

A taste of compiler optimization

姓名：張彥豪

日期：2017/9/21

Part.I Answer the following questions. You may check the manual page of cc or gcc (with “man cc” on unix systems or using google to search for the cc manual page).

1. What optimizations are included with the -O1 option?

在 O1 選項開啟的情況下，compiler 會試圖減少編譯完的 code size 及執行時間，但 compiler 並不會去做需要花很多編譯時間的優化。

2. What optimizations are included with the -O2 option?

在 O2 優化選項開啟的情況下，GCC compiler 會使用所有不會利用空間換取時間的優化手段；同時，O2 優化選項會增加編譯時間與編譯完的程式效能。

3. What optimizations are included with the -O3 option?

在 O3 優化選項開啟的情況下，GCC compiler 會使用所有的優化手段，這些手段包含以空間換取時間的優化手段，像是 loop unrolling，與 O2 優化選項相比，程式執行時間可能會較短。

Part II. Write a report on your experiment.

1. A brief introduction.

這次實驗的目的是要讓我們觀察在編譯過程中所加入的最佳化對於程式執行時間的影響。

2. Experimental environment.

I. Platform 1

- Machine : Desktop PC
- CPU : Intel Core i5-4460
- Clock rate : 3.2Ghz
- Memory size : 16GB 1600 MHz DDR3
- OS : Windows 7 Ultimate
- Compiler and version : GCC 6.3.0
- Benchmark : P10-1.c & P10-2.c

II. Platform 2

- Machine : ASUS UX-303 Notebook
- CPU : Intel Core i5-4210U
- Clock rate : 1.7Ghz
- Memory size : 4GB 1600 MHz DDR3
- OS : Ubuntu 16.04
- Compiler and version : GCC 5.4.0
- Benchmark : P10-1.c & P10-2.c

3. Experimental results.

- I. 從下表可以看到當執行沒有使用最佳化參數優化的程式執行時間會比有使用最佳化參數優化的程式花更多的時間。

II. Tables of execution time(seconds)

● Platform 1:

	Plain	-O1	-O2	-O3
P10-1	65.517	48.581	30.461	30.575
P10-1 (add -sort)	57.499	29.873	16.965	16.967
P10-2	79.792	36.361	35.029	34.554

● Platform 2:

	Plain	-O1	-O2	-O3
P10-1	82.806	34.874	29.567	31.307
P10-1 (add -sort)	66.572	17.097	14.796	14.398
P10-2	90.297	29.075	26.338	30.350

4. Conclusion

由上表的結果可以看出編譯器有無做最佳化對程式的執行時間有很大的影響，尤其是有最佳化跟無最佳化的程式相比，也就是有加最佳化參數與沒有加最佳化參數，執行時間的差距最為明顯，幾乎有快要一倍的差距。而 O1 與 O2 間的時間差距則是第二明顯的，可能原因應該是 O2 選項把 O1 有的優化手段且加

上剩下會增加編譯時間的優化手段全部用上了，讓程式的執行時間可以再縮短。

但到了 O3 程式的執行時間沒有明顯的下降，甚至有的程式執行時間還些許多過 O2 程式的執行時間，推測是因為 O3 使用了以空間換取時間的優化手段，導致編譯後的程式執行時間不減反增。

從上表也可以發現一個有趣的現象，在沒有加上任何優化參數的情況下，CPU 效能較差的平台 2 執行時間皆多過 CPU 效能較好的平台 1，但在加上優化參數後，發現平台 2 的執行時間大概跟平台 1 差不多，甚至還有比平台 1 更快的情況，目前我還沒有辦法解釋為什麼會發生這種情形，可能要等到課程進行到後面才能對這種情形有合理的解釋。