# OS Project 1 Report

Group 13

## 1. 設計：

### 整體架構設計：

利用 sched 中的 FIFO 模式實現 real time 排程，藉由調整不同 process 的優先度來決定現在該執行哪個 process。scheduler 和 process 都在同一顆 CPU 上運行。

Parent process 的 priority 為 2。

API:

make(TSK *tsk):

fork 出一個 process，由於 priority 為 1，所以還不會跑，並將 tsk -> pid 被設成該 process 的 pid，在 task ready 時呼叫。

run(TSK *tsk, int run_time):

讓 tsk -> pid 的 process 的 priority 上升為 3 後，跑 run_time 單位時間，結束後其 priority 重新設為 1。

clear(TSK *tsk):

收屍。

### FIFO/SJF/PSJF 設計方法:

在整個執行過程中，會有三種動作，(1)wait,(2)run,(3)make。因此在各個演算法中，只要將可以 input 轉化為指令，指令包含了是屬於哪一個動作、以及要執行多久、對哪一個 pid 執行。預先將這些都處理好，如此一來最後只要將每一個指令依序執行，即可達成指定的成果。以下再分別討論三個的細節:

**FIFO :**

先照 arrive time 由小到大排好，看下一個 task 要 wait 多久，以及是否需要 make(可能之前在跑其他 task 時已經先 make 了)。然後去看

在跑這個 task 時，是否有其他 task arrive，有的話就必須中斷 run，去make。譬如說

```
FIFO
p1 0 5
p2 2 5
```

就會是
make(p1)
run(p1, 2)
make(p2)
run(p1, 3)
run(p2,5)

**SJF：**

大致與 FIFO 一樣，只是多了需要去記錄目前有幾個 task 是已經arrive，但尚未執行的。對這些 task 已執行時間長短去 sort，比較小的即是下一個要執行的 task。

**PSJF：**

先照 arrival time 及 task time 由小排到大，再開一個約略 4000 的陣列去去預處理此刻應該要執行哪一個指令。在下一個 task arrive前，都是用 SJF 來決定哪一個 process 先跑，如果先行能跑的 process都跑完了，就會記錄要空等；如果超過下一個 task arrival time 就先把正在跑的 process 停止，然後再去檢查以 SJF 哪一個 process 先跑

## RR 設計方法：

利用雙指標維護 queue(fifo)，每次將頭取出並檢查這段時間(500 單位時間)內有無其他 task 開始，若有則將其排入 queue，結束後若還有剩餘所需執行時間則再將其排入 queue。

## 2. 執行範例結果：

(以下 Output 中的紅色時間是代表總過程的頭尾)

input:

```
FIFO
10
P0 0 500
P1 1000 500
P2 2000 500
P3 3000 500
P4 4000 500
P5 5000 500
P6 6000 500
P7 7000 500
P8 8000 500
P9 9000 500
```

Output:

```
[Project1] 25090 1556535551.998336599, 1556535552.671563721
[Project1] 25091 1556535553.332428659, 1556535553.989236285
[Project1] 25092 1556535554.654427455, 1556535555.317614491
[Project1] 25093 1556535555.980587039, 1556535556.641470316
[Project1] 25094 1556535557.309442976, 1556535557.969548889
[Project1] 25095 1556535558.631236361, 1556535559.288489485
[Project1] 25096 1556535559.953183164, 1556535560.609638149
[Project1] 25097 1556535561.278671390, 1556535561.940674030
[Project1] 25098 1556535562.604337024, 1556535563.261824970
[Project1] 25099 1556535563.924977916, 1556535564.591205631
```

```
P0 25090   -1556535551.998336599 + 1556535552.671563721 = 0.673
P1 25091   -1556535553.332428659 + 1556535553.989236285 = 0.657
P2 25092   -1556535554.654427455 + 1556535555.317614491 = 0.663
P3 25093   -1556535555.980587039 + 1556535556.641470316 = 0.661
P4 25094   -1556535557.309442976 + 1556535557.969548889 = 0.660
P5 25095   -1556535558.631236361 + 1556535559.288489485 = 0.657
P6 25096   -1556535559.953183164 + 1556535560.609638149 = 0.656
P7 25097   -1556535561.278671390 + 1556535561.940674030 = 0.662
P8 25098   -1556535562.604337024 + 1556535563.261824970 = 0.658
P9 25099   -1556535563.924977916 + 1556535564.591205631 = 0.666
```

Average of ten:
(0.673+0.657+0.663+0.661+0.660+0.657+0.656+ 0.662+0.658+ 0.666)/10
= 0.661

500 units 跑 0.661 秒 => 1000 units 跑 1.322 秒

# FIFO:

## FIFO_1:

```
FIFO
5
P1 0 500
P2 0 500
P3 0 500
P4 0 500
P5 0 500
```

理論總時間：(500+500+500+500+500)/1000 * 1.322 = 3.305

Output:

```
[Project1] 24363 1556534522.138770991, 1556534522.829885256
[Project1] 24364 1556534522.830428149, 1556534523.499013848
[Project1] 24365 1556534523.499561899, 1556534524.172957258
[Project1] 24366 1556534524.173507756, 1556534524.847395712
[Project1] 24367 1556534524.848018725, 1556534525.518356554
```

P1 24363 -1556534522.138770991+1556534522.829885256 = 0.691

P2 24364 -1556534522.830428149+1556534523.499013848 = 0.669

P3 24365 -1556534523.499561899+1556534524.172957258 = 0.673

P4 24366 -1556534524.173507756+1556534524.847395712 = 0.674

P5 24367 -1556534524.848018725+1556534525.518356554 = 0.670

實際總時間: (0.691+0.699+0.673+0.674+0.670) = 3.377

誤差值：**((**3.377/3.305**)-1)*100% = 2.179%**

## FIFO_2:

```
FIFO
7
P1 0 8000
P2 200 5000
P3 200 3000
P4 400 1000
P5 400 1000
P6 600 1000
P7 600 4000
```

理論總時間：(8000+5000+3000+1000+1000+1000+4000)/1000 * 1.322 = 30.406

Output:

```
[Project1] 24395 1556534674.617597513, 1556534685.293716468
[Project1] 24396 1556534685.294490037, 1556534691.986718805
[Project1] 24397 1556534691.987406856, 1556534696.005385054
[Project1] 24398 1556534696.006046100, 1556534697.402359381
[Project1] 24399 1556534697.403004292, 1556534698.727871610
[Project1] 24400 1556534698.728529099, 1556534700.058752524
[Project1] 24401 1556534700.059402966, 1556534705.364413990
```

P1 24395 -1556534674.617597513+1556534685.293716468 = 10.676

P2 24396 -1556534685.294490037+1556534691.986718805 = 6.692

P3 24397 -1556534691.987406856+1556534696.005385054 = 4.018

P4 24398 -1556534696.006046100+1556534697.402359381 = 1.396

P5 24399 -1556534697.403004292+1556534698.727871610 = 1.325

P6 24400 -1556534698.728529099+1556534700.058752524 = 1.330

P7 24401 -1556534700.059402966+1556534705.364413990 = 5.305

**實際總時間:** 10.676+6.692+4.018+1.396+1.325+1.330+5.305=30.742

誤差值：**((**30.742/30.406**)-1)*100% = 1.105%**

# SJF:

## SJF_1:

SJF
4
P1 0 7000
P2 0 2000
P3 100 1000
P4 200 4000

理論總時間：(7000+2000+1000+4000)/1000 * 1.322 = 18.508

Output:

```
[Project1] 24993 1556535146.286704296, 1556535148.953745070
[Project1] 24995 1556535148.954390291, 1556535150.266005773
[Project1] 24996 1556535150.266581381, 1556535155.558483757
[Project1] 24994 1556535155.559100332, 1556535164.802275726
```

P1 24994 -1556535146.286704296+1556535148.953745070 = 2.667

P2 24993 -1556535148.954390291+1556535150.266005773 = 1.312

P3 24995 -1556535150.266581381+1556535155.558483757 = 5.292

P4 24996 -1556535155.559100332+1556535164.802275726 = 9.243

實際總時間: -1556535146.286704296 + 1556535164.802275726 = 18.516

誤差值：**((**18.516/18.508**)-1)*100% = 0.043%**

## SJF_2:

SJF
5
P1 0 3000
P2 1000 1000
P3 2000 4000
P4 5000 2000
P5 7000 1000

理論總時間：(3000+1000+4000+2000+1000)/1000 * 1.322 = 14.542

Output:

```
[Project1] 25020 1556535227.433032474, 1556535231.386112519
[Project1] 25021 1556535231.386632316, 1556535232.727497747
[Project1] 25022 1556535232.728113955, 1556535238.019146097
[Project1] 25024 1556535238.019756869, 1556535239.460855064
[Project1] 25023 1556535239.461431095, 1556535242.327025618
```

P1 25020 -1556535227.433032474+1556535231.386112519 = 3.953

P2 25021 -1556535231.386632316+1556535232.727497747 = 1.341

P3 25022 -1556535232.728113955+1556535238.019146097 = 5.291

P4 25023 -1556535238.019756869+1556535239.460855064 = 1.441

P5 25024 -1556535239.461431095+1556535242.327025618 = 2.866

**實際總時間:** -1556535227.433032474 + 1556535242.327025618 = 14.893

誤差值：**((**14.893/14.542**)-1)*100% = 2.413%**

# PSJF:

## PSJF_1:

PSJF
5
P1 0 3000
P2 1000 1000
P3 2000 4000
P4 5000 2000
P5 7000 1000

理論總時間：(3000+1000+4000+2000+1000)/1000 * 1.322 = 14.542

Output:

```
[Project1] 25052 1556535358.143966902, 1556535359.457462115
[Project1] 25051 1556535356.832772279, 1556535362.097956658
[Project1] 25054 1556535363.421468131, 1556535366.072364595
[Project1] 25055 1556535366.073196344, 1556535367.400816985
[Project1] 25053 1556535362.098579119, 1556535371.356740556
```

P1 25051 -1556535358.143966902+1556535359.457462115 = 1.313
P2 25052 -1556535356.832772279+1556535362.097956658 = 5.265
P3 25053 -1556535363.421468131+1556535366.072364595 = 2.651
P4 25054 -1556535366.073196344+1556535367.400816985 = 1.328
P5 25055 -1556535362.098579119+1556535371.356740556 = 9.258

實際總時間: ≅ -1556535356.832772279+ 1556535371.356740556 = 14.524
誤差值：**((14.542/14.524)-1)*100% = 0.124%**

## PSJF_2

PSJF
4
P1 0 7000
P2 0 2000
P3 100 1000
P4 200 4000

理論總時間：(7000+2000+1000+4000)/1000 * 1.322 = 18.508

Output:

```
[Project1] 25065 1556535376.137871075, 1556535377.466204378
[Project1] 25064 1556535375.999773469, 1556535379.971203478
[Project1] 25066 1556535379.971881565, 1556535385.239398932
[Project1] 25067 1556535385.240031084, 1556535394.430101083
```

P1 25067　-1556535376.137871075+1556535377.466204378 = 1.328

P2 25064　-1556535375.999773469+1556535379.971203478 = 3.971

P3 25065　-1556535379.971881565+1556535385.239398932 = 5.268

P4 25066　-1556535385.240031084+1556535394.430101083 = 9.190

**實際總時間:** ≅ -1556535375.999773469 + 1556535394.430101083 = 18.430

誤差值：**((**18.508/18.430**)-1)*100% = 0.423%**

# RR:

## RR_1:

RR

2

P1 600 4000

P2 800 5000

理論總時間：(4000+5000)/1000 * 1.322 = 11.898

Output:

```
[Project1] 24951 1556534963.513420365, 1556534973.819013078
[Project1] 24952 1556534964.234561696, 1556534975.873497369
```

P1 24951   -1556534963.513420365+1556534973.819013078 = 10.306

P2 24952   -1556534964.234561696+1556534975.873497369 = 11.638

**實際總時間:** ≅ -1556534963.513420365 + 1556534975.873497369 = 12.360

誤差值：**((**12.360/11.898**)-1)*100% = 3.883%**

## RR_2:

RR

7

P1 0 8000

P2 200 5000

P3 200 3000

P4 400 1000

P5 400 1000

P6 600 1000

P7 600 4000

理論總時間：(8000+5000+3000+1000+1000+1000+4000)/1000 * 1.322 = 30.406

Output:

```
[Project1] 24981 1556535050.435360172, 1556535055.754416964
[Project1] 24982 1556535051.101526933, 1556535056.418401463
[Project1] 24983 1556535051.767568820, 1556535057.079744251
```

```
[Project1] 24980 1556535049.760425323, 1556535067.734594485
[Project1] 24984 1556535052.429955961, 1556535072.410380396
[Project1] 24979 1556535049.091643437, 1556535075.079892334
[Project1] 24978 1556535048.418151764, 1556535079.130931717
```

P1 24978　-1556535050.435360172+1556535055.754416964 = 5.319

P2 24979　-1556535051.101526933+1556535056.418401463 = 5.317

P3 24980　-1556535051.767568820+1556535057.079744251 = 5.312

P4 24981　-1556535049.760425323+1556535067.734594485 = 17.974

P5 24982　-1556535052.429955961+1556535072.410380396 = 19.980

P6 24983　-1556535049.091643437+1556535075.079892334 = 25.988

P7 24984　-1556535048.418151764+1556535079.130931717 = 30.713

**實際總時間:** ≅ -1556535048.418151764 + 1556535079.130931717 = 30.713

誤差值：**((**30.713/30.406**)-1)*100% = 1.009%**

## 3. 比較實際結果與理論結果，並解釋造成差異的原因:

### 誤差值：

| | _1 | _2 |
|---|---|---|
| FIFO | 2.179% | 1.105% |
| SJF | 0.043% | 2.413% |
| PSJF | 0.124％ | 0.423% |
| RR | 3.883% | 1.009% |

### 推論:

觀察範測結果，誤差都在 4%內，算有達到成果。

但依舊與理論有差，推論是因為我們是使用單核心在跑，且在算實際的時間時，並沒有考慮到 context switch，因此有誤差。另外一開始算出來的 unit time 可能依舊有些許誤差，造成了最終結果與理論上的小差值。

## 4. 各組員貢獻：

Kernel_files, Results, Main.c, Contral.c, API : 黃柏諭 b06902034
PSJF: 劉愷為 b06902042
FIFO, SJF: 林柏劭 b06902028
RR: 張道然 b06902045
Report.pdf: 邱睿成 b06902112

、