

Creative Software Programming

Lab 0: Environment settings

ITE1015

2020 second semester

Jiun Bae (maytryark@gmail.com)

Topic

- OS / compiler
- Git
- IDE/texteditor

Computer programs sometimes do not work properly in other environments.

In this class, it is recommended to conduct the same environment as possible.

Keep in mind that code submitted in unequal environments may not work properly.

Ubuntu user

- 18.04 LTS is recommended

Windows user

- Using Virtualbox
- Using Windows Subsystem for Linux
- Using MinGW, Cygwin
- Using MSVC (Visual Studio)
- Highly recommended
- Highly recommended
- already familiar with gcc
- understands the difference of MSVC

macOS user

- Using Virtualbox
- Using clang
- Highly recommended
- Highly recommended

Using Virtualbox (for Windows, macOS user)

1. Install virtualbox (<https://www.virtualbox.org/>)
2. Download Ubuntu .iso file (<https://releases.ubuntu.com/18.04/>)
3. Setup virtualbox (<https://askubuntu.com/questions/142549/how-to-install-ubuntu-on-virtualbox>)

Using WSL (for Windows 10)

1. Require Windows 10 version above 14393 (can check 'winver')
2. Enable "Windows Subsystem for Linux" optional feature
 - a. Control panel / Programs / Turn Windows features on or off
 - b. Check Windows Subsystem for Linux and apply(require restart)
3. Setup follow (<https://docs.microsoft.com/en-us/windows/wsl/install-win10>)

Using Clang (for macOS)

1. Install 'Homebrew' package manager for macOS (<https://brew.sh/>)
2. type in terminal 'brew install gcc g++'

Install GCC



```
# Ubuntu
sudo apt install build-essential

# macOS
xcode-select --install
brew install gcc

# install (MinGW)
# http://www.mingw.org/

# install (Cygwin)
# https://www.cygwin.com/

# install (Visual Studio, MSVC)
# https://visualstudio.microsoft.com/
```




```
$ gcc --version

gcc (Ubuntu 9-20190428-1ubu...
Copyright (C) 2019 Free Sof...
This is free software; see ...
warranty; not even for MERC...

# recommend version above 7.4
```

Install git



```
# Ubuntu  
$ sudo apt install git
```

```
# macOS (brew)  
$ brew install git
```

```
# otherwise  
# download from  
# https://git-scm.com/downloads
```

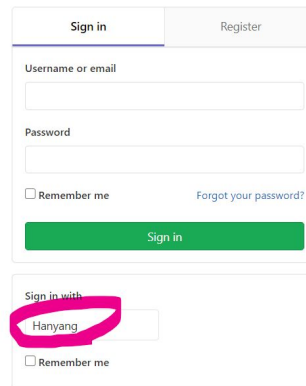


```
$ git --version  
  
git version 2.28.0
```


GitLab (hconnect.hanyang.ac.kr)

In this course, you should submit your assignments on the gitlab server of our department: (<https://hconnect.hanyang.ac.kr/>)

1. Click 'Hanyang' button below.
2. Login Hanyang portal account
 - a. After login, change password (settings / password)



Sign in Register

Username or email

Password

☐ Remember me [Forgot your password?](#)

Sign in

Sign in with

Hanyang

☐ Remember me

IDE/texteditor

- CLion
 - VS Code
 - Vim
 - Visual Studio
 - X code
- Highly recommended
 - Highly recommended
 - Difficult, but highly recommended
 - Only for Windows user, not recommend
 - Only for macOS user, not recommend

CLion

A Cross-Platform IDE for C and C++ by JetBrains, support student plans for free!

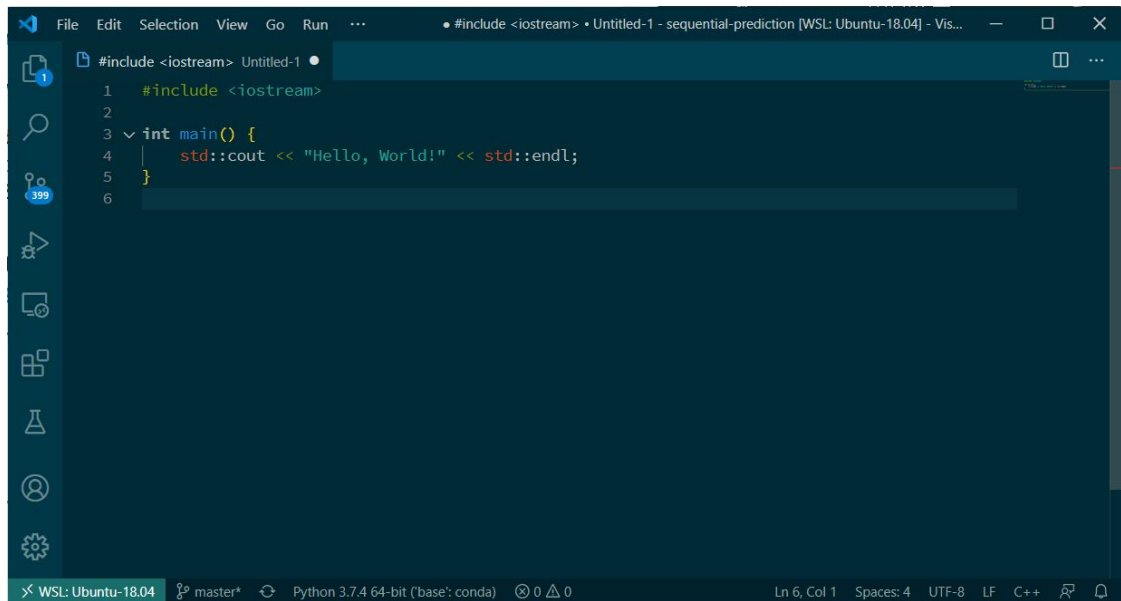
<https://www.jetbrains.com/community/education/#students>

1. Apply using 'hanyang.ac.kr' email address
 - a. may take few hours (or day?)
2. Download CLion (Ubuntu, Windows, macOS)

VS Code

VS Code is a code editor optimized for building and debugging. (also free!)

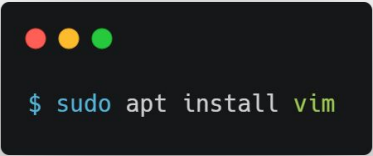
<https://code.visualstudio.com/>



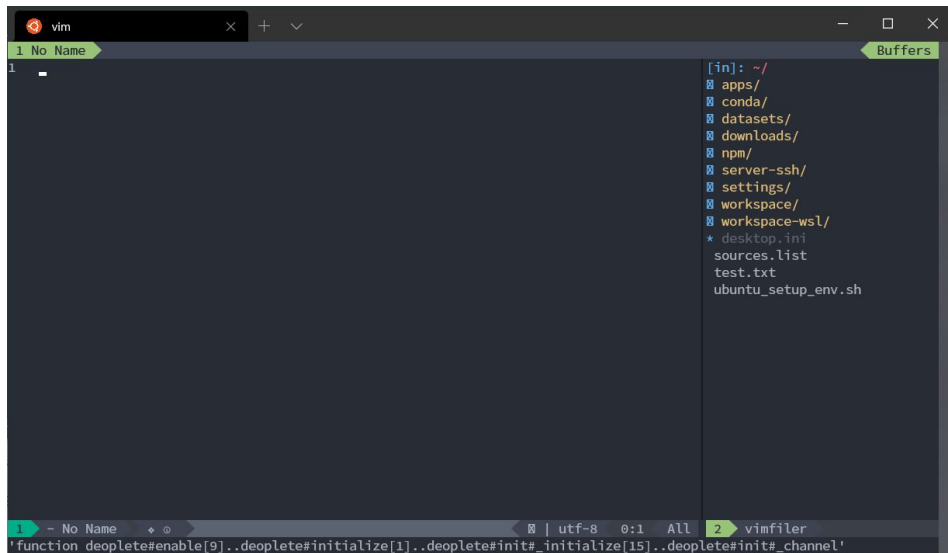
Vim

Vim is a highly configurable text editor. (traditional and often used today)

But very difficult for beginners (not for GUI, operate only with the keyboard)



```
$ sudo apt install vim
```



vi / vim graphical cheat sheet

Esc

normal
mode

~ toggle case	! external filter	@ play macro	# prev ident	\$ eol	% goto match	^ "soft" bol	& repeat :s	* next ident	(begin sentence) end sentence	_ "soft" bol down	+ next line
\ goto mark	1	2	3	4	5	6	7	8	9	0 "hard" bol	- prev line	= auto ³ format
Q ex mode	W next word	E end word	R replace mode	T back 'till	Y yank line	U undo line	I insert at bol	O open above	P paste before	{ begin parag.	}	end parag.
q record macro	w next word	e end word	r replace char	t 'till	y yank ^{1,3}	u undo	i insert mode	o open below	p paste after ¹	[misc]	misc
A append at eol	S subst line	D delete to eol	F "back" find ch	G eof/ goto ln	H screen top	J join lines	K help	L screen bottom	. ex cmd line	' reg. ¹ spec	bol/ goto col	
a append	s subst char	d delete ^{1,3}	f find char	g extra ⁶ cmds	h ←	j ↓	k ↑	l →	: repeat t/T/f/F	' goto mk. bol	\ not used!	
Z quit ⁴	X back-space	C change to eol	V visual lines	B prev word	N prev (find)	M screen mid'l	< un- ³ indent	> indent ³	? find (rev.)			
Z extra ⁵ cmds	X delete char	c change ^{1,3}	V visual mode	b prev word	n next (find)	m set mark	reverse t/T/f/F	. repeat cmd	/ find			

motion moves the cursor, or defines the range for an operator

command direct action command, if **red**, it enters insert mode

operator requires a motion afterwards, operates between cursor & destination

extra special functions, requires extra input
q. commands with a dot need a char argument afterwards

bol = beginning of line, eol = end of line,
mk = mark, yank = copy

words: quux(foo, bar, baz);

WORDS: quux(foo, bar, baz);

Main command line commands ('ex'):

:w (save), :q (quit), :q! (quit w/o saving)
 :e f (open file f),
 :%s/x/y/g (replace 'x' by 'y' filewide),
 :h (help in vim), :new (new file in vim),

Other important commands:

CTRL-R: redo (vim),
 CTRL-F/-B: page up/down,
 CTRL-E/-Y: scroll line up/down,
 CTRL-V: block-visual mode (vim only)

Visual mode:

Move around and type operator to act on selected region (vim only)

Notes:

- (1) use "x before a yank/paste/del command to use that register ('clipboard') (x=a..z,") (e.g.: "ay\$ to copy rest of line to reg 'a')
- (2) type in a number before any action to repeat it that number of times (e.g.: 2p, d2w, 5i, d4j)
- (3) duplicate operator to act on current line (dd = delete line, >> = indent line)
- (4) ZZ to save & quit, ZQ to quit w/o saving
- (5) zt: scroll cursor to top, zb: bottom, zz: center
- (6) gg: top of file (vim only), gf: open file under cursor (vim only)

Vim

Esc
명령 모드

Vim 명령어 단축키

손에 잡히는 Vim 입문

~ 대소문자 전환 .` 표식으로 이동	! 외부 명령 1	@. 매크로 실행 2	# 이전 검색 3	\$ 행 끝으로 이동 4	% 짝 괄호 찾기 5	^ 첫 글자 6	& :s 반복 7	* 다음 검색 8	(문장 시작 9) 문장 끝 0 앞으로	_ 아래 행 - 이전 행	+ 다음 행 = 자동 들여쓰기
Q 실행 모드 q. 매크로 기록	W 다음 단어 (의미상) w 다음 단어	E 단어의 끝으로 e 단어의 끝으로	R 수정 모드 r. 한 문자 교체	T. 행에서 후방 검색 t. 행에서 전방 검색	Y 행 단위 복사 y 복사 ^{1 3}	U 행 단위 실행 취소 u 실행 취소	I 행 시작에 삽입 i 커서 앞에 삽입	O 행 위에 삽입 o 행 아래에 삽입	P 커서 이전에 붙여넣기 p 커서 이후에 붙여넣기 ¹	{ 문단 시작 [} 문단 끝]	
A 행 끝에 삽입 a 커서 뒤에 삽입	S 삭제 후 새 줄 입력 s 삭제 후 단어 입력 모드	D 행 삭제 d ^{1 3} 삭제	F. 행에서 후방 검색 f. 행에서 전방 찾기	G 파일 끝으로 이동 g. 확장 명령	H 화면 상단 h ←	J 아래 행 합치기 j ↓	K 도움말 k ↑	L 화면 하단 l →	: 명령행 모드 ; /T/t/F 반복	" 레지스터 지정 ' 표식으로 이동	I 열 이동 \ 사용하지 않음	
Z. 종료 ¹ z. 확장 명령 ⁵	X 백스페이스 x 글자 삭제	C 행 끝까지 바꾸기 c ^{1 3} 바꾸기	V 비주얼 라인 모드 v 비주얼 모드	B 이전 단어 (의미상) b 이전 단어	N 이전 (찾기) n 다음 (찾기)	M 화면 가운데 m 표시 설정	< 내어 쓰기 ⁵ , /T/t/F 반대 방향 검색	> 들어 쓰기 ⁵ . 명령 반복	? 찾기 (위쪽) , / 찾기 (아래쪽)			

동작

커서를 이동하거나, 연산자가 동작할 범위를 지정합니다.

명령

바로 동작하는 명령입니다.
빨간색은 입력 모드로 변경됩니다.

오퍼레이션 팬딩 모드

커서 위치부터 목적지까지를 대상으로 명령을 실행합니다.

확장

추가적인 키 입력이 필요합니다.

q.

입력 후 글자를 입력해야 합니다.

단어

공백 문자나 특수 문자로 구분된 단어
test(123, 456, 789)

단어 (의미상)

공백 문자로 구분된 단어
test(123, 456, 789)

• 명령행 모드의 주요 명령어

:w 저장
:q 종료
:q! 저장없이 종료
:ef f 파일 열기
:%s/x/y/g 파일 전체에서 'x'를 'y'로 교체
:h name name 명령에 대한 도움말
:new 새 파일

• 일반 모드의 주요 명령어

CTRL-R 재실행
CTRL-F/B 페이지 위로/아래로
CTRL-E/Y 스크롤 위로/아래로
CTRL-V 비주얼 모드

• 참고

- 복사/붙여넣기/지우기 명령을 사용하기 전에 "a"를 입력하여 레지스터 a를 지정할 수 있습니다. (레지스터 이름은 a부터 z까지 사용 가능)
예를 들어 "ay\$는 커서 위치부터 행 끝까지의 내용을 레지스터 a에 저장합니다.
- 명령어 입력 전 숫자를 지정하면, 해당 숫자만큼 명령어가 반복됩니다.
- 연속으로 입력하면, 현재 행에 반영됩니다.
예를 들어 dd는 현재 행이 지워집니다.
- ZZ는 저장 후 종료, ZQ는 저장 없이 종료입니다.
- zt는 커서가 위치한 부분을 화면 상단으로 스크롤합니다. zb는 바닥으로, zz는 가운데로 스크롤합니다.
- gg는 커서를 파일 처음으로 이동합니다. gf는 커서 위치의 파일명을 인식하여 파일을 엽니다.

Notice

1. I will create and announce a personal repository on GitLab. All future assignments must be uploaded to the repository.
2. This course is basically conducted in English. However, you can request a Korean explanation.
3. In order to facilitate non-face-to-face classes, questions are frequently asked through chat rooms. (<https://open.kakao.com/o/glEhaRtc>, or mailto: maytryark@gmail.com)

If possible, we plan to replace the class by uploading a video rather than a real-time video class and viewing the video within a specified period. I will notify you again when confirmed.