作業系統 課堂作業7 (Chapter 5) 2018/05/30

1. 名詞解釋:
2. 飢餓(starvation)問題 (b) 處理器親和性(Processor Affinity) (c) 護送效應(convoy effect)

(e) 低優先權行程可能永遠無法執行

(b) 行程對於它目前執行的處理器較有親和性，因行程最近存取的資料已放在處理器的快

取記憶體

(c) 所有行程都在等待一個長行程離開的情況

1. 說明並比較可搶先(preemptive)排班和不可搶先(nonpreemptive)排班。

一個行程在執行時，他的cpu使用權會不會因為其他因素而被搶走

可搶先排班- 一個行程在執行時，使用權可被別人搶走

不可搶先排班-一個行程在執行時，除非自己放棄cpu使用權，否則別人不能搶走他的cpu

使用權

1. 當一個行程發生哪四種情狀之一時，即須執行一次CPU的排班決策？

一) 從執行狀態轉變成等待狀態

二) 從執行狀態轉變成就緒狀態

三) 從等待狀態轉變成就緒狀態

四) 行程終止

1. 分派程式(Dispatcher)主要負責的工作為何？

內容轉換(context switching)

切換成使用者模式

跳到使用者程式的適當位置繼續執行該程式

1. 請描述五種 CPU的排班原則(Scheduling Criteria)及其欲達成的目標。

CPU使用率 (CPU utilization)– 使CPU盡可能地忙碌

產量 (Throughput)– 單位時間所能完成的行程數目

回復時間(Turnaround time) – 單一行程執行完畢所需的時間

等候時間 (Waiting time) – 一個行程在就緒佇列等待的時間

反應時間 (Response time )– 提出一個要求到第一個反應出現的時間(對於分時環境)

1. 給定多個行程相關資訊，如何利用先來先做排班演算法（FCFS Scheduling）、最短工作先做(Shortest-Job-First)排班演算法、最短剩餘時間先做 (Shortest-Remaining-Time-First)排班、優先權排班(Priority Scheduling)、依序循環排班(Round-Robin Scheduling)來進行排班，將排班結果以甘特圖表示並計算平均等待時間(average waiting time)。
2. 請說明並比較多層佇列(Multilevel Queue)排班與多層回饋佇列(Multilevel Feedback Queue)排班。

多層佇列(Multilevel Queue)排班- 對行程進行分類，將不同類型的行程放入不同佇列中，在

每個佇列內使用最適合該類行程的排程方法

(一個行程永遠只在某個佇列中)

多層回饋佇列(Multilevel Feedback Queue)排班- 行程可以在不同的佇列之間移動，可以決定

甚麼時候把行程提升到較高優先全佇列

1. 了解每一種排班方法的優缺點。如:實作SJF排程法的困難。在RR排程法中，時間切片太長或太短各會造成什麼影響？

先來先做排班演算法（FCFS Scheduling）

優點：簡單

缺點：等待時間變動很大，而且平均等待時間

並不短->可預測性低

最短工作先做(Shortest-Job-First)排班演算法

優點: SJF是最佳演算法，所有不可搶先排班法中平均等待時間最短的一個

缺點: 無法預先知道每一個行程的下一個CPU burst時間

最短剩餘時間先做 (Shortest-Remaining-Time-First)排班

優點: SJF排班的可搶先版

缺點: 當有新的行程進入就緒佇列時，如果比現在執行中行程所剩下的CPU Burst時間更短

，則現在執行的行程會被趕出去，由新行程搶先執行

優先權排班(Priority Scheduling)

優點: 分別可採取可搶先及不可搶先作法

CPU優先分配給最高優先權的行程(若最小整數代表最高優先權)

缺點: 可能造成飢餓問題

依序循環排班(Round-Robin Scheduling)

優點: 平均等待時間比較長，但是能提供較好的反應時間

缺點: q非常大時，近似FCFS排程的效果

q非常小時，會造成內文切換的頻率過高，而影響系統效能

1. 負載平衡(Load balancing )的作法有哪兩種？
2. 請說明Multi-Level Queue with Feedback排程之概念。

可搶先的優先權排程

行程可以在不同的佇列之間移動

可以用這種方法實現老化