

Cygwin 설치 및 Vi 기본

DSLAB

조교 정보 & 과제 관련

- 과제 제출 및 수업 질문
 - 연구실: DSLab (신공학관 1219호)
 - E-mail: dslab.pp@gmail.com
 - 조교: 김민우, 손준익

Cygwin Install

1. Cygwin download

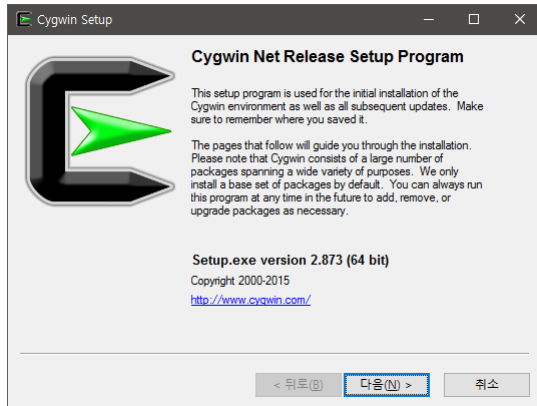
- Cygwin 공식 홈페이지 (<http://www.cygwin.com/>)에 접속
- 32bit/64bit를 구분하여 다운로드 (저장 위치는 자유롭게)



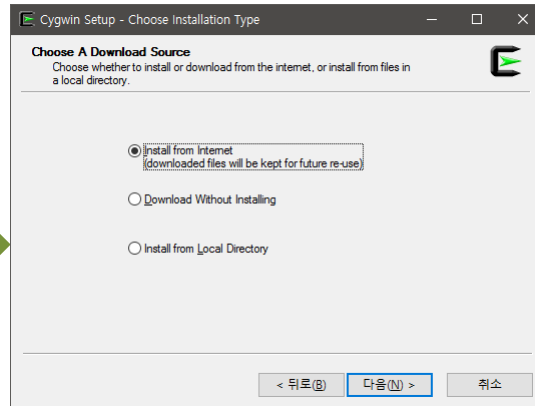
빠른 다운로드 위치

2. Cygwin Installation

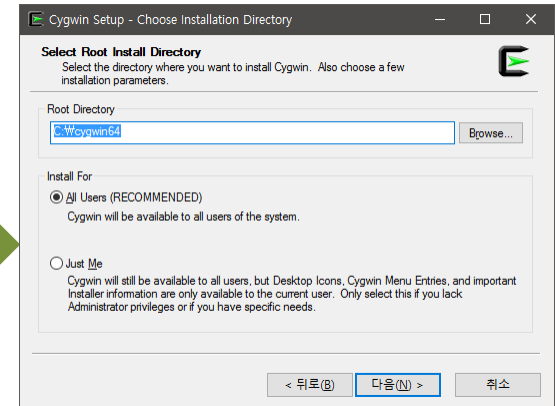
- 다운 받은 설치 파일(setup-x86(_64).exe)을 실행



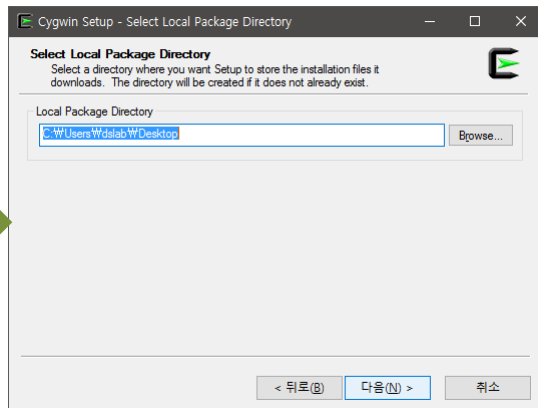
1. 다음



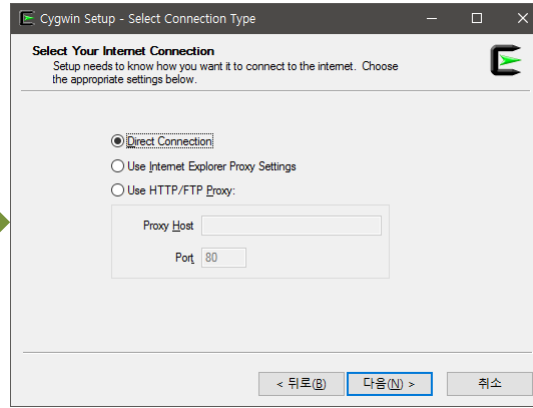
2. Install from Internet, 다음



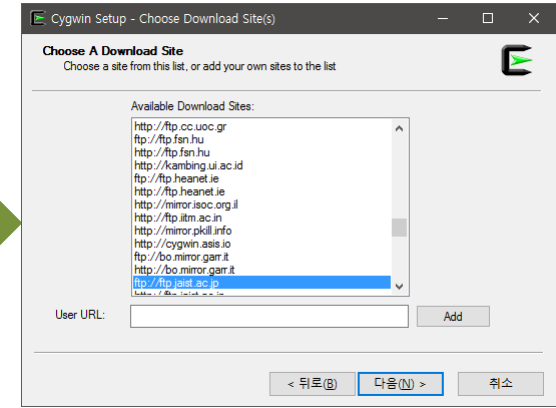
3. 다음 (설치 경로 변경 자유롭게)
("C:\cygwin64" 추천)



4. 다음



5. 다음

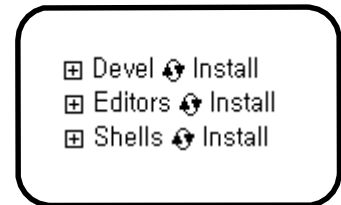
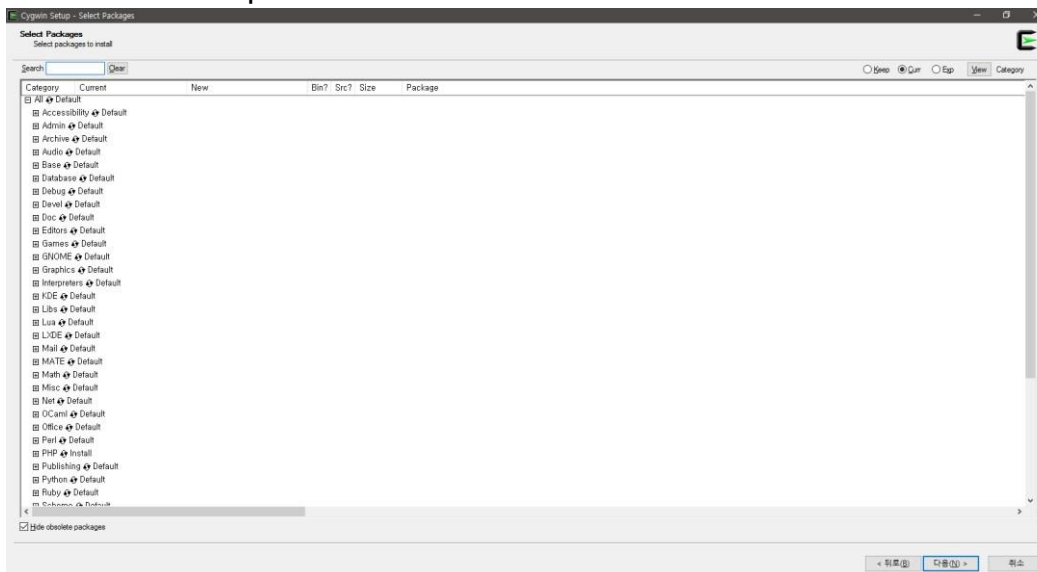


6. 다음 (아무거나 선택)

3. Package Selection

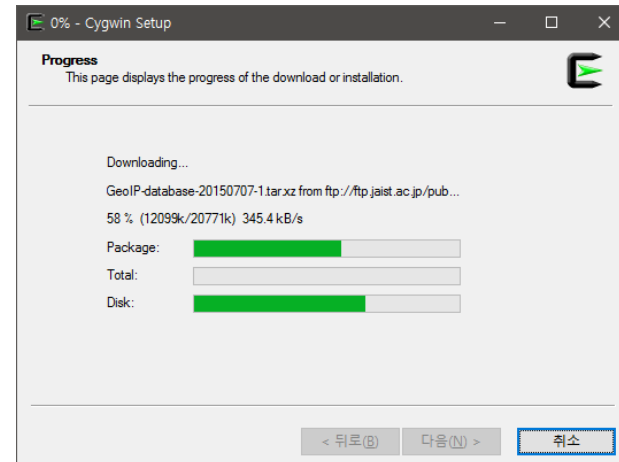
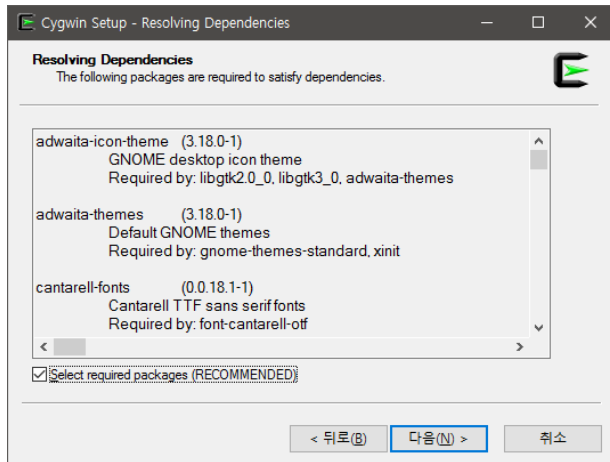
- 앞선 페이지에서 6번 수행 후 다운로드를 진행 중 아래와 같은 화면 출력
- Development, Editor, Shell 항목의 default를 클릭하여 Install로 변경
- 최소한의 설치를 위해서는 다음 패키지를 선택하여 설치
 - Devel
 - gcc-core: c compiler
 - gdb: The GNU Debugger
 - make: The GNU version of the 'make' utility
 - Editor
 - vim: vi Improved – enhanced vi editor

◎ 패키지는 추후 설치 파일을 통해 관리 가능



3. Package Selection (Cont.)

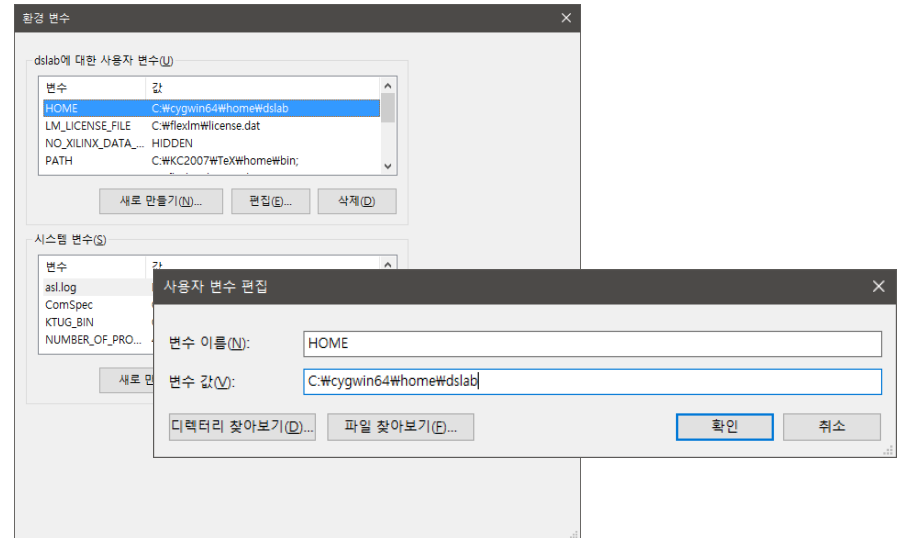
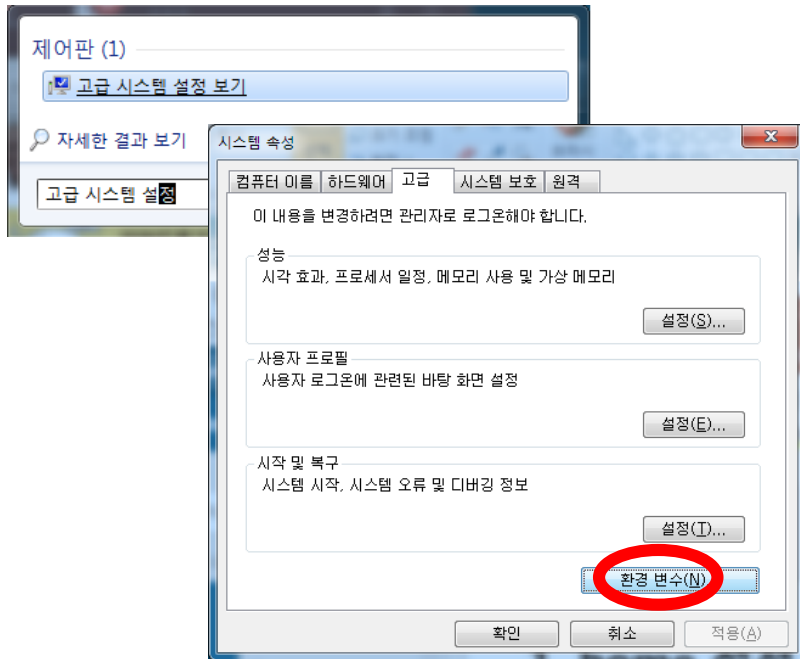
- 아래와 같은 확인 창이 나오면 '다음' 선택 후 다운로드 진행
- 인터넷 환경에 따라 속도에 따라 다른 설치 속도가 나옴



Cygwin Setting

1. home 설정

- 환경 변수에서 HOME 변수가 지정하는 디렉토리를 변경
 - 제어판 - 시스템 - 고급 시스템 설정 - 고급 - 환경 변수
 - 사용자 변수 내 HOME을 선택하여 변경 (없을 경우 새로 만들기)
 - Cygwin 설치 완료 시 자동 생성됨
 - 변경 후 Cygwin 재실행 (Cygwin terminal 실행 중일 경우)



2. .vimrc 설정

- **source .bashrc**를 입력하여 변경사항을 적용

```

ds lab@DESKTOP-2D85V68 ~
$ vi .bashrc

ds lab@DESKTOP-2D85V68 ~
$ source .bashrc

ds lab@DESKTOP-2D85V68 ~
$
  
```

2. .vimrc 설정

- **vi .vimrc** 입력 후 명령어를 입력하여 설정
- .vimrc 명령어
 - set nu 행 번호 표시
 - (set nonu 행 번호 표시 안함)
 - set list 탭 문자와 개행 문자를 각각 ^, \$로 표시
 - set autoindent 자동 들여쓰기
 - set wrap 자동으로 다음 행으로 넘어가는 기능을 사용
 - set ruler 화면 우측 하단에 현재 커서의 위치를 표시
 - set showmatch 상태 행에 현재 편집모드를 출력
 - set hlsearch 문자열 검색 시 검색어 강조 표시
 - set ignorecase 문자열 검색 시 대소문자 구별 x
 - set showmatch 커서 위치의 괄호와 짝을 이루는 괄호 표시
 - set bs=2 백스페이스 사용
 - syntax on 구문 강조 표시 사용
 - filetype on 파일 종류에 따라 구문 강조 표시 사용
 - etc..

vi

vi Editor 명령어

- C 파일 생성: vi filename.c
- 입력(편집) 모드
 - i: 현재 커서 위치 바로 앞에 내용 입력
 - I: 현재 행의 맨 앞에 내용 입력
 - a: 현재 커서 위치 바로 뒤에 내용 입력
 - A: 현재 행의 맨 뒤에 내용 입력
 - o: 커서 아래에 행을 추가하여 내용 입력
 - O: 커서 위에 행을 추가하여 내용 입력
 - <Esc>: 명령 모드로 전환
- 커서 이동
 - j,k,h,l: 아래, 위, 좌, 우 이동 (방향키도 가능)
 - ^, \$: 현재 행의 맨 앞, 뒤로 이동
 - gg, G: 문서의 맨 처음, 끝으로 이동

vi Editor 명령어

- 텍스트 삭제
 - x: 글자 하나 삭제
 - dw: 단어 하나 삭제
 - D: 현재 커서 위치에서 행 끝까지 삭제
 - dd: 한 행 삭제 (3dd: 3행 삭제) (1,10d: 1행에서 10행까지 삭제)
- 텍스트 대체(덮어쓰기)
 - r: 한 글자 대체
 - rw: 한 단어 대체
 - R: 현재 커서 뒤의 모든 글자를 대체
- 문자열 복사와 붙여넣기
 - v: 블록 지정 시작
 - <ctrl>+v: 사각형 모양으로 블록 지정
 - <shift>+v: 라인 블록
 - y: 블록 영역의 문자영을 버퍼에 복사
 - yy: 한 행을 버퍼에 복사
 - p, P: 현재 커서 뒤, 앞에 붙여 넣기

vi Editor 명령어

- 콜론 명령
 - :w : 문서 저장
 - :w filename : 문서를 filename 이름으로 저장
 - :q : vi 종료
 - :q! : 변경 내용을 무시하고 종료(강제종료)
 - :wq : 문서 저장 후 종료
 - :e filename : 지정한 파일 편집
 - :r filename : 지정한 이름의 파일 내용 읽기
 - :!<command> : 콘솔 명령어 실행
 - :r!<command> : 셸 명령 실행 후 결과를 현재 창에 삽입

vi Editor 명령어

vi 시작		텍스트 삭제	
vi filename	파일열기, 작성	x	문자 삭제
vi +18 filename	18행으로 파일 열기	dw	단어 삭제
vi +/"string" fn	"string"의 처음 발생 단어부터	dd	행 삭제
vi -r filename	손상된 파일 회복	D	커서 오른쪽 행 삭제
view filename	읽기 전용으로 파일 열기	:5,10 d	5-10번째 행 삭제
커서명령(이동)		텍스트 복사 및 이동	
h(←)	왼쪽으로 커서 이동	yy	행 yank 또는 복사
j(↓)	아래로 커서 이동	Y	행 yank 또는 복사
k(↑)	위로 커서 이동	dd	행 삭제
l(→)	오른쪽으로 커서 이동	P	yank되거나 삭제된 행 현재 행 위에 삽입
w	한 단어 오른쪽으로 커서 이동	p	yank되거나 삭제된 행 현재 행 아래에 삽입
b	한 단어 왼쪽으로 커서 이동	:1,2 co 3	1-2행을 3행 다음으로 복사
Return	한 행 아래로 커서 이동	:4,5 m 6	4-5행을 6행 위로 이동
Back Space	한 문자 왼쪽으로 커서 이동		
Space Bar	한 문자 오른쪽으로 커서 이동		
H	화면의 맨위로 이동		
M	화면의 중간으로 이동		
L	화면의 맨 아래로 이동		
Ctrl + F	한 화면 앞으로 이동		
Ctrl + D	반 화면 앞으로 이동		
Ctrl + B	한 화면 뒤로 이동		
Ctrl + U	반 화면 뒤로 이동		
문자와 행 삽입		행 번호 설정	
a	커서 오른쪽에 문자 삽입	:set nu	행 번호 표시
A	커서 오른쪽, 행의 끝에 문자 삽입	:set nonu	행 번호 숨기기
i	커서 왼쪽에 문자 삽입		
I	커서 왼쪽, 행의 처음에 문자 삽입		
o	커서 아래에 행 삽입		
O	커서 위에 행 삽입		
텍스트 변경		행 찾기	
cw (종료:ESC)	단어변경	G	파일의 마지막 행으로 가기
cc (종료:ESC)	행 변경	21G	파일의 21번째 행을 가기
C (종료:ESC)	커서 오른쪽의 행 변경		
s (종료:ESC)	커서가 위치한 문자열 대체		
r	커서 위치의 문자를 다른 문자로 대체		
r - Return	행 분리		
J	현재 행과 아래 행 결합		
xp	커서 위치 문자와 오른쪽 문자 교환		
~	문자형(대,소문자)변경		
u	이전 명령 취소		
U	행 변경 사항 취소		
:u	이전의 최종 행 취소		
.	이전 최종 명령 반복		
		탐사 및 대체	
		/string/	string 탐색
		?string?	string 역방향 탐색
		n(N)	string의 다음(이전) 계속 탐색
		:g/search-string/s//replace-string/gc	각 발생 탐색 후 확인하고 대체
		:s/str/rep	현재 행의 str을 rep로 대체
		:1..s/str/rep/	1부터 현재 행의 str을 rep로 대체
		:%s/str/rep/g	파일 전체 str을 rep로 전부 대체
		화면정리	
		:Ctrl-I	불필요한 화면정리 후 다시 표시
		파일들 파일로 삽입	
		:r filename	커서 다음에 파일 삽입
		:34 r filename	파일을 34번째 행 다음에 삽입
		보관 및 종료	
		:w	변경사항 보관
		:w filename	버퍼를 파일로 보관
		:wq	변경사항 보관 후 vi 종료
		ZZ	변경사항 보관 후 vi 종료
		:q!	변경사항 보관하지 않고 종료

컴파일 및 실행

- 컴파일 명령어
: gcc -o filename.exe filename.c
- Ex) gcc -o helloworld.exe helloworld.c
- 실행 명령어
: ./filename.exe or ./filename
- Ex) ./helloworld.exe

helloworld.c

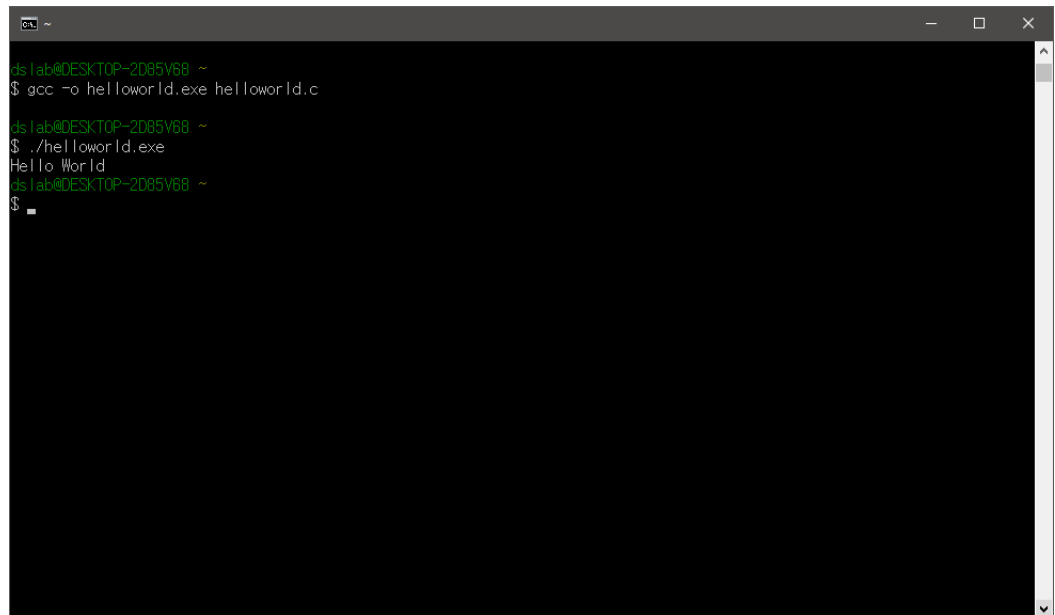
```
#include <stdio.h>

int main(){

    printf("Hello World!");

    return 0;

}
```



```
ds lab@DESKTOP-2D85V68 ~
$ gcc -o helloworld.exe helloworld.c

ds lab@DESKTOP-2D85V68 ~
$ ./helloworld.exe
Hello World
ds lab@DESKTOP-2D85V68 ~
$
```

컴파일 옵션

- -o : 출력 파일명을 지정할 때 사용
- -v : 컴파일 과정을 보여주는 옵션
- -c : Linking 과정을 진행하지 않고 .o 파일까지만 생성
- -S : 어셈블러까지 진행하지 않고 .S 파일을 생성
- -O1 ~ O3 : 최적화 수준 지정. 숫자가 클 수록 높은 수준의 최적화
- -g : 디버깅을 위한 정보를 컴파일 하면서 생성
- -D : define을 할 수 있는 옵션
- -I : 라이브러리 이름을 지정
- -L : 추가 라이브러리 디렉토리를 지정
- -w : 모든 에러 메시지 출력 하지 않음
- -W : 모든 에러 메시지 출력
- -I : 추가 헤더 파일이 있는 디렉토리를 지정