

作業十四：

學習目標：

- 相較於 process，使用 thread 撰寫多核心、平行計算的程式不只能獲得比較好的執行效能，在程式設計上也因為共用所有記憶體，因此資料交換比較容易
- 在 multithreaded 應用程式中，如何存取「共享資料」是一個重要的議題
- Multithreaded 應用程式中，到底是怎樣處理 signal？

題目：

1. (40pt) 請撰寫應用程式，使用蒙特卡羅法計算 pi，應用程式可以有二個參數，第一個參數是總共打多少個點，第二個參數是使用多少 thread 做運算
2. (20pt) 當按下 ctrl-c 的時候，顯示截至目前計算出來的 pi 是多少
3. (20pt) 當在一秒內連續按下二次 ctrl-c 的時候，顯示截至目前計算出來的 pi 是多少，並結束程式
4. (20pt) 請說明你的應用程式比授課老師所給的範例程式快或者是慢，並『具體』說明變快或者變慢的原因

(<https://github.com/shiwulo/system-programming/blob/master/ch12/pi/pi.c>)

繳交：

1. 繳交原始程式碼及 makefile。助教執行 make，必須產生 pi
2. 繳交報告，在報告中說明 1~4 項中完成了哪些，並在第四項中說明你量測出的程式效能，並說明效能較好或壞的原因
3. 將原始程式碼、makefile、報告（pdf 形式）壓縮成.tar.bz2
4. 再次提醒，助教會將所有人的作業於 dropbox 上公開
5. 如果真的不會寫，記得去請教朋友。在你的報告上寫你請教了誰即可。
6. 繳交期限：2020/06/29 早上 8:00