1. 作業系統內所稱的外殼(Shell)用來做什麼？

接收使用者輸入的命令，解析命令並執行命令。

1. 何謂即時系統(Real Time Systems)?即時系統與連線(On Line)的關係為何?
2. 立即針對問題反應，並在所規範的條件內自動執行相關軟體，且得到正確結果。
3. 由於即時系統必須立即反應，故他一定是連線的(On-line)，連線系統不一定需要即時。
4. 程式執行系統呼叫的方式?
5. 使用組合語言(Assembly)執行系統呼叫
6. 使用系統呼叫函數庫(Library)的系統呼叫函數(System Call Function)
7. 高階語言的I/O statement
8. API Library Function
9. 請說明Spooler運作的方式。

將專屬設備模擬成分享設備，例如印表機，當很多輸出要執行時，Spooler會將這些輸出存在disk中，並依順序由印表機印出。

1. 當某個程式執行系統呼叫時，作業系統如何處理此系統呼叫？

系統呼叫是一種中斷

1. 中斷發生，先將控制權交給OS
2. OS將執行到一半的程式狀態存起來
3. OS判斷是什麼中斷，呼叫ISR來執行
4. ISR執行完，將CPU status Restore回去
5. 控制權交還給執行到一半的程式，並繼續執行

[CPU如何處理中斷?]

CPU硬體結構設計會把CPU當時的status先透過內部硬體線路或Microcode把它存在CPU內部的某處(ex.Global memory)，再讓PC指向OS，OS一出面就知道剛剛有東西被中斷，且知道存在CPU的哪裡，所以OS的save CPU status是去CPU的global area去save到OS

1. 使用中斷處理輸出/輸入與使用直接記憶體存取(Direct Memory Access)處理輸出/輸入，中央處理器(CPU)扮演的腳色有何不同？

中斷處理I/O -> CPU會處理中斷

DMA ->CPU閒置到Memory存取完畢

1. 請說明區塊設備(Block Device)的緩衝區法(Buffering)在進行邏輯紀錄(Logical Record)讀取(Read)時的運作方式。

Buffer = 1Physical Record = N Logical Record

執行read時，CPU從Device讀取整塊Physical Record擺入Buffer，由於Buffer裡有N個Logical Record，所以下次讀取時可以直接到Buffer存取。

1. 請說明階層式記憶體(Hierarchical Memory)。

由上層到下層為Register、Cache、Main Memory、Secondary Memory，任何須執行的資料或程式都必須搬至Main Memory中，再交由CPU執行。

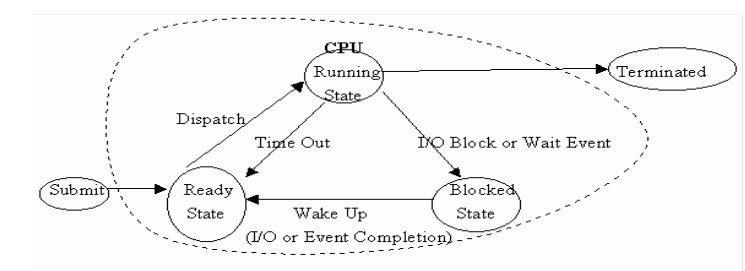
1. 作業系統能否確認某些程式永不停止？ 為什麼？

可以，某些程式可以從程式碼就判斷出她永不停止，例如: while(true) {}

1. 請說明VMware Architecture。

一部電腦有Host OS，其上架有很多Guest OS，user兩種OS皆可操作，若操作Guest OS，會透過virtualizaion layer轉換成Host OS處理，VMware可以使應用程式在不同的OS環境下執行。

1. 請繪圖並說明處理程序(Process)三態的變化。



假設只有1顆CPU，Ready state從Queue的最前面取一個Process Dispatch到Running Queue裡面，交給CPU執行，若Time Slice用完了，就進入Ready State的Queue的最後面去等待執行，而若Process執行完畢，則teminate

1. 請寫出一個父處理程序(Parent Process)衍生(Fork)出子處理程序(Child Process)的程式範例，並以此範例說明子處理程序與父處理程序之執行行為有何不同？

Parent(){

pid = fork() ;

if ( pid == 0 ) {

play1() ; // Child process的pid為0

exit(0) ; // 所以執行這兩行

}

Play2() ; // parent的pid不為0，所以不會進if

exit(0) ; // 直接執行這兩行

}

1. 請說明處理程序與執行緒的關係為何？

一個Process可以有一個或一個以上的thread，這些thread彼此共用code section、data section、resources，但每個thread都有自己的PC、register set、stack。

1. 請就排程(Scheduling)及輸出/輸入這二個角度說明使用者執行緒(User Thread)與核心執行緒(Kernel Thread)有何不同？

[Scheduling]

在Time Sharing的環境下，若Time Slice是50ms，且一個process產生多個thread，若這些thread為user thread則多個thread共用50ms，若這些thread為Kernel Thread，則每條Thread都可以擁有50ms的Time Slice

[I/O]

當某條thread做I/O時，若為user thread，則其他thread會被

blocked，若為Kernel Thread則其他Thread不會被Blocked

1. 為何優先等級排程法(Priority Scheduling)會造成餓死(Starvation)？

因低Priority的Process可能一直被高Priority的Process搶先佔有CPU，則可能發生低Priority的Process一直無法佔有CPU

1. 請說明現代電腦作業系統內使用的排程法是如何運作？

是使用Multilevel Feedback Queue

1. 提供多個Queue，每個Queue對應一個Priority，Process依Priority在對應的Queue排隊，系統以Time Sharing機制到最高Priority的Queue取Process佔有CPU，當Time Slice用完，該Process就會排到Queue後面。執行某一Queue時，若較高Priority的Queue有process，則改成執行較高Priority的queue裡的process
2. 使用aging，允許process依aging提升或降低priority。
3. 請說明多階層佇列(Multilevel Queue)排程法的運作方式。

提供多個Queue，每個Queue對應一個Priority，Process依

Priority在對應的Queue排隊，系統以Time Sharing機制到最

高Priority的Queue取 Process佔有CPU，若較高Priority的

Queue有Process，則改成執行較高Priority的Queue裡的

Process，由於Process的Priority一開始就決定好了，所以可能

會有餓死的狀況。

1. 何謂系統呼叫(System Call)？

系統呼叫是程式設計者所撰寫之程式與OS的介面，且任何使用系統資源的程式都必須透過System call。

1. 請分辨Multiprocessing與Multiprogramming有何不同？

Multiprocessing是多顆CPU在一部電腦中，同時執行工作並共用同一個OS、設備，使處理能力增加

Multiprogramming是同時有許多程式在Main memory中，輪流且快速的由CPU來執行，且等待I/O時，CPU可以至Main Memory取另一程式來執行。

1. 虛擬機器(Virtual Machine)提供時間多工(Time Multiplex)及空間多工(Space Multiplex)，請問時間多工及空間多工分別做什麼？

時間多工:程式可以分時使用CPU

空間多工:程式可以存在Main Memory不同位置

1. 何謂環境切換(Context Switch)？環境切換有那幾種情況？
2. 當Time Slice用完之後，CPU會執行Time out，將Process變成Ready State，並至Ready Queue裡Dispatch另一個Process進入Running State。
3. 當目前Process要做I/O或Event

由Running State -> Block State

1. 當目前Process Terminate

由Running State -> Terminate

1. 下列那些排程法可以實現為可奪取(Preemptive)？(First Come First Serve、Shortest Job First、Priority、最高反應時間比率優先、Round Robin、Multilevel Queue、Multilevel Feedback Queue)

SJF、Priority、最高反應時間比率優先、Round Robin、Multilevel Queue、Multilevel Feedback Queue

1. 在什麼情況下優先等級排程法(Priority Scheduling)會造成餓死(Starvation)？請問要如何解決此問題？

(1)因低Priority的Process可能一直被高Priority的Process搶先佔有CPU，則可能發生低Priority的Process一直無法佔有CPU

(2)利用Aging解決，利用Response Ratio隨時間升降Priority

1. 現代電腦的結構是一種范紐曼結構(Von Neumann Architecture)，請問范紐曼結構如何運作？

Von Neumann Architecture是一種儲存架構，任何程式欲被執行，都必須由硬碟中經由BUS搬至Main Memory再透過CPU執行，CPU計算好的資料也要經由Main Memory傳入設備儲存。

1. 請說明Java Virtual Machine運作方式。

Java所寫的程式經由Java Compiler轉換成Byte Code，該Byte Code可以跨平台，傳送到不同型電腦，該不同型電腦有該電腦的Java Run Module，他解析Byte Code並透過該電腦執行程式

1. 何謂pipe？

將某個Process的Output做為另一個Process的Input。

1. 何謂多重程式(Multiprogramming)？

Multiprogramming是同時有許多程式在Main memory中，輪流且快速的由CPU來執行，且等待I/O時，CPU可以至Main Memory取另一程式來執行。

1. 何謂分時系統(Time Sharing System)？

OS依CPU Scheduling，將CPU的時間切割為Time Slice，例如50ms。任何程式佔有中央處理器時，若用罄一個時間片段，則輪由下一個程式佔有中央處理器。同樣的，若下一個程式用罄時間片段，再由另一個程式執行。這樣的系統稱為分時系統（Time Sharing）。

1. 何謂Microkernel？

將OS內不必要的功能去掉，只留下需要的功能。

1. 設備驅動程式(Device Driver)如何操控周邊設備?

Device Driver將要執行的工作填入Device Controller的Register內，Device controller讀取裡面的Register並依據內容工作，完成後再將控制權給OS。

1. 假如我們同時執行三個Word。此Word程式使用處理程序來寫，與使用執行緒(Thread)來寫，有何不同?
2. 使用Main Memory的方式不同

* Process:有3份Address Space在Main Memory，也就是有3份Code、Data、Resource、PC、Register Set、Stack。
* Thread:有1份Address Space在Main Memory由3個Thread共用，

有1份Code、Data、Resource

另有3份PC、Register Set、Stack

1. 使用Data方式不同

* Process:3個Process必須透過內部通訊來傳輸資料
* Thread:1個Address Space裡3個Thread可直接存取Address Space的Data或Resources

1. 衍生和消失的方式不同

* Process:每執行一個Word就要Fork一個對應的Process，OS需處理所有事項，當多個Word結束，OS也要處理exit
* Thread:因共用Address Space，結束時OS只處理1次exit。

1. Process需做中斷式Context Switch，Thread Context Switch代價較輕，因不用切換或Copy Address Space，僅需存PC、Register Set跟Stack
2. 請說明執行緒的應用方式?
3. 一個Process有很多Thread做相同的事:

利用Thread共用Code、Data、Resources

1. 一個Process有很多Functions並行執行:

->每個function輪流使用CPU

->每個function都能寫成一個Process，但有包袱問題

(3) 兼具做相同的事及許多Functions並行執行

33.有一種執行緒(Thread)的應用是達到Concurrent Functions的目的，請問是什麼意思？

程式內每一個function均對應一個thread，它使得每一個function均能輪流執行幾個statement。