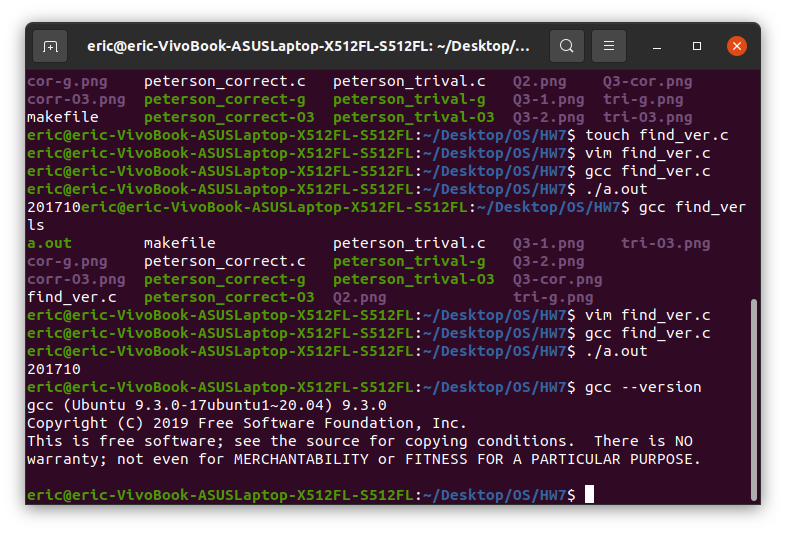
HW7

409510095 黃偉城

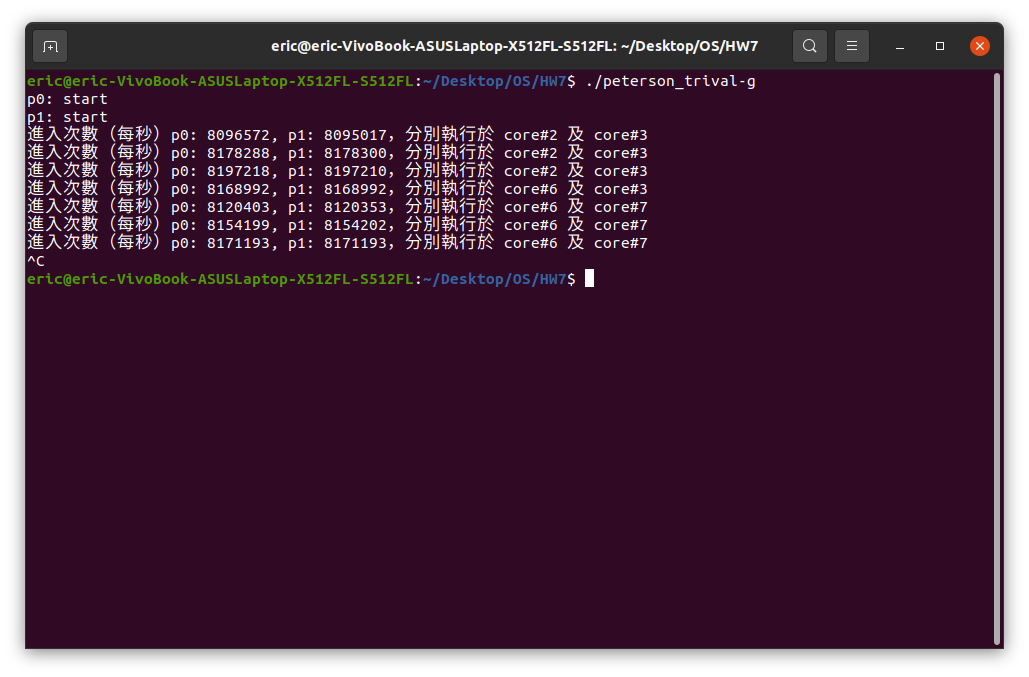
1. **說明你的執行環境（請自己想一下，哪些會和結果的正確語法有關，列出有關的執行環境就好）。**

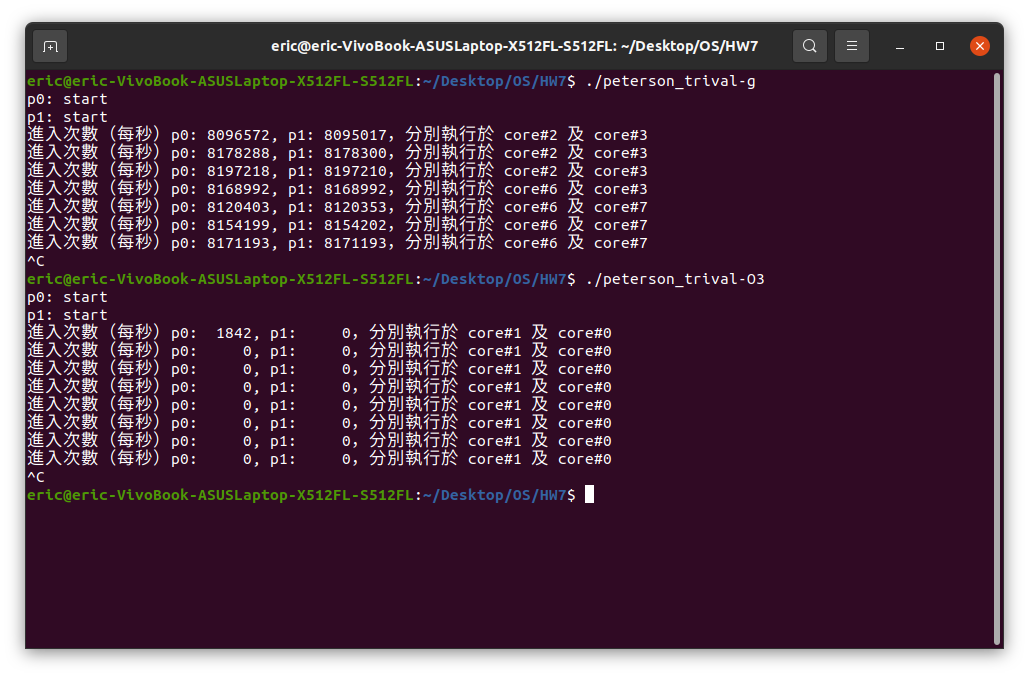


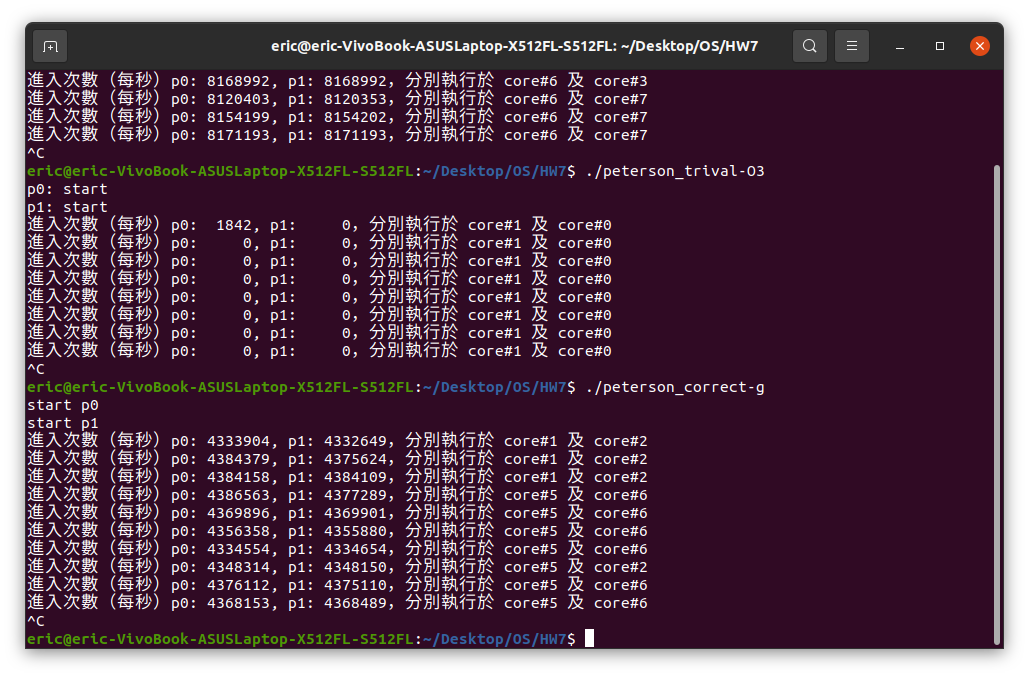
我的執行環境是ubuntu 20.04版本，GCC版本為9.3.0，另外我用\_\_STDC\_VERSION\_\_這個function列出目前C的版本，可以看到結果為201710，也就是C18的這個版本。

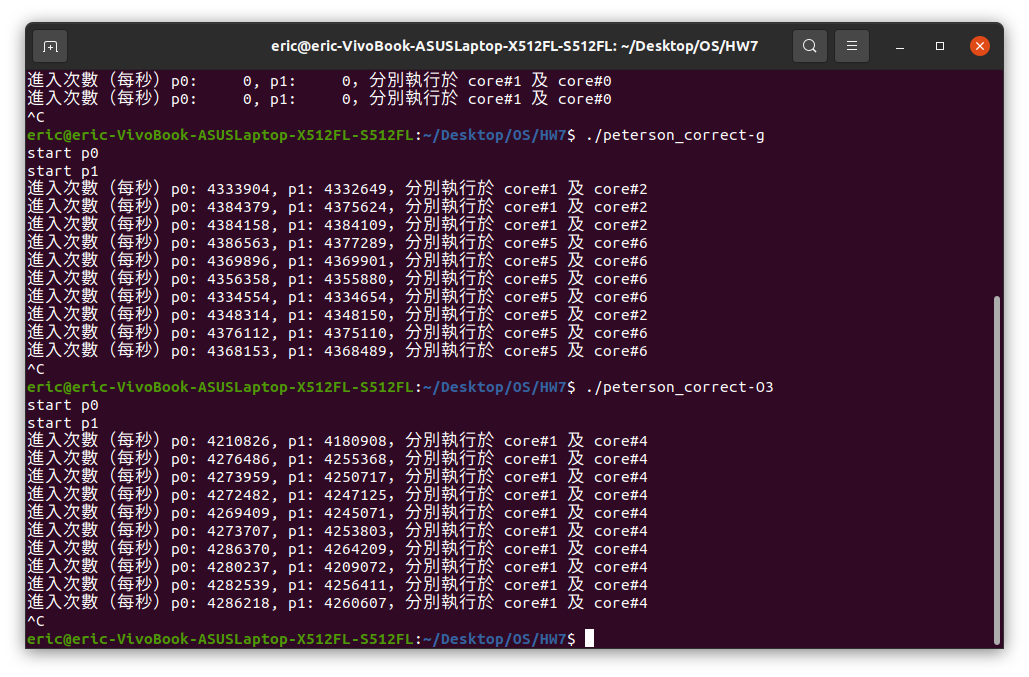
乙.

**1. 執行make，之後會產生四個執行檔案。請問你的執行節果為何？請附上畫面截圖**

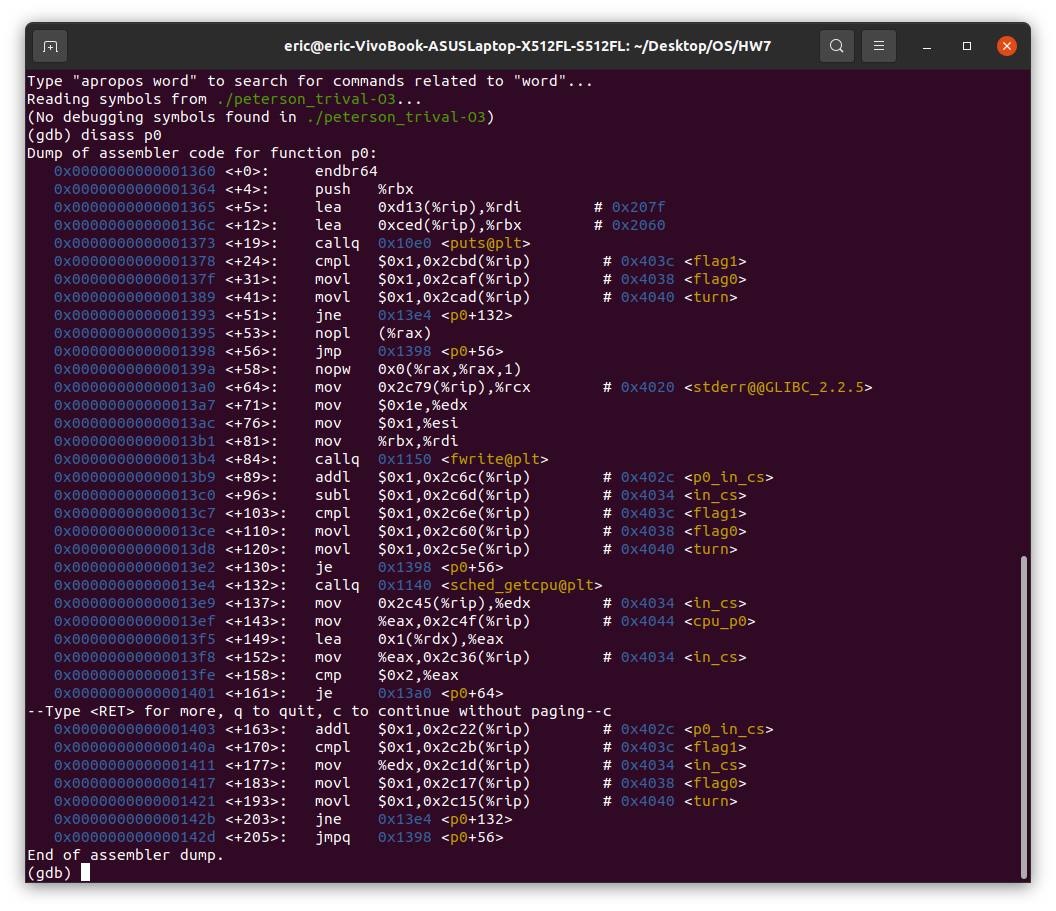








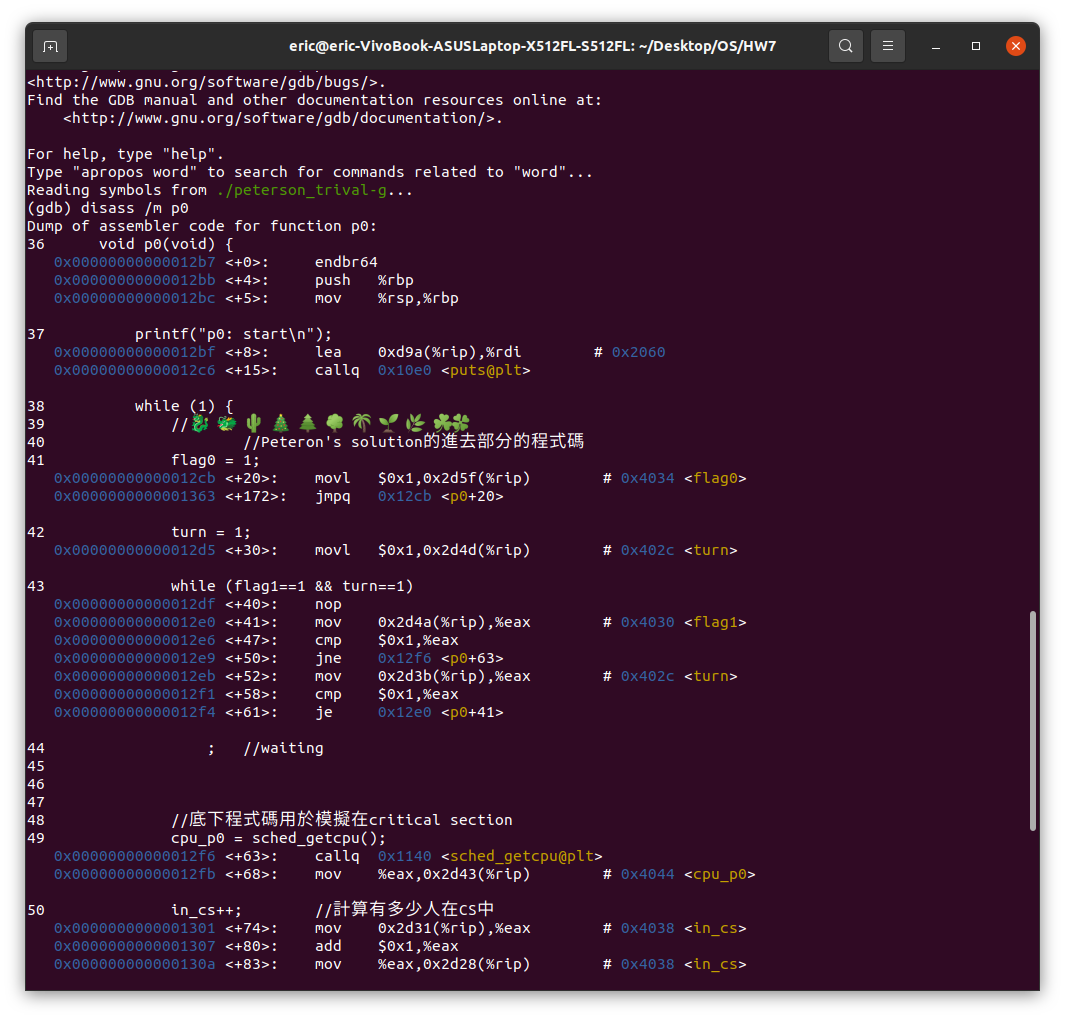
**2. 「確實的」解釋「為什麼」peterson\_trival-O3的執行結果是錯的**

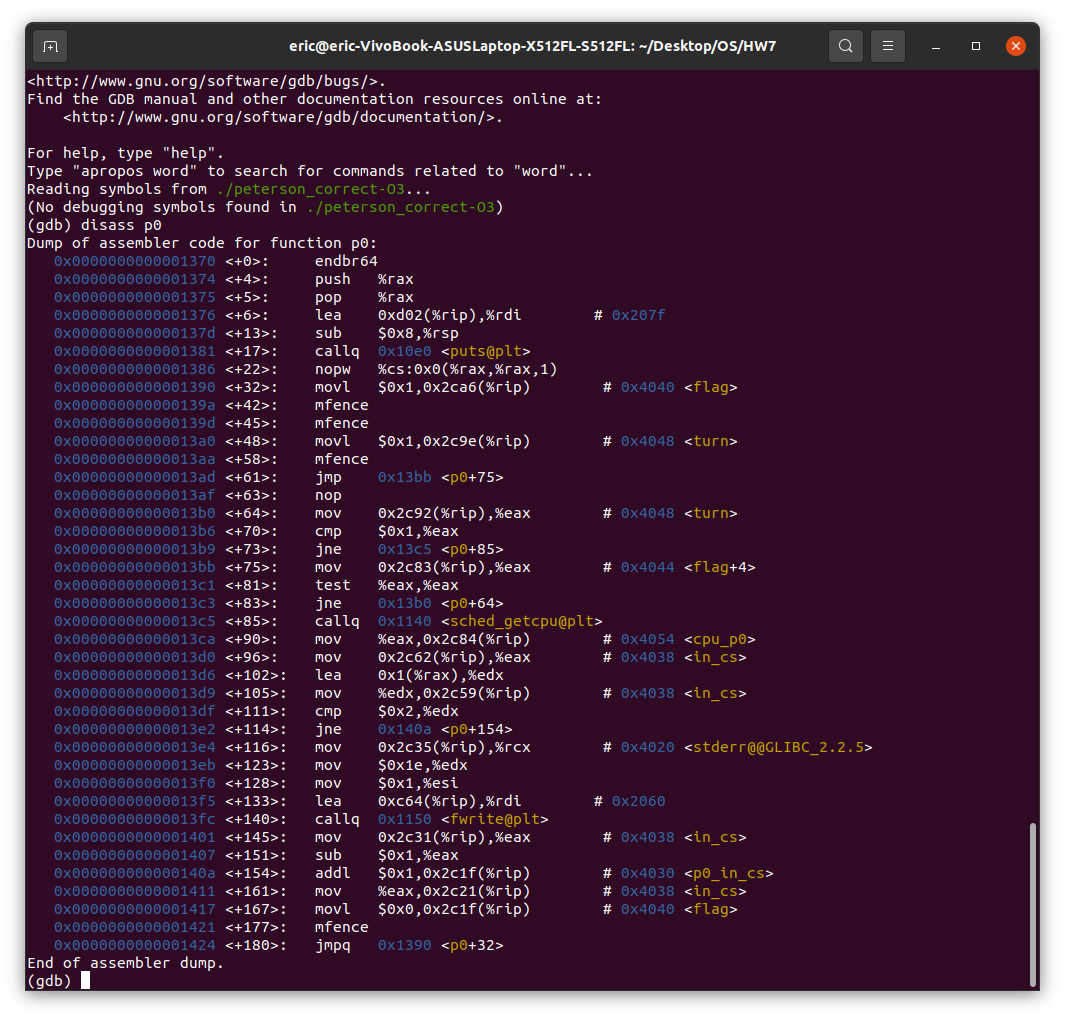


可以看到在24、31、41這三行上，本來應該要先將1移到flag0跟turn中，再去執行flag1的compare，但是copiler為了優化效能所以將compare移到前面，導致程式結果錯誤。

**3. 請問在你的電腦上「peterson\_trival-g」的速度比「peterson\_correct-O3」快或者是賣？上述二個程式的正確與否？**

在我的電腦上，peterson\_trival-g跑的速度比peterson\_correct-O3還要快





可以看到無論是peterson\_trival-g或是peterson\_correct-O3，他們在執行的時候沒有為了效能把順序調換，因此可以正確地執行判斷進入cs的任務

**4. 請「確實的」解釋「題三」，某個程式比另一個程式快或者慢的理由。**

peterson\_trival-g沒有保證mutual exclution，因此可能會有p0及p1同時進入的情況，雖然最終執行速度較快，但卻可能是錯的;相反的，peterson\_correct-O3雖然執行的慢，但他保證了mutual exclution，因此執行的結果會是對的。