

# 408410035 鄭 x 辰

甲:

```
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi
precision=7    pi=3.1415926475

real    0m3.625s
user    0m6.271s
sys     0m0.009s
```

Real time:真實經過的時間

User time:cpu 花在這個程式上的時間->全部核心所花的時間的加總

Sys time:花在系統指令的時間(像是建立 thread 跟印出結果)

乙:

```
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi 7 10
precision=7    pi=3.1415926424

real    0m1.213s
user    0m3.097s
sys     0m0.016s
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi 7 9
precision=7    pi=3.1415926430

real    0m1.260s
user    0m3.610s
sys     0m0.008s
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi 7 8
precision=7    pi=3.1415926438

real    0m2.982s
user    0m6.988s
sys     0m0.028s
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi 7 11
precision=7    pi=3.1415926420
```

```
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi 7 1
precision=7    pi=3.1415926499

real    0m4.645s
user    0m4.645s
sys     0m0.000s
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi 7 2
precision=7    pi=3.1415926464

real    0m2.912s
user    0m3.826s
sys     0m0.005s
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi 7 3
precision=7    pi=3.1415926475

real    0m3.717s
user    0m6.337s
sys     0m0.017s
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi 7 4
precision=7    pi=3.1415926447

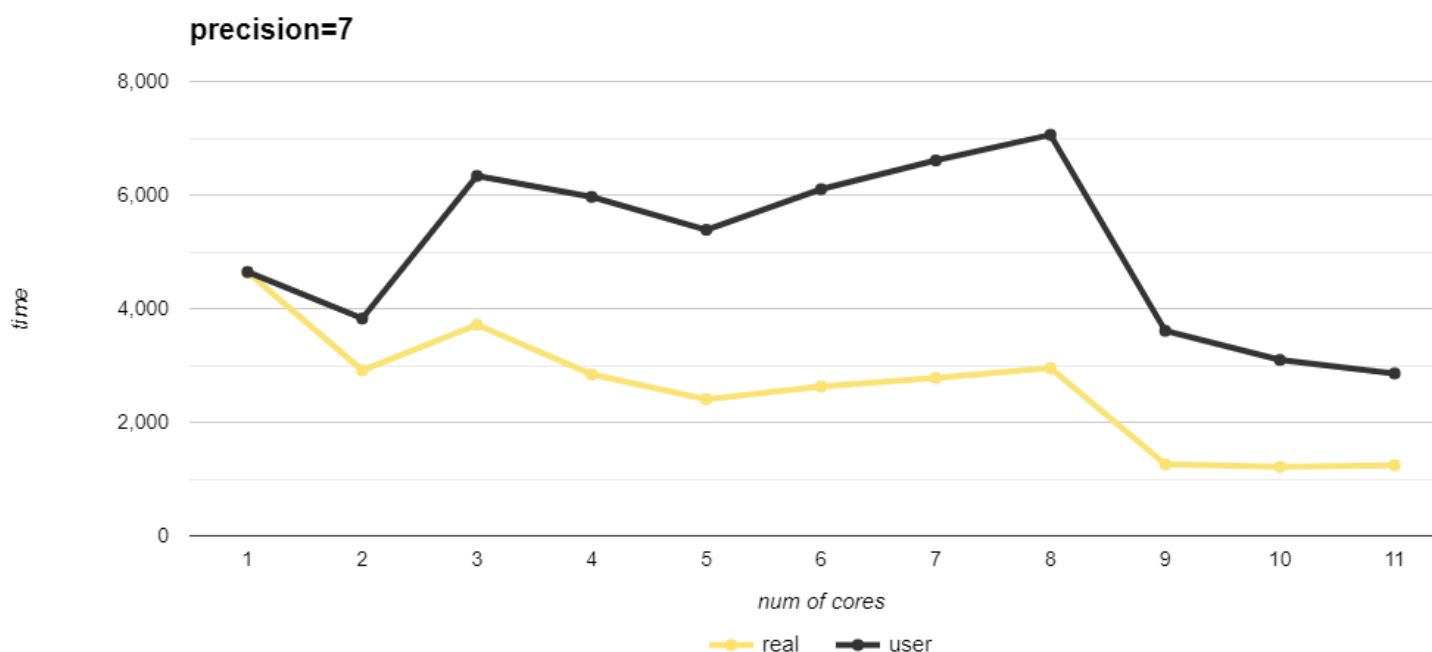
real    0m2.843s
user    0m5.968s
sys     0m0.025s
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi 7 5
precision=7    pi=3.1415926446

real    0m2.405s
user    0m5.388s
sys     0m0.013s
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi 7 6
precision=7    pi=3.1415926436

real    0m2.630s
user    0m6.103s
sys     0m0.000s
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi 7 7
precision=7    pi=3.1415926441

real    0m2.780s
user    0m6.609s
sys     0m0.005s
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi 7 8
precision=7    pi=3.1415926438

real    0m2.953s
user    0m7.061s
sys     0m0.012s
hsa1155@TKwaterHeater-VirtualBox:~/os/hw5$ time ./pi 7 10
```



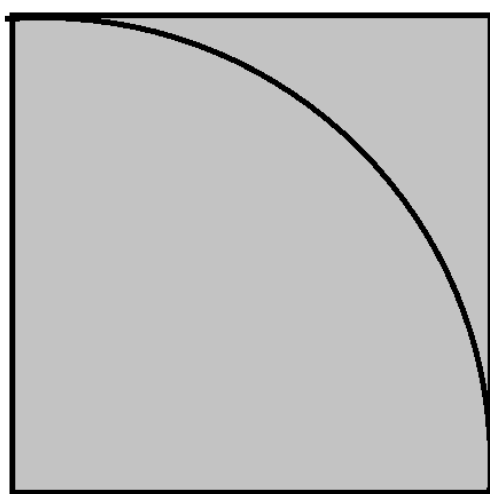
上圖中 `./pi x y` `x` 代表精確位數 `y` 代表核心數量 ( 預設為 7 跟函式回傳指定 )

從圖表中可以看到執行時間基本上隨著核心增加而減少, 但是不完全是線性的, 那可能是因為由於我對計算過程的調整導致每個 `c p u` 結束運算的時間不同, 但是保證每個 `c p u` 算完後回傳的答案都可以不會被彼此運算的速度互相影響

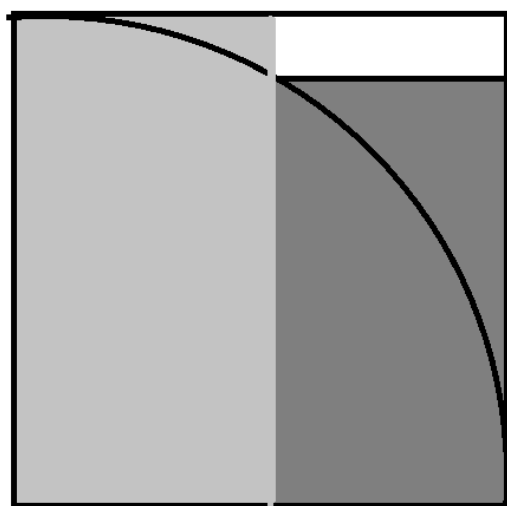
( 註：我虛擬機預設為三核心 )

丙：

在我的程式中，在增加精度時，我的程式會將原本上 / 下界中儲存的值除以二，之後再加上由於精度提升而增加的長方形的量，可以減少一半的運算次數

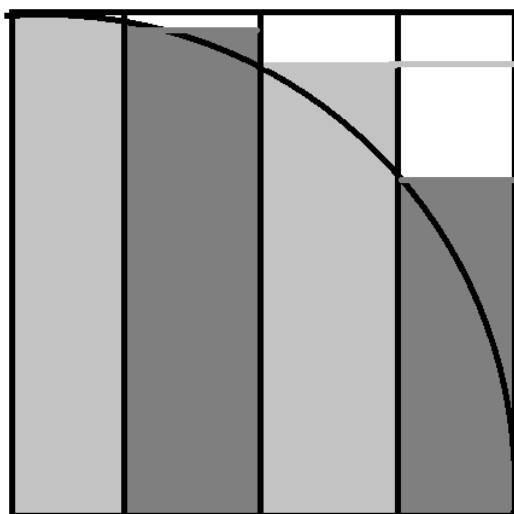


最開始時的上界



第一次逼近時的上界

淺灰色為上次運算的一半



第二次逼近時上界  
淺灰為上次結果的  
一半 深灰為新增  
加的區間