長庚大學106學年度第一學期作業系統期中測驗 (滿分103)

系級: 姓名: 學號:

1. (8%) 在作業系統中，Interrupts是用來通知處理器來處理特殊事項的一個硬體機制與軟體技術，通常收到Interrupts後，作業系統會透過一連串的操作達成需要的結果。Interrupts又可以分為Software Interrupts與Hardware Interrupts，請分別舉出一項Software Interrupts與一項Hardware Interrupts？

Answer: Software Interrupts: signals, invalid memory access, division by zero, system calls, etc. (a correct answer +4%, a wrong answer -4%)
Hardware Interrupts: services requests of I/O devices, e.g., keyboards, Ethernet adapters, touch panels, etc. (a correct answer +4%, a wrong answer -4%)

2. (8%) 在作業系統中請說明Multiprogramming及Time Sharing的定義。

Answer: Multiprogramming: The operating system keeps several jobs in memory simultaneously (4%).
Time Sharing: Time sharing is a logical extension of multiprogramming, in which CPU switches jobs frequently so that users can interact with each job while it is running (4%).

3. (8%) Real-Time Embedded Systems有許多常見的應用，譬如行車安全系統、手持式多媒體系統、居家或工廠自動控制系統。請說明Real-Time所指的意思為何？

Answer: Real-time means on-time instead of fast. Real-time systems have to get the correct (or qualified) computing results and have to make sure that the results can be derived on time.

4. (8%) 請定義Application Programming Interface (API)與System Call，並說明兩者之間的關係。

Answer: System calls provide the routines for user applications to use the functions provided by operating systems (4%). The API of a programming language serves as a user-friendly link to system calls made available by the operating system (4%). Thus, most of the details of the operating-system interface are hidden from programmers by the API and are managed by run-time support libraries.

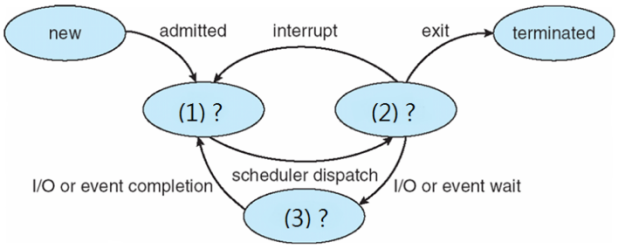
5. (8%) 作業系統中在做Inter Process Communication (IPC)時有兩種方法：Message Passing and Shared Memory。兩相比較下，Message Passing對應用程式的開發者而言較容易使用；而需要大量資料傳輸與頻繁溝通時，妥善地使用Shared Memory將可以得到較好的效能。請說明為何Message Passing效能較好。

Answer: If we use message passing, a serious of system calls has to be invoked for sending each message. By using shared memory, multiple processes can directly access a shared memory area multiple times without invoking many system calls.

6. (8%) 一次的IPC中，會有資料傳送者(Sender)與資料接收者(Receiver)，請問對於資料傳送者而言，使用 Synchronous Message Passing IPC 與使用 Asynchronous Message Passing IPC 有何不同？

Answer: Synchronous Message Passing IPC: The sender has to wait until the message is received.
Asynchronous Message Passing IPC: The sender sends the message and continues.

7. (6%) 下面這張圖簡單地陳述了Process的一生，請填上圖中(1)、 (2)、(3)的內容



Answer: (1) Ready, (2) Running, (3) Waiting.

8. (9%) 請說明 (a) User Thread與Kernel Thread有何不同？Kernel Thread與User Thread有許多種應對關係中請說明何謂 (b) Many-to-One Model 、 (c) One-to-One Model。

Answer: (a) User threads are managed by user-level threads library, and kernel threads are supported by the kernel and are the unit for CPU scheduling.
(b) Many user threads of a process are mapped to one kernel thread.
(c) Each user thread is mapped to a kernel thread.

9. (8%) 請說明Thread Local Storage (TLS)的用途，並說明TLS與local variable有何不同。

Answer: Purpose: TLS allows each thread to have its own copy of data. (4%)
Difference: Local variables are visible only during single function invocation, but TLS visible across function invocations in a thread. (4%)

10. (8%) 對於 Symmetric Multiprocessing (SMP)的多處理器環境，想要做負載平衡有兩種方法：Push Migration 以及 Pull Migration。請說明何謂 (a) Push Migration 以及 (b) Pull Migration。

Answer: (a) If there is an overloaded processor, tasks are moved from the overloaded processor to idle or less-busy processors. (4%)
(b) An idle processor pulls waiting tasks from busy processors. (4%)

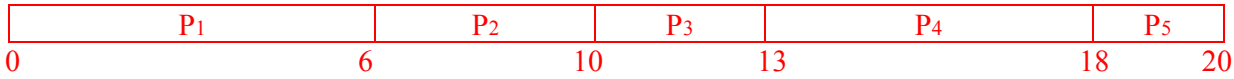
11. (18%) 考慮已經就緒的五個工作，依序為P_1, P_2, P_3, P_4, P_5。使用三個排程演算法FCFS (First-Come, First-Served)、SJF (Shortest-Job-First)以及RR (Round Robin)來排程，而RR所使用的time quantum為4ms。(1)請畫下三個排程演算法的排程圖，(2)請分別算出三個排程演算法中每個工作的等待時間，若無算式一率不給分(算式可以只是簡單的加減法運算)，(3)請分別算出三個排程演算法的平均等待時間，若無算式一率不給分。

Process	Burst Time
P_1	6 ms
P_2	4 ms
P_3	3 ms
P_4	5 ms
P_5	2 ms

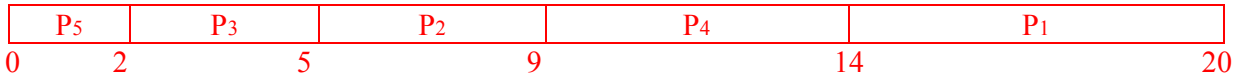
Answer:

(1)

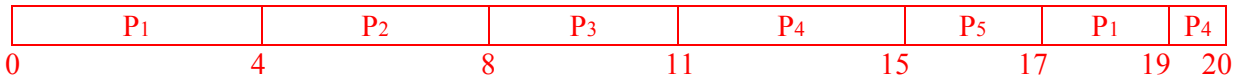
FCFS:



SJF:



RR:



(2)

FCFS: P_1: 6-6=0, P_2: 10-4=6, P_3: 13-3=10, P_4: 18-5=13, P_5: 20-2=18

SJF: P_1: 20-6=14, P_2: 9-4=5, P_3: 5-3=2, P_4: 14-5=9, P_5: 2-2=0

RR: P_1: 19-6=13, P_2: 8-4=4, P_3: 11-3=8, P_4: 20-5=15, P_5: 17-2=15

(3)

FCFS: (0+6+10+13+18)/5= 9.4

SJF: (14+5+2+9+0)/5= 6

RR: (13+4+18+15+15)/5= 13

12. (6%) 有了多處理器(Multiple Processors)平台後，在一個處理器上也開發出多核心的(Multi-Core)環境。有了多核心的環境後，在單一核心上也設計出Hyper-Threading的技術。請說明何謂Hyper-Threading？

Answer: Hyper-Threading (HT) Technology duplicates the register file of a core so as to allow more than one thread to be brought into the core. When there is some memory stall of a thread, another thread in the same core can immediately takeover the ALU for doing its computation. Thus, HT has the potential for improving parallelization of computations.