

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(a)一項Software Interrupt與(b)一項Hardware Interrupt。

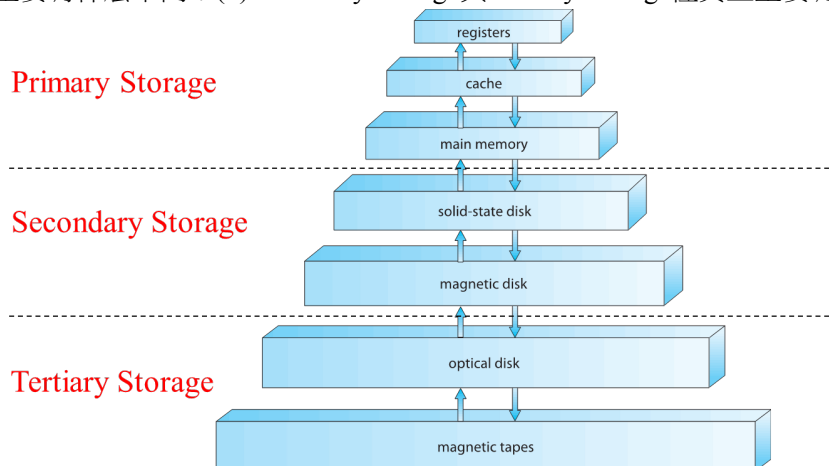
Answer: Software Interrupts: signals, invalid memory access, division by zero, system calls, etc. (4%)

Hardware Interrupts: service requests of I/O devices, e.g., keyboards, Ethernet adapters, touch panels, etc. (4%)

2. (8%) 請定義Program和Process的不同。

Answer: A program is a passive entity stored on hard disk, and a process is an active entity which is loaded into memory for the execution on a CPU.

3. (10%) 在記憶體與儲存裝置的架構中主要可以分為三層，請問(a)Primary Storage與Secondary Storage性質上主要有什麼不同？(b)Secondary Storage與Tertiary Storage性質上主要有什麼不同？



Answer: (a) Primary storage is volatile, and secondary storage is nonvolatile. (5%)

(b) Tertiary storage is some removable media, and secondary storage is not during the online using period. (5%)

4. (8%) 在作業系統中請說明(a)Multiprogramming及(b)Time Sharing的定義。

Answer: (a) The operating system keeps several jobs in memory simultaneously (4%).

(b) 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%).

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Shared Memory效能較好。

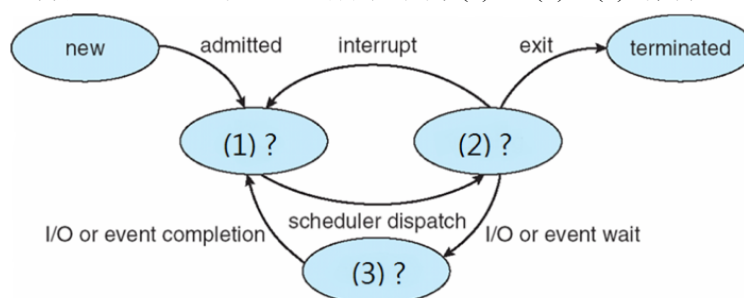
Answer: If we use message passing, a serious of system calls has to be invoked for sending each message. By using shared memory, multiple processes can directly access a shared memory area multiple times without invoking many system calls

6. (8%) Monolithic Kernel與Microkernel是兩種不同的作業系統設計方式，與Monolithic Kernel相比較之下，請舉出 (a)一個Microkernel的優點、 (b)一個Microkernel的缺點。

Answer: (a) Advantage: Microkernel is more modulized, and thus, it is more portable, reliable, and easy for extensions. (4%)

(b) Disadvantage: There are more inter-process communication (IPC) calls in Microkernel. Thus, the performance might be worse. (4%)

7. (9%) 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. (10%) 在 Multithreaded Programming 的環境底下我們有三種不同的系統資源分配與排程管理單元的架構，其中兩種是 Many-to-One Model 與 One-to-One Model，請說明(a)何謂 Many-to-One Model、(b)何謂 One-to-One Model。

Answer: (a) In a process, there are one kernel thread and multiple user threads, and the user threads share the kernel thread. (5%)

(b) In a process, there could be multiple kernel threads and multiple user threads, and each user thread maps to a kernel thread, individually. (5%)

9. (8%) 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.

10. (12%) 假設每次呼叫fork()都是成功的，請寫出以下程式在POSIX環境下執行後的輸出結果。

```

#include<sys/types.h>
#include<stdio.h>
#include<unistd.h>
int main()
{
    pid_t pid1, pid2, pid3;
    pid1 = fork();
    if (pid1 > 0)
    {
        wait(NULL);
        printf("AAA\n");
        pid2 = fork();
        if (pid2 == 0)
        {
            printf("BBB\n");
        }
        else
        {
            wait(NULL);
            printf("CCC\n");
        }
    }
    else
    {
        printf("DDD\n");
        pid3 = fork();
        if (pid3 > 0)
        {

```

```

        wait(NULL);
        printf("EEE\n");
    }
    else
    {
        printf("FFF\n");
    }
    printf("GGG\n");
}
return 0;
}

```

Answer:

DDD

FFF

GGG

EEE

GGG

AAA

BBB

CCC

11. (18%) 考慮已經就緒的五個工作，依序為P1, P2, P3, P4, P5。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 |
|---------|------------|
| P1 | 12 ms |
| P2 | 1 ms |
| P3 | 2 ms |
| P4 | 3 ms |
| P5 | 8 ms |

Answer:

(1)

FCFS: (2%)

| | | | | |
|----|----|----|----|----|
| P1 | P2 | P3 | P4 | P5 |
| 0 | 12 | 13 | 15 | 18 |
| | | | | 26 |

SJF: (2%)

| | | | | |
|----|----|----|----|----|
| P2 | P3 | P4 | P5 | P1 |
| 0 | 1 | 3 | 6 | 14 |
| | | | | 26 |

RR: (2%)

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| P1 | P2 | P3 | P4 | P5 | P1 | P5 | P1 |
| 0 | 4 | 5 | 7 | 10 | 14 | 18 | 22 |
| | | | | | | | 26 |

(2)

FCFS: P1: 12-12=0, P2: 13-1=12, P3: 15-2=13, P4: 18-3=15, P5: 26-8=18 (2%)

SJF: P1: 26-12=14, P2: 1-1=0, P3: 3-2=1, P4: 6-3=3, P5: 14-8=6 (2%)

RR: P1: 26-12=14, P2: 5-1=4, P3: 7-2=5, P4: 10-3=7, P5: 22-8=14 (2%)

(3)

FCFS: (0+12+13+15+18)/5= 11.6 (2%)

SJF: (14+0+1+3+6)/5= 4.8 (2%)

RR: (14+4+5+7+14)/5= 8.8 (2%)