G++, Make, GDB

Overview

- G++
- Make
- GDB

G++

- Open sourced C++ compiler
- 대부분의 입출력 형식 및 옵션은 기본 C compiler (cc) 와 동일
- g++ [options] <infile> ...
  - -c : compile only. 실행파일을 만들지 않고 object file (.o) 까지만 생성
  - -g : debug info. Debugging에 필요한 정보(소스코드 등)를 포함
  - -o <outfile> : 컴파일한 output file. 또는 링크후 만들 파일의 이름
  - -I<dir> : include directory. 컴파일할 때 헤더를 찾아볼 디렉토리 이름
  - -L<dir> : library directory. 링크할 때 라이브러리 파일을 찾아볼 디렉토리 이름
  - -D<symbol>[=def] : define a macro. 컴파일 시 사용할 매크로 정의
  - ... : 이 외에도 수많은 옵션이 있음

Made by MDB

G++ example

```
$emacs main.cpp
```

```
void print_hello();

int main(){
    print_hello();
    return 0;
}
```

G++ example

```
$emacs print.cpp
```

```
#include <iostream>
#using namespace std;

void print_hello(){
    cout << "hello world" << endl;
}
```

G++ example

- 2개의 소스파일(main.cc, print.cc)을 컴파일 및 링크하기

```
$g++ -o hello_world main.cpp print.cpp
```

- 2개의 소스파일(main.cc, print.cc)을 먼저 컴파일한 후 링크하기

```
$g++ -c -o main.o main.cpp
$g++ -c -o print.o print.cpp
$g++ -o hello_world main.o print.o
```

Made by MDB

Make

- Unix 계열에서 오랫동안 사용된 빌드 툴
  - 소스를 어떻게 컴파일하고 링크해서 실행파일을 만들지에 대한 규칙

- Makefile
  - make 가 실행되면 해당 디렉토리에서 Makefile (또는 makefile)을 찾아 규칙대로 실행

Makefile 작성 방법

- target : 만들고자 하는 파일 또는 상태 (.o 또는 실행파일 등)
- prerequisites : target을 만드는데 필요한 파일의 목록
- command(s) : target을 만드는 각 단계별 명령어. 명령어 앞에 tab 이 있어야함

```
target: prerequistites
    command1
    command2
```

Makefile example

```
$emacs makefile
```

```
hello_world: main.o print.o
    g++ -o hello_world main.o print.o
main.o: main.cpp
    g++ -c main.cpp
print.o: print.cpp
    g++ -c print.cc
clean:
    rm hello main.o print.o
```

GDB

- 디버깅 툴 – 프로그램을 실행하면서 또는 프로그램이 비정상 종료했을 때 상태를 검사하여 잘못된 부분을 찾을 수 있도록 도와줌
- 프로그램을 빌드할 때 -g 옵션을 주어야 필요한 정보를 볼 수 있음

GDB

- gdb [options] <command>
  - <command> : 실행파일. 현재 디렉토리가 PATH에 들어있지 않으면 ./ 를 포함해야함
- 기본 명령
  - r [arguments] : 주어진 명령을 실행함
  - bt : backtrack. 현재 call stack 상태를 보여줌
  - up/down [steps] : call stack의 현재 위치에서 주어진 단계만큼 올라감/내려감
  - p <variable> : 주어진 변수의 값을 표시함
  - q : gdb 실행 종료
  - cgdb, ddd 등 좀 더 사용하기 쉬운 개량된 프로그램을 사용

Made by MDB

GDB example

```
$emacs test.cpp
```

```
void IncorrectAccess(int* array, int i, int n){
    if (i < n){
        array[i] = 0;
        IncorrectAccess(array, i + 1, n);
    }
}

int main(){
    int array[10];
    IncorrectAccess(array, 0, 20);
    return 0;
}
```

```
$g++ -o test test.cpp
$gdb ./test
...
(gdb) r
...
```