

SP Programming hw4 report

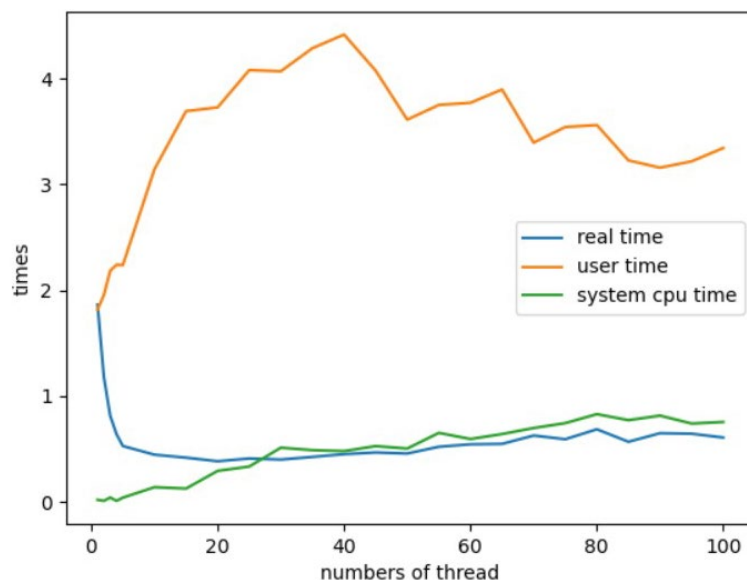
1. Thread 2 Thread 20

real: 0m1.166s

real: 0m0.428s

因為 Thread 20 相較於 Thread 2 同時有更多的 Thread 在同時計算，所以會更快的跑完整個 process。

2.



real time: 一開始的時候因為同時在跑的 Thread 顯著變多而急速下降，不過 Thread 比 20 還多後，我推測因為 CPU 沒有足夠的核心供那麼多的 Thread 同時計算，加上 system cpu time 的增加，而有緩慢的上升。

user time: 因為每個 Thread 都算在 user time 裡面，所以一開始隨著 Thread 數量的上升而變多，不過大概到 40 以後，就呈

現震盪狀態，我認為是因為這段時間後 CPU 沒有足夠的核心供這麼多 Thread 同時運作，震盪的原因則是被工作站系統當時的狀態影響。

system cpu time: 大致呈線性上升，我認為與我的 implement 有關，因為我是在每次 epoch 都 create 跟 join 一次，而 pthread_create() 跟 pthread_join() 都會吃到 system cpu，所以當 thread 數量越來越多的時候自然 system cpu 就跟著上升。

3.

```
for(int t=0;t<epoch;t++){
    if((pid=vfork())==0){
        for(int i=0;i<row;i++){
            for(int j=0;j<col/2;j++){
                cal(i,j);
            }
        }
        _exit(0);
    }
    else{
        for(int i=0;i<row;i++){
            for(int j=col/2;j<col;j++){
                cal(i,j);
            }
        }
        char **tmp = src;
        src = des;
        des = tmp;
        wait();
    }
}
```

我把 board 切成兩半分給 parent 跟 child 做，因為 vfork 本身就 share memory，所以不需要特別處理這塊。