd test1
gr120080186@cspro1 ~/test1 $ ls
test2 testfile2
```

특정 디렉토리로 복사를 할 수 있다.  
위의 예제는 test1 디렉토리로 testfile1을 복사하는 것을 보여준다.

# Linux 명령어 실습

- mv
  - 파일을 이동하는 명령어 (move의 약자)

`mv [filename1] [filename2]` filename1을 filename2라는 파일 이름으로 이동시킨다.

```
gr120080186@cspro1 ~/test1 $ ls
test2  testfile2
gr120080186@cspro1 ~/test1 $ mv testfile2 sample
gr120080186@cspro1 ~/test1 $ ls
sample  test2
```

```
gr120080186@cspro1 ~/test1 $ ls
sample  test2
gr120080186@cspro1 ~/test1 $ mv sample ..
gr120080186@cspro1 ~/test1 $ ls
test2
gr120080186@cspro1 ~/test1 $ cd ..
gr120080186@cspro1 ~ $ ls
sample  test1  testfile1  testfile2
```

현재 디렉토리에 있는 sample 파일을  
상위 디렉터리( .. )로 이동 시킨다.

🤔 cp 명령어와 mv 명령어의 차이점은 무엇일까?



# Linux 명령어 실습

- mv
  - 파일을 이동하는 명령어 (move의 약자)

`mv [filename1] [filename2]` filename1을 filename2라는 파일 이름으로 이동시킨다.

```
gr120080186@cspro1 ~/test1 $ ls
test2  testfile2
gr120080186@cspro1 ~/test1 $ mv testfile2 sample
gr120080186@cspro1 ~/test1 $ ls
sample  test2
```

```
gr120080186@cspro1 ~/test1 $ ls
sample  test2
gr120080186@cspro1 ~/test1 $ mv sample ..
gr120080186@cspro1 ~/test1 $ ls
test2
gr120080186@cspro1 ~/test1 $ cd ..
gr120080186@cspro1 ~ $ ls
sample  test1  testfile1  testfile2
```

현재 디렉토리에 있는 sample 파일을  
상위 디렉터리( .. )로 이동 시킨다.

🤔 cp 명령어와 mv 명령어의 차이점은 무엇일까?

# Linux 명령어 실습

- man
  - 명령어의 도움말을 보여주는 명령어 (manual의 약자)

**man** [*command\_name*] 명시한 명령어의 내용을 출력한다.

```
gr120080186@cspro1 ~ $ man ls
LS (1)
NAME
    ls, dir, vdir - list directory contents

SYNOPSIS
    ls [options] [file...]
    dir [file...]
    vdir [file...]

    POSIX options: [-CFRacdilqrtl] [--]

    GNU options (shortest form): [-labcdfghiklmnopqrst
    [-w cols] [-T cols] [-I pattern] [--full-time] [--
    [--block-size=size] [--format={long,verbose,commas,a
    gle-column}] [--sort={none,time}]
    [--time={atime,access,use,ctime,status}] [--color[=
    [--help] [--version] [--]
```

명령어 ls에 대한 도움말을 찾아보자.

man 화면에서의 주요키(key)

Q: 종료

[space]: 한 페이지 넘기기

[enter]: 다음 한 줄 이동

[B]: 한 페이지 위로 올라가기

# vi 편집기 실습

- vi
  - vi를 입력하면 vi 편집기가 실행된다.

```
gr120080186@cspro1 ~ $ vi

      VIM - Vi IMproved

      version 6.3.84
      by Bram Moolenaar et al.
Vim is open source and freely distributable


      Help poor children in Uganda!
type  :help iccf<Enter>          for information

type  :q<Enter>                  to exit
type  :help<Enter> or <F1>       for on-line help
type  :help version6<Enter>    for version info
```

[ vi 실행 후, 첫 화면 ]

# vi 편집기 실습

- vi 시작하기
  - vi 파일명 <enter>

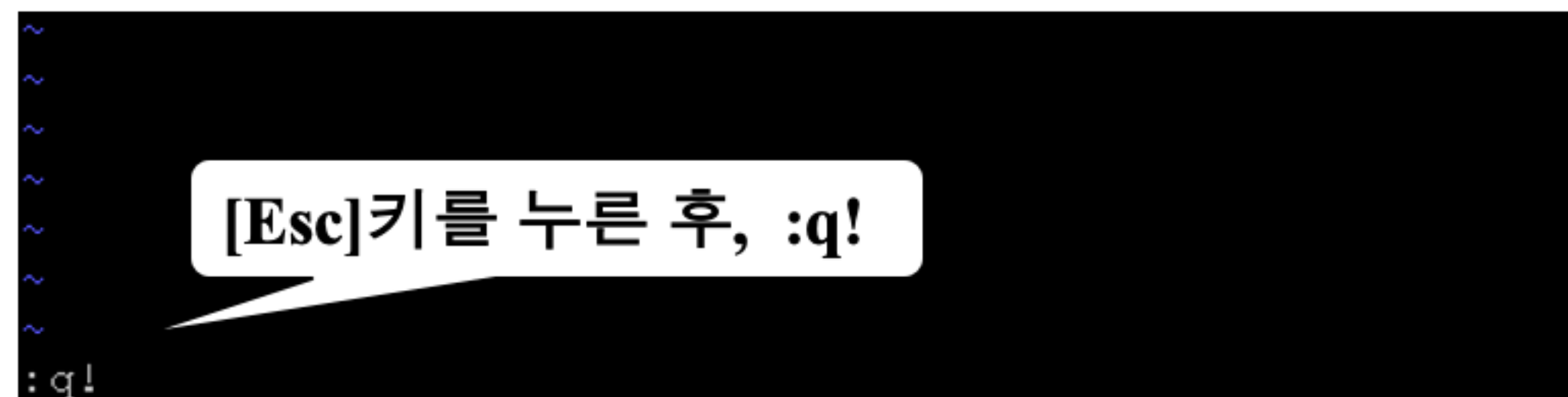
[illegible]

# vi 편집기 실습

- vi 종료하기
  - 먼저 [Esc]키를 누른 후에 아래와 같은 명령을 입력

명령어	내용
:q	그대로 종료하기
:q!	변경된 내용을 저장하지 않고 종료하기
:wq	변경된 내용을 저장하고 종료하기
:x	변경된 내용을 저장하고 종료하기
ZZ	변경된 내용을 저장하고 종료하기

- 이전에 실행한 예제 파일을 그대로 종료해보자



# vi 편집기 실습

- vi의 모드 (mode)
  - 명령모드
    - 입력된 텍스트에 대해 여러 가지 명령을 내릴 수 있는 모드
  - 입력 모드
    - 실제 텍스트를 입력시키는 모드
  - 커서 이동 모드
    - 문장의 위아래나 좌우로 커서를 이동하게 하는 모드
- 모드 사이의 전환은 콜론(:)과 [Esc]키, 텍스트 입력 키를 이용

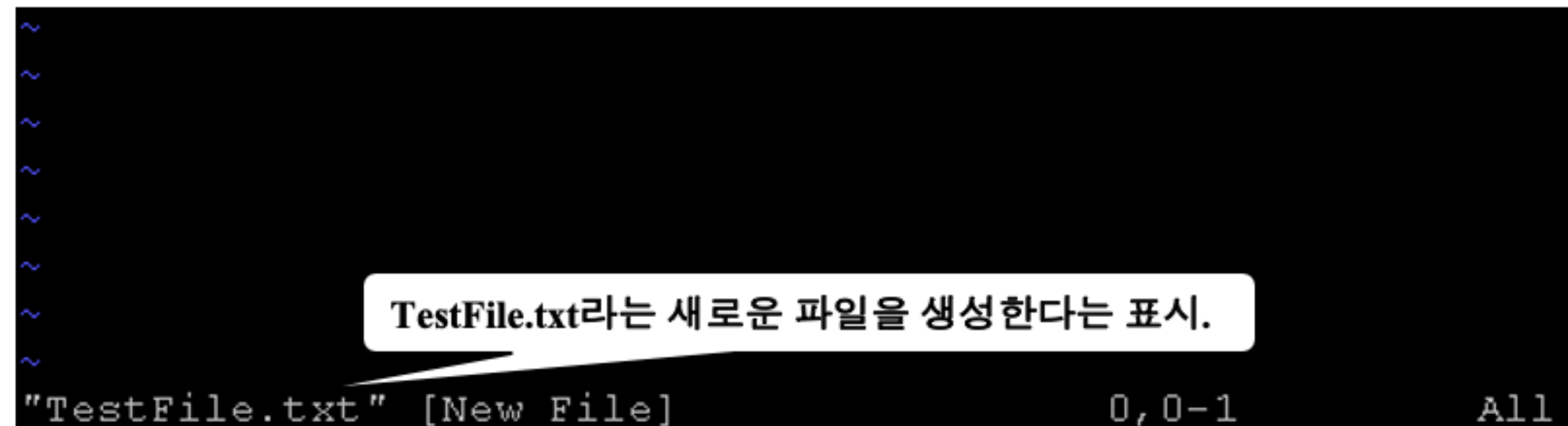


# vi 편집기 실습

- vi 실습
  - 다음의 과정을 하나씩 따라 하며, vi의 세 가지 모드에 대해서 배워보자

1. vi로 만들게 될 파일의 이름을 입력한다.

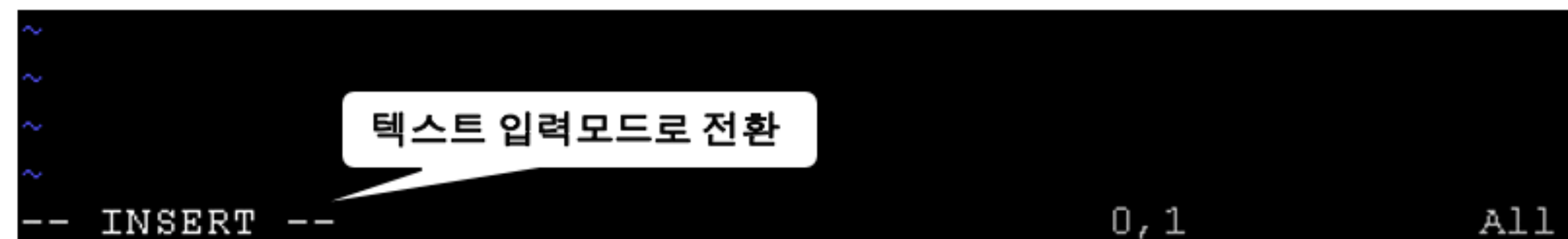
```
gr120080186@cspro1 ~ $ vi TestFile.txt
```



TestFile.txt라는 새로운 파일을 생성한다는 표시.

"TestFile.txt" [New File] 0,0-1 All

2. i나 [Insert]키를 누르면 텍스트 입력모드로 전환한다.



텍스트 입력모드로 전환

-- INSERT -- 0,1 All

# vi 편집기 실습

3. 이제 실제로 텍스트를 입력해보자. 다음의 내용을 모두 입력 후, [Esc]키를 누른다.

```
Test File

There are so many ways that the Internet has improved our
lives. I think the biggest improvement is that the Internet
has enabled us to find information instantly.
Indeed, online searches have become a huge part of how we
get infomation
~
~
```

[Esc]키를 누르면, -- INSERT - 라는 메시지가 사라지면서  
커서 이동 모드로 전환한다.

8, 0-1 All

4. 내용을 저장하기 위해서는 저장 명령을 내릴 수 있는 명령 모드로 변경해야 한다.  
:(콜론)을 입력하여 명령모드로 전환한다.

```
Test File

There are so many ways that the Internet has improved our
lives. I think the biggest improvement is that the Internet
has enabled us to find information instantly.
Indeed, online searches have become a huge part of how we
get infomation
~
~
```

명령모드로 전환

# vi 편집기 실습

5. 이제 “변경된 내용을 저장하고 종료하기” 명령인 wq를 입력한다.

```
Test File

There are so many ways that the Internet has improved our
lives. I think the biggest improvement is that the Internet
has enabled us to find information instantly.
Indeed, online searches have become a huge part of how we
get infomation

~
~
:wq
```

6. TestFile.txt 파일이 생성되었는지 확인해보자.

```
gr120080186@cspro1 ~ $ ls
TestFile.txt
gr120080186@cspro1 ~ $ cat TestFile.txt
Test File

There are so many ways that the Internet has improved our
lives. I think the biggest improvement is that the Internet
has enabled us to find information instantly.
Indeed, online searches have become a huge part of how we
get infomation

gr120080186@cspro1 ~ $
```

# vi 편집기 실습

7. 파일의 내용을 수정하려면 파일을 처음 만들 때처럼 다음과 같이 입력하면 된다.

```
gr120080186@cspro1 ~ $ vi TestFile.txt
Test File

There are so many ways that the Internet has improved our
lives. I think the biggest improvement is that the Internet
has enabled us to find information instantly.
Indeed, online searches have become a huge part of how we
get infomation
```

# vi 편집기 실습

- 다양한 vi 명령
  - 이제부터 명령 모드에서 입력 모드로 전환하는 키와 각 모드에서 사용되는 명령키를 알아보자
- 입력 키의 종류
  - 명령모드에서 텍스트 모드로 전환하려면 [Insert]키를 누르면 된다.
  - 이 밖에도 vi 편집기에서는 여러 가지 기능키를 제공

a	커서 위치의 다음 칸부터 입력하기
A	커서가 있는 줄의 끝부터 입력하기
i	커서 위치부터 입력하기(키보드의 [Insert]키도 같은 기능을 합니다.)
I	커서가 있는 줄의 맨 앞에서부터 입력하기
o	커서 바로 아래에 줄을 만들고 입력하기(open line)
O	커서 바로 위에 줄을 만들고 입력하기
s	커서가 있는 단어를 지우고 입력하기
S	커서가 있는 행을 지우고 입력하기

★ 대문자와 소문자 키의 기능이 다르다는 것을 잊지 말자.



# vi 편집기 실습

- 이동 키의 종류
  - [Esc]키를 이용하면 커서 이동모드로 돌아온다.
  - 커서의 이동은 윈도우와 마찬가지로 방향키 (←, →, ↓, ↑)를 사용

## ❖ 방향키가 안되는데요?

커서 이동키로 이동하는 것도 가능하지만 어떤 리눅스 시스템에서는 Vi를 실행했을 때 커서 이동키가 작동하지 않습니다. 그럴 경우에는 h (←), l (→), j (↓), k (↑)를 이용해 방향을 이동하세요.

## ▼ 그 밖의 커서 이동키 열람

글자 단위	h	한 칸 왼쪽으로 이동
	L	한 칸 오른쪽으로 이동
	J	한 줄 아래로 이동
	K	한 줄 위로 이동
단어 단위	w, W	다음 단어의 첫 글자로 이동(W는 특수문자 무시)
	b, B	이전 단어의 첫 글자로 이동(B는 특수문자 무시)
	e, E	단어의 마지막 글자로 이동(E는 특수문자 무시)
줄 단위	^	그 줄의 첫 글자로 이동
	\$	그 줄의 마지막 글자로 이동
	0	(숫자) 그 줄의 처음으로 이동
	+	다음 줄의 첫 글자로 이동
	-	윗줄의 첫 글자로 이동



# vi 편집기 실습

- 삭제 키의 종류
  - 기본 삭제 키
    - [Del]키 : 커서 이동 모드에서, 한 글자 삭제
    - [Backspace]키 : 입력 모드에서, 한 글자 삭제
  - 추가 삭제 명령키 (커서 이동모드에서)

---

x	커서 위치의 글자 삭제
X	커서 바로 앞의 글자 삭제
dw	한 단어를 삭제
d0	커서 위치부터 줄의 처음까지 삭제
D	커서 위치부터 줄의 끝까지 삭제
dd	커서가 있는 줄을 삭제
dj	커서가 있는 줄과 그 다음 줄을 삭제
dk	커서가 있는 줄과 그 앞줄을 삭제

---

# vi 편집기 실습

- 수정 키의 종류
  - 수정 명령키 (커서 이동모드에서)

---

r	커서 위치의 한 글자 수정하기
R	커서 위치부터 [Esc]를 누를 때까지 다른 글자로 수정하기. 단, 같은 줄에만 해당
s	커서 위치의 한 글자를 여러 글자로 수정하기
ch	커서 바로 앞의 한 글자를 여러 글자로 수정하기
cw	커서 위치의 한 단어를 수정하기
c0	커서 위치부터 줄의 처음까지 수정하기
C	커서 위치부터 줄의 끝까지 수정하기
cc	커서가 있는 줄을 수정하기
cj	커서가 있는 줄과 그 다음 줄을 수정하기
ck	커서가 있는 줄과 그 앞줄을 수정하기
~	대문자를 소문자로, 소문자를 대문자로 수정한다.

---

```
There are many ways that the Internet has improved our
lives. I think the most important is that the Internet
has enabled information to be shared instantly.
Email have innovated the way we communicate with one
another.....
Indeed, online searches have become a part of how we
get infomation!!!!!!
```

s 입력 후, ...를 여러번 입력

r을 입력 후, n을 대문자로 변경

# vi 편집기 실습

- 복사와 붙여넣기 키의 종류
  - 텍스트의 삭제나 교체도 임시 기억 장소인 버퍼(buffer)에 저장된다.

---

yw	커서 위치부터 단어의 끝까지 복사
y0	커서 위치부터 줄의 처음까지 복사
y\$	커서 위치부터 줄의 끝까지 복사
yy	커서가 있는 줄 복사
yj	커서가 있는 줄과 그 다음 줄을 복사
yk	커서가 있는 줄과 그 앞줄을 복사
p	커서의 다음 위치에 붙여넣기
P	커서가 있는 위치에 붙여넣기

---

```
iNDEED, ONLINE SEARCHES HAVE BECOME A PART OF HOW WE GET IN
FOMATIOn!!!!!!!!!!
iNDEED, ONLINE SEARCHES HAVE BECOME A PART OF HOW WE GET IN
FOMATIOn!!!!!!!!!!
iNDEED, ONLINE SEARCHES HAVE BECOME A PART OF HOW WE GET IN
FOMATIOn!!!!!!!!!!
```

yy를 이용하여 커서가 있는 줄을 복사합니다. 그리고 p  
를 두번 눌러 붙여넣기를 실행합니다.

# vi 편집기 실습

- 취소 키의 종류
  - 작업 취소 명령

---

u	작업 취소(undo)
U	그 줄에 행해진 작업 모두 취소
.	조금 전에 했던 명령을 반복

---

- 검색 키의 종류
  - 검색 명령

---

/pattern	텍스트에서 앞으로 패턴 검색
>pattern	텍스트에서 뒤로 패턴 검색
n	앞 또는 이전 검색 반복
N	반대방향으로 이전 검색 반복
/	전 검색을 앞으로 반복
?	전 검색을 뒤로 반복

---

“information”이라는 단어를  
찾으려면,

```
~/information
```

# 리눅스 컴파일러 -gcc

- gcc
  - GNU에서 만든 C와 C++을 위한 컴파일러 (Compiler)
  - gcc는 많은 옵션들을 가지고 있다.

```
gr120080186@cspro1 ~ $ man gcc
GCC (1)                                GNU

NAME
    gcc - GNU project C and C++ compiler

SYNOPSIS
    gcc [-c|-S|-E] [-std=standard]
        [-g] [-pg] [-Olevel]
        [-Wwarn...] [-pedantic]
        [-Iidir...] [-Ldir...]
        [-Dmacro[=defn]...] [-Umacro]
        [-foption...] [-mmachine-option...]
        [-o outfile] infile...
```

gcc 옵션들

# 리눅스 컴파일러 -gcc

- 컴파일 명령 `gcc 파일명`
  - [gcc 파일명]을 입력하면 실행 파일 a.out이 생성
  - 실행 파일은 [./a.out]을 통해 실행
- 실행 파일명 지정 `gcc -o 실행파일명 소스파일명`
  - [gcc -o 실행파일명 소스파일명]을 입력하면 실행파일명으로 입력한 실행 파일이 생성
  - [./실행파일명]으로 실행

```
gr120200190@cspro:~$ ls
test.c
gr120200190@cspro:~$ gcc test.c
gr120200190@cspro:~$ ls
a.out test.c
```

```
gr120200190@cspro:~$ ./a.out
Hello, world!
```

```
gr120200190@cspro:~$ gcc -o test test.c
gr120200190@cspro:~$ ls
a.out test test.c
```

```
gr120200190@cspro:~$ ./test
Hello, world!
```

- 수업에서 진행하는 모든 실습은 cspro 상의 gcc 컴파일러를 기준으로 채점



# C프로그래밍 실습

# 실습 1

- "Hello, world!"를 출력하는 C파일을 작성하시오.
  - C파일은 vi를 이용해 생성
  - 컴파일은 gcc를 사용

```
1 #include <stdio.h>
2
3 int main(void){
4     printf("Hello, world!\n");
5
6     return 0;
7 }
```

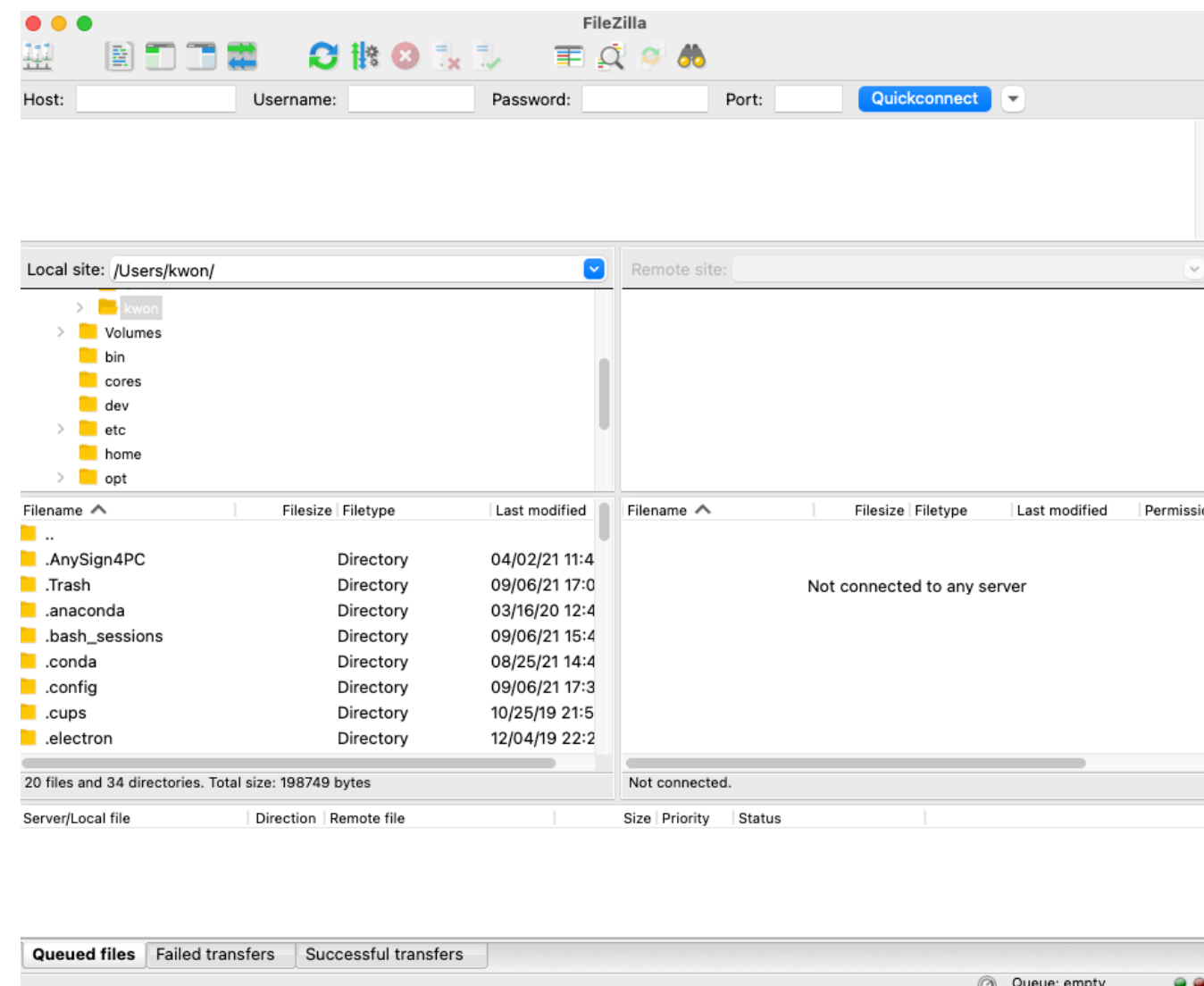
# 실습 2

- 공백으로 구분된 1개의 문자, 1개의 정수, 1개의 실수를 입력으로 받고, 이를 출력하는 프로그램을 작성하시오.
- 출력 예시

```
gr120200190@cspro:~$ vi codes/2.c
gr120200190@cspro:~$ gcc -o runs/2 codes/2.c
gr120200190@cspro:~$ ./runs/2
a 3 1.5
a 3 1.500000
gr120200190@cspro:~$ |
```

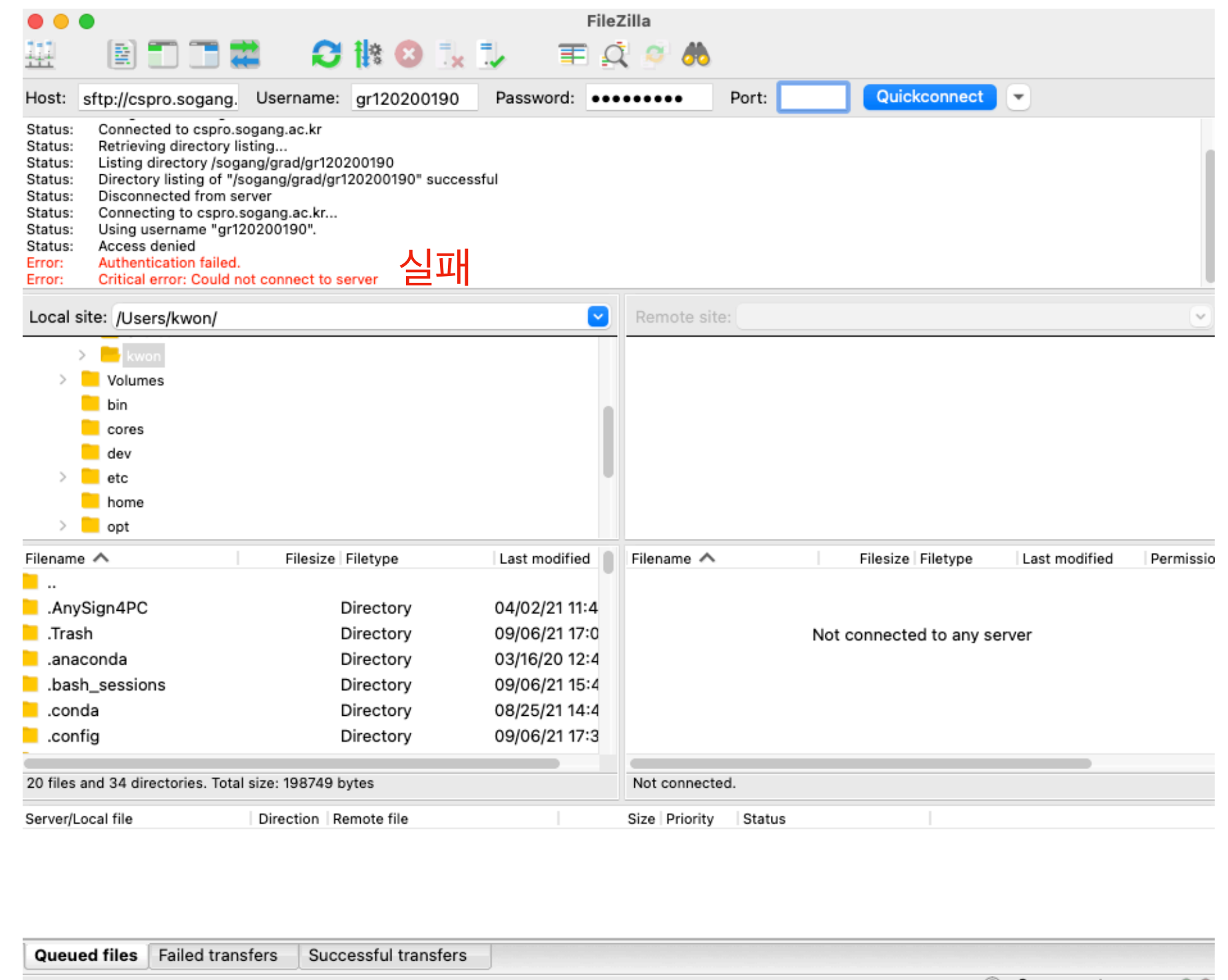
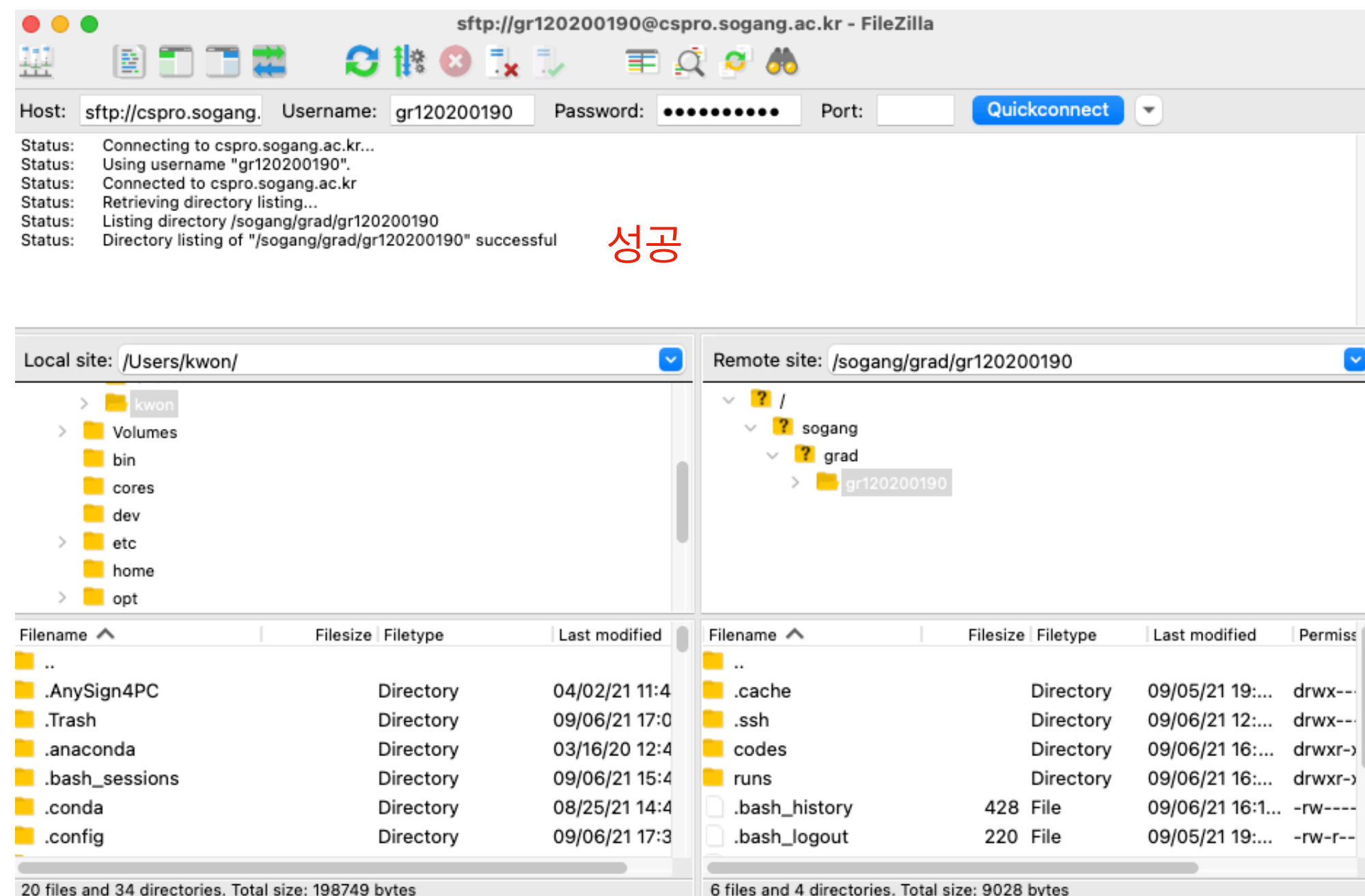
# 실습 제출 방법 1 (FileZilla)

- cspro 서버에 있는 코드를 내 PC로 옮겨와서 제출
- 이를 위해 FileZilla라는 tool을 이용
- 아래 링크를 통해 FileZilla를 다운로드 (본인의 운영체제에 맞는 버전을 다운로드)
- <https://filezilla-project.org/>
- 다운받은 FileZilla 프로그램을 실행하면 아래와 같은 화면이 등장

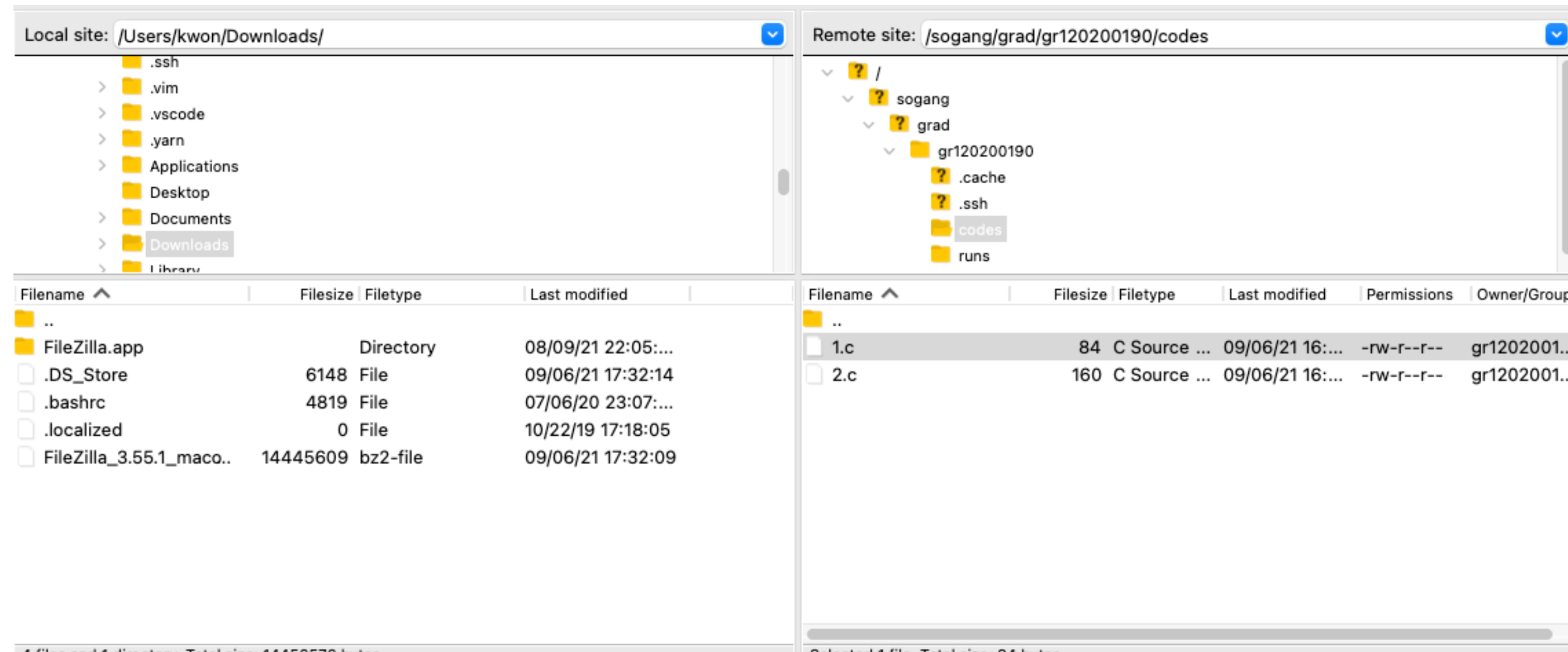


# 실습 제출 방법 1 (FileZilla)

- 아래 정보들을 입력해준 뒤 연결을 클릭
  - Host : cspro.sogang.ac.kr
  - Username : cse학번
  - Password : 본인이 설정한 비밀번호
  - Port : 22



# 실습 제출 방법 1 (FileZilla)



- 왼쪽이 본인의 PC, 오른쪽이 cspro 서버의 파일시스템
- 오른쪽에서 본인이 작성한 코드를 왼쪽으로 드래그해서 복사해주면 됨 (반대도 가능)
- 본인이 작성한 코드를 PC로 옮긴 뒤, 이를 사이버캠퍼스에 제출



# 실습 제출 방법 (scp)

- cspro 서버에 있는 코드를 내 PC로 옮겨와서 제출
- 이를 위해 scp 명령어를 이용
- 1. pwd로 파일의 경로 확인
  - 오른쪽 예시의 경우 /sogang/grad/gr202201/ex1.c
- 2. ssh 접속 종료(Ctrl+D) 후 다음 명령어 입력
  - `scp gr202201@cspro.sogang.ac.kr:/sogang/grad/gr202201/ex1.c ./ex1.c`
- 3. 로컬디렉토리에 파일이 잘 옮겨진 것을 확인

```
junsukchoe — gr202201@cspro: ~ — ssh gr202201@cspro.sogang.ac.kr —...
14:37:14
[> ssh gr202201@cspro.sogang.ac.kr
gr202201@cspro.sogang.ac.kr's password:
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-193-generic x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/advantage

246 packages can be updated.
114 updates are security updates.

New release '18.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Tue Mar 15 22:34:05 2022 from 163.239.24.175
gr202201@cspro:~$ pwd
/sogang/grad/gr202201
gr202201@cspro:~$ ls
ex1  ex1.c
gr202201@cspro:~$ pwd
/sogang/grad/gr202201
gr202201@cspro:~$
```

# 실습 제출 방법

- 각 실습의 파일명은 {학번}\_{실습번호}.c로 저장  
e.g.) 실습1의 파일명은 20211234\_1.c
- 작성한 c파일 코드들 (.c 파일들)을 사이버캠퍼스 과제란에 업로드
- 제출기한 : 사이버캠퍼스에 명시된 기한까지
- **지각제출은 불가능**
- **파일제목 등 제출 양식이 틀리면 오답처리**