**作業系統**

**Project2**

**多個CPU排程法比較**

資工三甲

彭義翔

10527142

**開發平台**

Python 3.6.3 |Anaconda, Inc.| (default, Oct 15 2017, 03:27:45) [MSC v.1900 64 bit (AMD64)] on win32

**使用開發環境**

Window10 1803 64-bit

**使用的程式語言**

python

**說明你的程式設計**

1. **功能**

雖然是分為六項，但第六項其實是前五項都做一次，再把它印在一起，幾乎沒有設計難度(只要data分開存 最後一起印即可)，所以設計思路是把五項都分開作，首先請使用者把想要計算的資料寫進TXT檔案裏頭，格式如下，第一行的第一個數字是想要作哪一種排程，1先到先作FCFS 2知更鳥 RR 3 可奪取最小工作優先 PSJF 4不可奪取最小工作優先 NSJF 5優先等級 PRIORITY ，第二個數字是顯示要作的timeslice 即如果還CPU還沒做完此時間還是要換人作的時間片段，第二行是單純字串” ID CPU Burst Arrival Time Priority”，我可以依照給不同的順序去讀不一樣的資料，依照順序是每個process獨有的ID、CPUBurst、ArrivalTime、Priority，接來來數行即是每個process的資料，皆為正整數，中間用空白或TAB隔開，可一次好幾個不同的txt檔案一直作，然後會輸出”輸入黨”後面多\_output的txt以區隔檔案，裡面內容依序是你選的排程方式名稱，下行是甘特圖(Gantt Chart)，接這是waitingtime，包含數行輸入的prossce的ID及他在此排程的Waiting time，再來以此類推的Turnaround time

1. **流程**

讀進一個依照使用者輸入檔案名稱的txt，並把資料整理存好，重點的prossce存好創的class在用list包起來，分別複製五次(用以區個資料完整性)，在看看要作的哪一樣排程分別呼叫function 如果是6就全部各作一次，每個function基本上都是用來算秒數，看做完了沒，或是做完了要存時間再把它POP，迴圈作完之後在依照ID排好，並回傳甘特圖(Gantt Chart)的STRING ，上述任務的function都會各自呼叫自己不同的排程function (因為每個任務都有差異)，最後出來開檔案把作的資料都輸出至新創的txt檔案裏面。

1. **data structure**

最重要process的class存自己的ID BURST ARRIVAL PRIORITY

還有四個比較小function 存WAITTIME&TURNAROUND(要存的時候不一樣 不是剛創的時候)，外面再使用這些process的時候都是用list整個傳來傳去的所以有不少依照不同的順序排的function，還有gantt 是用list跑船到外面時候再轉成string，其餘都是區域變數用來算時間或是給條件判斷式使用。

**未完成的功能**

基本上作業要求的都完整的處理好也表達出來

美中不足是時間有點小趕

不知道來不來的及作好完美的模組化

程式大概可以再少100行