



[GNU Make] Makefile 教學：使用條件編譯建立



開源教學 (<https://opensourcedoc.com/>)
/ [Makefile \(https://opensourcedoc.com/gnu-make/\)](https://opensourcedoc.com/gnu-make/)
/ [conditional compilation/#main-content](#)
/ 使用條件編譯建立靈活的 Makefile
[回首頁 \(/\)](#)



018

參數化，通用性改善一些，但仍然有一些小缺失，
，每次要換編譯器時都要修改檔案；另外，
是我們想在編譯函式庫時想加入 `-g` 參數，也是要
開設定檔來修改。我們希望 **Makefile** 保留足夠的彈性，不用時時修改 **Makefile**
就能夠符合我們的需求。

我們將前文的 **Makefile** 加入條件編譯及一些小的語法特性後修改如下：

贊助商連結



新申辦刷卡禮
最高\$8,000









謹慎理財

贊助商連結



新申辦刷卡禮
最高\$8,000









謹慎理財



```

ifndef CC
    CC=gcc
endif

ifndef CFLAGS
    ifeq ($(TARGET),Debug)
        CFLAGS=-Wall -Wextra -O2 -std=c99
    endif # TARGET
endif # CFLAGS

OBJ=$(dir $(main-content))
OBJS=test_deque_int.o deque_int.o
TEST_PROG=test_deque_int.out

test clean

-c deque_int.c
-d -o libalgotest.so deque_int.o

static: deque_int.o
    $(AR) rcs -o libalgotest.a deque_int.o

test: $(TEST_PROG)
    ./$(TEST_PROG)
    echo $$?

$(TEST_PROG): $(OBJS)
    $(CC) $(CFLAGS) -o $(TEST_PROG) $(OBJS)

```

```
# Pattern rules.
%.o: %.c
    $(CC) $(CFLAGS) -c $<

clean:
    $(RM) $(TEST_PROG) *.o *.so *.a
```

在這個專案中，除了原先的使用方式外，新增的使用方式如下文所述。

在命令列中修改 *CC* 參數時，可指定所用的 C 編譯器：

```
$ make CC=clang static
```

在本指令中，我們用 Clang 製作靜態函式庫。

在命令列中指定 *TARGET* 為 *Debug* 時，會加入 *-g* 參數：

```
$ make CC=gcc-4.9 TARGET=Debug
```

在本指令中，我們用 GCC-4.9 製作帶有除錯訊息的動態函式庫。

贊助商連結

贊助商連結





```
$ make CC=clang CFLAGS="-Wall -Wextra -g -Werror" test
```

在本指令中，我們使用 **Clana** 編譯後執行測試程式，並採用更嚴格的參數 (-



接著，我們來看實作的部分。首先，來看 **CC** 的設定：

[回頁頂 \(/gnu-make/conditional-](#)

```
## Clear the default value of CC.
CC=
ifndef CC
```

[回頁頂 \(/\)](#)



行內是註解，註解是給程式設計者閱讀的，這個部

為空值，因為 **CC** 預設為 **cc**，但我們不想使用這個預設值，故將其清空。接著，我們使用條件編譯在 **CC** 為空時將其設為 **gcc**，在類 **Unix** 系統上，這是一個合理的預設值。

如果我們在命令列加入 **CC** 的值，則 **make** 會按照該值所設定的 **C** 編譯器來編譯程式；反之，則使用 **gcc** 來編譯程式。在這樣的設置下，缺點在於使用者設定的環境變數會無效，因為在 **Makefile** 中清空了。

贊助商連結

如果想使用先前設置的環境變數，可參考以下指令：

贊助商連結



```
$ make CC=$CC

接著，來看 CFLAGS 的設定：

ifndef CFLAGS
  ifeq ($(TARGET),Debug)
    CFLAGS=-Wall -Wextra -g -std=c99
  else
    CFLAGS=-Wall -Wextra -O2 -std=c99
  endif # TARGET
endif # CFLAGS
```

在這裡，我們使用巢狀條件編譯。條件如下：

- 使用者在命令列設置 **CFLAGS** 則直接使用
- 如果使用者未自行設置 **CFLAGS** 時
 - 若使用者將 **TARGET** 設為 **Debug**，則使用適合於除錯的 **CFLAGS**，
 - 在其他情形，使用適合於發布的 **CFLAGS**

透過這樣的設置，我們就不會將 **CFLAGS** 寫死，較先前更靈活。





^

.PHONY: all dynamic static test clean

.PHONY的意思是說，該任務不代表某個檔案，所以一定會執行。基本上，非檔案名稱的任務都應設置此項目。如果刻意將某個檔案名稱設為 .PHONY則該任務會強迫執行。



接下來，我們有一項 pattern rule，先前沒有此項設置：

```
%.o: %.c
    compilation/#main-content)
    $(CC) $(CFLAGS) -c $<
回首頁 (/)
```

Pattern rule 算是 Makefile 中的通用規則；在本例中，代表每個 .o 檔案都對應



，\$< 指向某個來源檔案，在本例中即為某個 .c 檔案。此 Makefile 更通用，但也會較難閱讀。

，這個 Makefile 比先前來得靈活一些，我們不需要可以直接透過命令列更改 make 的行為。如果專案動態修正 make 的行為是更合理的使用方式。

分享本文



([https://www.facebook.com/sharer.php?](https://www.facebook.com/sharer.php?u=https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f)

[u=https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-](https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f)



([https://twitter.com/intent/tweet?url=https%3a%2f%2fopensource.doc.com%2fgnu-](https://twitter.com/intent/tweet?url=https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f&text=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e)

[make%2fconditional-](https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f&text=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e)

贊助商連結



(<https://www.linkedin.com/shareArticle?>

贊助商連結

[u=https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-](https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f&title=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e)

[compilation%2f&title=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%](https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f&title=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e)

[https://lineit.line.me/share/ui?url=https%3a%2f%2fopensource.doc.com%2fgnu-](https://lineit.line.me/share/ui?url=https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f&text=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e)

[ke%2fconditional-](https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f&text=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e)

[compilation%2f&text=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%](https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f&text=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e)

[https://web.skype.com/share?url=https%3a%2f%2fopensource.doc.com%2fgnu-](https://web.skype.com/share?url=https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f&text=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e)

[ke%2fconditional-](https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f&text=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e)

[compilation%2f&text=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%](https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f&text=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e)

[http://www.evernote.com/clip.action?](http://www.evernote.com/clip.action?url=https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f&title=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e)

[https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-](https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f&title=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e)

[compilation%2f&title=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%](https%3a%2f%2fopensource.doc.com%2fgnu-make%2fconditional-compilation%2f&title=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e)

[https://mail.google.com/mail/?](https://mail.google.com/mail/?ui=cm&to=&su=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e%2fconditional-compilation%2f)

[/ui=cm&to=&su=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%](https://mail.google.com/mail/?ui=cm&to=&su=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e%2fconditional-compilation%2f)

[%e%2fconditional-compilation%2f\) Y! \(http://compose.mail.yahoo.com/?](https://mail.google.com/mail/?ui=cm&to=&su=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e%2fconditional-compilation%2f)

[subject=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e%2fconditional-compilation%2f\)](http://compose.mail.yahoo.com/?ui=cm&to=&su=%5bGNU%20Make%5d%20Makefile%20%e6%95%99%e5%ad%b8%ef%bc%9a%e%2fconditional-compilation%2f)

[Y! \(mailto:?](mailto:)

謹慎理財

謹慎理財

新申辦副卡禮
最高\$8,000

[f](#)
 (<https://www.facebook.com/%E9%96%8B%E6%94%BE%E5%8E%9F%E5%A7%8B%E7%A2%BC%E6%8A%1822973814671118>)
 [f](#) (<https://www.facebook.com/groups/198788411522441/>)
 [t](#) (<https://twitter.com/openSourceDok>)

0 responses

回頁頂 (/gnu-make/conditional-compilation/#maincontent)

Surprised

Angry

Sad

[回首頁 \(/\)](#)

0則留言
開源技術教學網
登錄 ▾

Copyright (c) 2014-2022 Michelle Chen. All Rights Reserved.

贊助商連結



贊助商連結

