OS PROJECT 1 REPORT

B05902002 資工四 李栢淵

1. 核心版本

- 4.14.25
- 使用雙核心

2. 設計

- 設計兩個 system call
  - `void sys_get_time(long *sec, long *nsec)`
    - 透過 `getnstimeofday` 拿到現在的秒數以及奈秒數。
  - `void sys_log_info(char* message)`
    - 透過 `pinrtk` 印 message 到 kernel 的 buffer。
- 使用 process 的第一次拿到 cpu 的時間當成 start time
- 使用 `sched_setaffinity` 來讓 scheduling 使用的 CPU 都固定在第一顆，將其他 process 的執行固定在第二顆。
- 使用 `sched_setscheduler` 來調整 process 的 priority。
- 四種 scheduling plicy
  - FIFO
    - 將所有 process 依據 ready time sort 過，然後等到前一個人結束再丟出來。
  - RR
    - Implement 一個 ready queue，將ready time 到了的 process 放進 queue，等到週期（500 個 timestamp）到了，再從 ready queue 拿一個出來，把自己 push 進 queue，如果找不到，自己就繼續做。
  - SJF & preemptive SJF
    - 對於 SJF，在開頭的部分判斷一下 current process 是否存在，如果有，就不能打斷。
    - 如果沒有，那剩下的演算法跟 preemptive SJF 一模一樣，從目前的 active process 挑一個最短的process 來執行。

3. 實驗結論

Calculate unit time

首先透過 T_MEASUREMENT 計算單位時間，根據 `TIME_MEASUREMENT_dmesg.txt` 的結果：

```
[Project1] 10600 1588162600.763452504 1588162601.410629676
[Project1] 10603 1588162602.038254775 1588162602.701593983
[Project1] 10604 1588162603.461285069 1588162604.169390028
[Project1] 10605 1588162604.849552828 1588162605.525174666
[Project1] 10606 1588162606.206607289 1588162606.886194620
[Project1] 10607 1588162607.552670351 1588162608.273273153
[Project1] 10608 1588162608.982802313 1588162609.672220888
[Project1] 10609 1588162610.374502603 1588162611.085916845
[Project1] 10610 1588162611.764333871 1588162612.419808220
[Project1] 10611 1588162613.117757842 1588162613.785345517
```

所有 process 的 執行時間依序為：

```
[647177172, 663339208, 708104959, 675621838, 679587331,
 720602802, 689418575, 711414242, 655474349, 667587675]
```

平均為 `681832815.1 (ns)` ，再除以500為 `1363665.6302 (ns)` 。

可以發現，這筆 test case 算是簡單的，可以假設其環境、cpu的狀態應該都要是差不多的，但是這些時間段裡，最小值647177172 ns，最大值 720602802 ns，幾乎快差了 0.1 秒，代表 user space 下的 scheduling 在計算時間上有不小的誤差。

以上的 python file 為 `test/get_t_unit.py` ，以下則會根據每一種 test case 的理論時間與實驗時間進行分析。

Conclusion

實驗的結果放在最下方的 Attachment，以下僅寫出結論。

1. FIFO test case conclusion

基本上FIFO不太會有 scheduling 的 cost，大部分的 process 間串接的時間都沒有差（前一個 end time 等於後一個 start time）。但總結果跟理論結果有一些出入，原因可能是 unit time 在每次的執行可能會差很多。舉例來說，以 FIFO_2 的unit time 應該要是我們計算出來的 98.017% 左右才會符合數據，而調整後，將會貼近理想結果。

	Experiment result	Adjusted experiment result	Theoritical result
P1	0 - 78414	0 - 80000	0 - 80000
P2	78414 - 83396	80000 - 85083	80000 - 85000
P3	83396 - 84395	85083 - 86102	85000 - 86000
P4	84395 - 85387	86102 - 87114	86000 - 87000

我後來有跑了三四次，發現 unit time 的時間範圍大概是 1280930.9722 ~ 1363665.6302，基本上還差蠻多的，就會造成一些基本的誤差。

2. RR test case conclusion

原先使用了 `i % n` 的方式去找 next process，後來改成用 queue 來 enqueue 跟 dequeue，這樣下來真的進步蠻多的，context switch 的時間有觀察到明顯的減少。

3. SJF test case conclusion

雖然從結果看不出來，但 SJF 我沒有用比較好的資料結構找最小的執行時間，因此 process 承上啟下間的時間比 FIFO 多出許多，多了大概十倍，但其實由於 testing data 的數量都比較少的關係，所以其實也只有差幾毫秒，但如果 process 數量上升，那誤差應該就會蠻明顯的。

但基本上除了 unit time 的問題，沒有明顯的誤差。

4. PSJF test case conclusion

跟上面的狀況差不多，又因為是 preemptive 的關係，使得每一個 time stamp 都會去找下一個 exec time 最小的 process，從我自己印出的 log 可以發現花了蠻多時間在 find next process ，可能實作上要盡可能減少這種 `O(n)` 搜尋。

4. ATTACHMENT: 實驗結果

- Experiment result
  - 記錄實驗的數據，將全部的時間轉換為相對的 time stsamps（取最早開始的 process 的 ready time 對應到其理論起始 time stamp）。
  - 對應的 python file 為 `test/log_experiment_result.py`
- Theoritical result
  - 我另外寫了一些 python file，再跑一次原先的 scheduling algroithm，得到參考用的結果，檔案為 `test/log_ground_truth.py` 。
- 註：開始的 time stamp 為 process 的第一次拿到 cpu 的時間。

FIFO_1.txt

Experiment result

```
P1 -> start from 0 to 489
P2 -> start from 489 to 983
P3 -> start from 983 to 1463
P4 -> start from 1463 to 1932
P5 -> start from 1932 to 2402
```

Theoritical result

```
P1 -> start from 0 to 500
P2 -> start from 500 to 1000
P3 -> start from 1000 to 1500
P4 -> start from 1500 to 2000
P5 -> start from 2000 to 2500
```

FIFO_2.txt

Experiment result

```
P1 -> start from 0 to 78414
P2 -> start from 78414 to 83396
P3 -> start from 83396 to 84395
P4 -> start from 84395 to 85387
```

Theoritical result

```
P1 -> start from 0 to 80000
P2 -> start from 80000 to 85000
P3 -> start from 85000 to 86000
P4 -> start from 86000 to 87000
```

FIFO_3.txt

Experiment result

```
P1 -> start from 0 to 8095
P2 -> start from 8095 to 13120
P3 -> start from 13120 to 16138
P4 -> start from 16139 to 17155
P5 -> start from 17155 to 18169
P6 -> start from 18170 to 19179
P7 -> start from 19180 to 23174
```

Theoritical result

```
P1 -> start from 0 to 8000
P2 -> start from 8000 to 13000
P3 -> start from 13000 to 16000
P4 -> start from 16000 to 17000
P5 -> start from 17000 to 18000
P6 -> start from 18000 to 19000
P7 -> start from 19000 to 23000
```

FIFO_4.txt

Experiment result

```
P1 -> start from 0 to 2036
P2 -> start from 2036 to 2538
P3 -> start from 2538 to 2739
P4 -> start from 2739 to 3237
```

Theoritical result

```
P1 -> start from 0 to 2000
P2 -> start from 2000 to 2500
P3 -> start from 2500 to 2700
P4 -> start from 2700 to 3200
```

FIFO_5.txt

Experiment result

```
P1 -> start from 0 to 8140
P2 -> start from 8140 to 13148
P3 -> start from 13148 to 16199
P4 -> start from 16199 to 17186
P5 -> start from 17186 to 18172
P6 -> start from 18172 to 19177
P7 -> start from 19177 to 23163
```

Theoritical result

```
P1 -> start from 0 to 8000
P2 -> start from 8000 to 13000
P3 -> start from 13000 to 16000
P4 -> start from 16000 to 17000
P5 -> start from 17000 to 18000
P6 -> start from 18000 to 19000
P7 -> start from 19000 to 23000
```

RR_1.txt

Experiment result

```
P1 -> start from 0 to 500
P2 -> start from 500 to 988
P3 -> start from 988 to 1491
P4 -> start from 1491 to 1985
P5 -> start from 1985 to 2481
```

Theoritical result

```
P1 -> start from 0 to 500
P2 -> start from 500 to 1000
P3 -> start from 1000 to 1500
P4 -> start from 1500 to 2000
P5 -> start from 2000 to 2500
```

RR_2.txt

Experiment result

```
P1 -> start from 600 to 7855
P2 -> start from 1080 to 9320
```

Theoritical result

```
P1 -> start from 600 to 8100
P2 -> start from 1100 to 9600
```

RR_3.txt

Experiment result

```
P1 -> start from 1200 to 20175
P2 -> start from 2373 to 19371
P3 -> start from 4276 to 18341
P4 -> start from 5681 to 30722
P5 -> start from 6619 to 29756
P6 -> start from 7557 to 27705
```

Theoritical result

```
P1 -> start from 1200 to 20700
P2 -> start from 2400 to 19900
P3 -> start from 4400 to 18900
P4 -> start from 5900 to 31200
P5 -> start from 6900 to 30200
P6 -> start from 7900 to 28200
```

RR_4.txt

Experiment result

```
P1 -> start from 0 to 23105
P2 -> start from 516 to 20111
P3 -> start from 1029 to 14560
P4 -> start from 1537 to 5556
P5 -> start from 2049 to 6051
P6 -> start from 2548 to 6560
P7 -> start from 3542 to 18559
```

Theoritical result

```
P1 -> start from 0 to 23000
P2 -> start from 500 to 20000
P3 -> start from 1000 to 14500
P4 -> start from 1500 to 5500
P5 -> start from 2000 to 6000
P6 -> start from 2500 to 6500
P7 -> start from 3500 to 18500
```

RR_5.txt

Experiment result

```
P1 -> start from 0 to 22972
P2 -> start from 484 to 20017
P3 -> start from 977 to 14563
P4 -> start from 1465 to 5500
P5 -> start from 1960 to 6026
P6 -> start from 2967 to 7031
P7 -> start from 3471 to 18562
```

Theoritical result

```
P1 -> start from 0 to 23000
P2 -> start from 500 to 20000
P3 -> start from 1000 to 14500
P4 -> start from 1500 to 5500
P5 -> start from 2000 to 6000
P6 -> start from 3000 to 7000
P7 -> start from 3500 to 18500
```

SJF_1.txt

Experiment result

```
P1 -> start from 6967 to 13971
P2 -> start from 0 to 2015
P3 -> start from 2015 to 3012
P4 -> start from 3012 to 6967
```

Theoritical result

```
P1 -> start from 7000 to 14000
P2 -> start from 0 to 2000
P3 -> start from 2000 to 3000
P4 -> start from 3000 to 7000
```

SJF_2.txt

Experiment result

```
P1 -> start from 100 to 201
P2 -> start from 421 to 4453
P3 -> start from 202 to 421
P4 -> start from 4453 to 8440
P5 -> start from 8440 to 15457
```

Theoritical result

```
P1 -> start from 100 to 200
P2 -> start from 400 to 4400
P3 -> start from 200 to 400
P4 -> start from 4400 to 8400
P5 -> start from 8400 to 15400
```

SJF_3.txt

Experiment result

```
P1 -> start from 100 to 3076
P2 -> start from 11135 to 16011
P3 -> start from 16011 to 22605
P4 -> start from 3076 to 3088
P5 -> start from 3088 to 3110
P6 -> start from 3110 to 7067
P7 -> start from 7067 to 11135
P8 -> start from 22605 to 30912
```

Theoritical result

```
P1 -> start from 100 to 3100
P2 -> start from 11120 to 16120
P3 -> start from 16120 to 23120
P4 -> start from 3100 to 3110
P5 -> start from 3110 to 3120
P6 -> start from 3120 to 7120
P7 -> start from 7120 to 11120
P8 -> start from 23120 to 32120
```

SJF_4.txt

Experiment result

```
P1 -> start from 0 to 2792
P2 -> start from 2792 to 3728
P3 -> start from 3728 to 7439
P4 -> start from 8376 to 10228
P5 -> start from 7439 to 8376
```

Theoritical result

```
P1 -> start from 0 to 3000
P2 -> start from 3000 to 4000
P3 -> start from 4000 to 8000
P4 -> start from 9000 to 11000
P5 -> start from 8000 to 9000
```

SJF_5.txt

Experiment result

```
P1 -> start from 0 to 1897
P2 -> start from 1897 to 2380
P3 -> start from 2380 to 2856
P4 -> start from 2856 to 3325
```

Theoritical result

```
P1 -> start from 0 to 2000
P2 -> start from 2000 to 2500
P3 -> start from 2500 to 3000
P4 -> start from 3000 to 3500
```

PSJF_1.txt

Experiment result

```
P1 -> start from 0 to 24611
P2 -> start from 984 to 15692
P3 -> start from 1971 to 9789
P4 -> start from 2944 to 5849
```

Theoritical result

```
P1 -> start from 0 to 25000
P2 -> start from 1000 to 16000
P3 -> start from 2000 to 10000
P4 -> start from 3000 to 6000
```

PSJF_2.txt

Experiment result

```
P1 -> start from 0 to 3996
P2 -> start from 987 to 1992
P3 -> start from 3996 to 11041
P4 -> start from 5016 to 7046
P5 -> start from 7046 to 8011
```

Theoritical result

```
P1 -> start from 0 to 4000
P2 -> start from 1000 to 2000
P3 -> start from 4000 to 11000
P4 -> start from 5000 to 7000
P5 -> start from 7000 to 8000
```

PSJF_3.txt

Experiment result

```
P1 -> start from 0 to 3519
P2 -> start from 544 to 1032
P3 -> start from 1032 to 1533
P4 -> start from 1534 to 2034
```

Theoritical result

```
P1 -> start from 0 to 3500
P2 -> start from 500 to 1000
P3 -> start from 1000 to 1500
P4 -> start from 1500 to 2000
```

PSJF_4.txt

Experiment result

```
P1 -> start from 6642 to 13578
P2 -> start from 0 to 2854
P3 -> start from 100 to 1079
P4 -> start from 2854 to 6642
```

Theoritical result

```
P1 -> start from 7000 to 14000
P2 -> start from 0 to 3000
P3 -> start from 100 to 1100
P4 -> start from 3000 to 7000
```

PSJF_5.txt

Experiment result

```
P1 -> start from 100 to 202
P2 -> start from 408 to 4258
P3 -> start from 202 to 407
P4 -> start from 4258 to 8044
P5 -> start from 8044 to 14910
```

Theoritical result

```
P1 -> start from 100 to 200
P2 -> start from 400 to 4400
P3 -> start from 200 to 400
P4 -> start from 4400 to 8400
P5 -> start from 8400 to 15400
```