

C getopt usage

getopt 是用來判斷程式參數的函式。像是 ./a.out -s 之類的，getopt 能對後面的 -s 這類參數做判斷處理。

一開始要使用的話記得先：

```
#include <unistd.h>
```

getopt 有三個參數 (int argc, char* argv[], const char* optstring)，第一個和第二個就直接把 main 裡頭的參數丟進去就好了，第三個參數就是用來處理前面講到的 -s 的這類參數。

舉個例子來講，像是 optstring = "abf" 的話，就能夠抓到這三個參數值，如：

./a.out -a 或 ./a.out -b 或 ./a.out -abf 都 OK。

而如果你的 f 需要一個字串值，像是後面需要檔案名稱的話，就可以這樣設 abf:，f: 表示 f 之後需要有個額外的參數，在處理的時候可以透過 optarg 來使用這額外的參數。

getopt 通常會用 switch 做處理，以上面的例子當作架構

```
while((c=getopt(argc, argv, "abf:")) != -1)
{
    switch(c)
    {
        case 'a':
            break;
        case 'b':
            break;
        case 'f':
            puts(optarg);
            break;
        case ':':
            puts("oops");
            break;
        case '?':
            puts("wrong command");
            break;
    }
}
```

這裡可以透過檢查是否有 : 這個字元來判斷是否有漏掉額外的參數，以這裡來講，就是 f 後面接的檔名。

程式如果照下面這樣下的話，就會印出 "oops" 因為後面少了檔名：

```
./a.out -f
```

? 則代表這參數錯誤，不存在。

那如果有時候只想接檔名呢 如

```
./a.out text.txt
```

這樣的話 可以在上面的 while 迴圈之後用 argv[optind] 來抓 text.txt。

如果是複數的檔名，如：

```
./a.out text.txt text2.txt
```

也可以用 for 迴圈 optind++ 到 argc：

```
for(;optind<argc;optind++)
{
    puts(argv[optind]); //像是這樣
}
```

參考網址：

<http://pubs.opengroup.org/onlinepubs/009695399/functions/getopt.html>