

Practice 10. Makefiles

Byoung-Tak Zhang

TA: Hanock Kwak

Biointelligence Laboratory

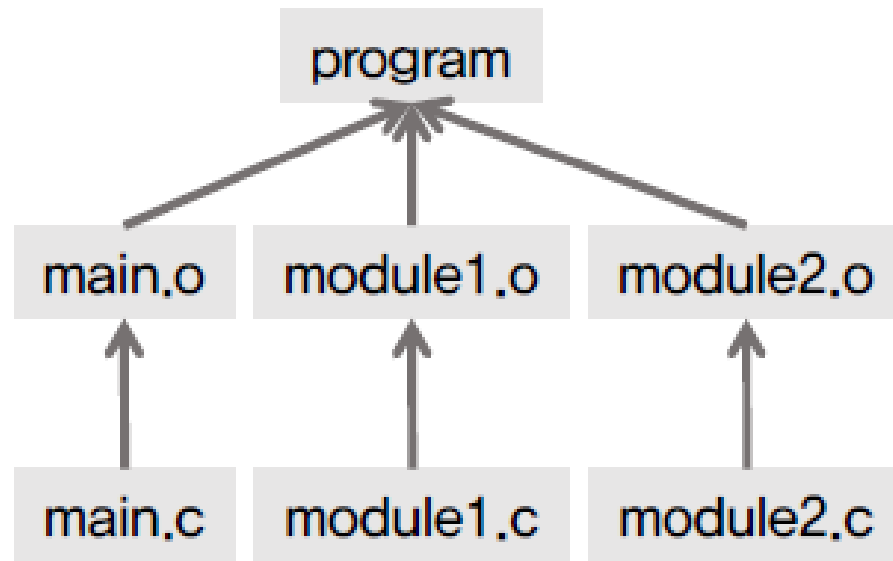
School of Computer Science and Engineering

Seoul National University

<http://bi.snu.ac.kr>

Makefiles

- It helps build a software from its source files, a way to organize code, and its compilation and linking.



Makefiles

- We are not going to use makefiles thoroughly, but just use it simply.
- Check the example project uploaded on our homepage.
(<http://bi.snu.ac.kr>)
- Suppose we have source files like below.
 - main.c move.c move.h common.c common.h
- Then compilation can be done in a single line of the command.
 - `gcc -o run main.c move.c move.h common.c common.h`
 - name of the execution file would be 'run'.

Makefiles

- The makefile in the example project.

```
all: main.c move.c move.h common.c common.h  
    gcc -o run main.c move.c move.h common.c common.h  
  
...
```

```
<target>: <dependencies>  
    <command>  
  
...
```

Makefiles

```
ta@pp:~/proj$ ls
common.c  common.h  in.txt  main.c  makefile  move.c  move.h
ta@pp:~/proj$
ta@pp:~/proj$
ta@pp:~/proj$ make
gcc -o run main.c move.c move.h common.c common.h
ta@pp:~/proj$
ta@pp:~/proj$
ta@pp:~/proj$
ta@pp:~/proj$ ls
common.c  common.h  in.txt  main.c  makefile  move.c  move.h  run
ta@pp:~/proj$
ta@pp:~/proj$
ta@pp:~/proj$
ta@pp:~/proj$ ./run < in.txt
```

Project Guideline 04

- Organizing the codes into multiple source files is recommended.
- The example project separated the functions into two categories.
 - common : overall definition of the structures, constants, and general functions
 - move : functions about basic movement and placements
- In future, we may have some additional functions about strategies, adding, for example, another source files like 'strategy.c', 'strategy.h'.

Project Guideline 04

- Generally, the constants, type definitions, and function prototypes go into the header file, and actual function definitions stay in the c file.
- The duplicate inclusion of a single header file can be prevented by using macros `#ifndef`, `#define`, `#endif`.

```
#ifndef _MY_H_  
#define _MY_H_  
  
...  
  
#endif
```

Assignment Submission

- Create a directory named **assignment** in your home directory.
- Create a directory named **10** in your **assignment** directory.
- Put your source files and a single **makefile**.
- An execution file named **run** should be created when **make** is done.
- Due to : **5/15(Fri) 23:59 pm**

Assignment [10 points]

- The inputs come from the project.
- Output all possible movements and placements.
- Use a makefile.

[Input]

loo
Heo
oGo
oLo
GE
c

[Output]

H A2 B2
H A2 B1
H A2 A1
H A2 A3
G B3 C3
G B3 A3
G B3 B2
L B4 C4
L B4 A4

G P B1
G P C1
G P C2
G P A3
G P C3
G P A4
G P C4
E P B1
E P C1
E P C2
E P A3
E P C3
E P A4
E P C4

Caution

- The example project does not consider the 'No Suicide' rule.
- You must strictly follow the all rules in the specification.

[Input]

ole
Cgo
oLo
ooo
GE
c

[Output]

L B3 C4
L B3 A4
L B3 C3
L B3 A3
L B3 B4

Caution

- If there's no available movement or placement, then output give_up.

[Input]

ooo

col

goe

Loo

GE

c

[Output]

give_up