

三民輔考一高考 計算機概論

類科：工業行政、電力工程、電子工程、電信工程

109 年

一、線路交換(circuit switching)和分封交換(packet switching)是兩個重要的網路資料交換技術。請詳述兩者的工作原理並加以比較。

【答】：

線路交換：

也稱為「電路交換」，是指通訊兩端在通訊之前，必須先建立實際的連線。我們常用的市內電話線路，也稱為「普通老式電話服務」(POTS: Plain Old Telephone Service)就是線路交換的一個例子。當 A 要撥打電話給 B 時，會透過建置在各處的中華電信機房內的交換機來建立一條連接雙方的通道，雙方的聲音就可透過此線路來傳遞。利用此線路，網路上的節點（電腦或其他設備）也能傳遞資料。

分封交換：

也稱為「封包交換」，是指資料在傳遞之前，需先將資料分成數個大小相等的區塊，分別加上標頭資訊後，再透過網路傳遞。這些封包在傳輸前不需先建立通訊路徑，各封包的傳送路徑也不一定相同。

兩者之比較：

	線路交換	分封交換
專屬線路	是	否
封包路徑是否相同	是	否
傳輸速度	快	較線路交換慢
線路使用率	低	高

二、什麼是跨平台編譯器(cross-compiler)？請詳加解釋並舉例說明其用途。

【答】：

跨平台編譯器指的是在某一個作業系統平台下，可以產生在另外一個作業系統平台上執行的檔案的編譯器。有時候程式檔案無法在原平台上編譯，例如嵌入式系統缺乏作業系統，無法直接在系統上編譯，就必須藉由跨平台編譯器的輔助。先原平台編譯出可在其他平台上執行的檔案（通常是目的檔），再透過燒錄或上傳的方式，將該目的檔放到目標平台上執行。

三、有一個二元樹(binary tree)共有 10 個節點，每個節點均儲存一個英文字母。若此二元樹：

- 使用中序走訪 (inorder traversal) 的結果為：R T D P X Y K G A B
 - 且使用層序走訪 (level order traversal) 的結果為：P R X D A T K B Y G
- 則此二元樹為何？請畫出此二元樹。

【答】：

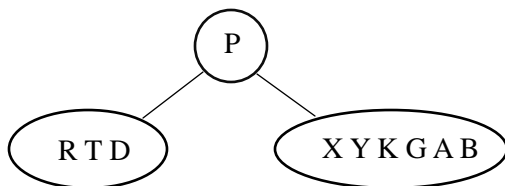
中序走訪：左子樹 -> 樹根 -> 右子樹

層序走訪是從樹根開始，依序由上而下、由左而右走訪，亦即廣度優先走訪。

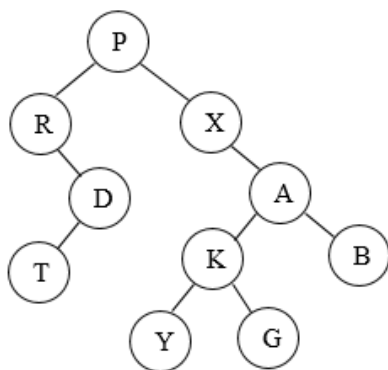
中序：R T D **P** X Y K G A B

層序：**P** R X D A T K B Y G

由上可推出樹根：



再依據中序追蹤的樹序，可以畫出此二元樹：

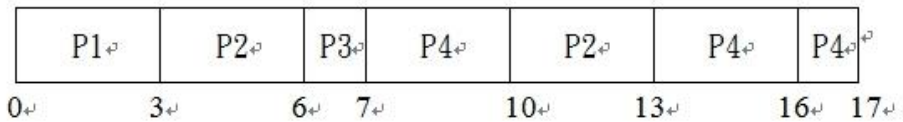


四、某一作業系統之 CPU 排程為循環分配方法(round-robin scheduling)，今有一排程，共有四個程序，其排隊順序為 P1、P2、P3 及 P4，個別所需執行時間如下表所示。請問在此排程中，若時間配額(time quantum)分別採用 3 毫秒與 5 毫秒，則那一種時間配額可以得到較小之平均回覆時間(average turnaround time)？請畫出甘特圖(Gant chart)及詳列計算過程。

程序	所需執行時間 (毫秒)
P1	3
P2	6
P3	1
P4	7

【答】：

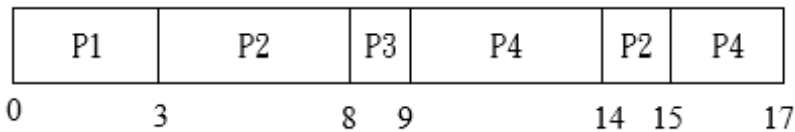
3 毫秒：



平均等待時間： $(0 + 3 + (10 - 3) + 6 + 7 + (13 - 3))/4 = 8.25$ 毫秒

平均回覆時間： $(3 + 6 + (13 - 3) + 7 + 10 + (17 - 3))/4 = 12.5$ 毫秒

5 毫秒：



平均等待時間： $(0 + 3 + (14 - 5) + 8 + 9 + (15 - 5))/4 = 9.75$ 毫秒

平均回覆時間： $(3 + 8 + (15 - 5) + 9 + 14 + (17 - 5))/4 = 14$ 毫秒

由上可得知，時間配額採用 3 毫秒可以得到較小之平均回覆時間。



五、以下 C++程式的目的為何？詳述執行流程並寫出程式的輸出。（20 分）

```
#include <iostream>
#include <iomanip>
using namespace std;

int main(){
    int x = 30, y = 100, ok = 1;
    int i, j;

    for(i = x ; i <= y; i++){
        ok = 1;
        for(j = 2; j < i ; j++){
            if(i % j == 0){
                ok = 0;
                break;
            }
        }
        if(ok == 1) cout << i << " ";
    }
    cout << endl;
    return 0;
}
```

【答】：

外層迴圈：i 從 30 到 100

內層迴圈：j 從 2 到 i-1

變數 ok：用來判斷 i 是否會被 j 整除。若是可以則結束內層迴圈，並將 ok 的值設為 1。內層迴圈結束後，若 ok 的值等於 1 則表示 i 不會被 j 的任一個值整除，亦即 i 是質數。最後再輸出 i。

此程式可以輸出 x~y 之間的所有質數，此例是找出 30~100 之間的質數。

以下是輸出結果：

31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

範例練習

一、有一個二元樹的中序走訪順序為 ABCDEFGHI，後序走訪順序為 BACFEIHGD。請劃出此二元樹。

【答】：

中序：

左子樹 樹根 右子樹

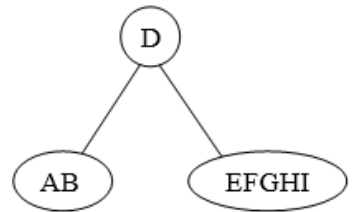
後序：

左子樹 右子樹 樹根

Sep 1：

中序： ABC D EFGHI

後序： BAC FEIHG D

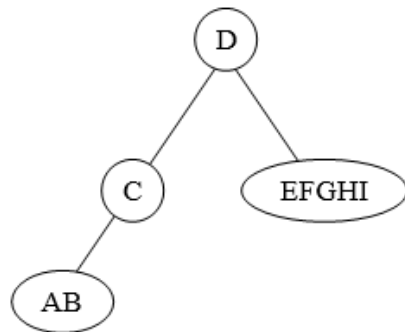


由上可知D一定是樹根。根據中序追蹤，在節點D左方的字母(ABC)一定都位在左子樹；在D右方的字母(EFGHI)一定都位在右子樹。

Sep 2：針對 ABC 再分

中序： A B C

後序： B A C

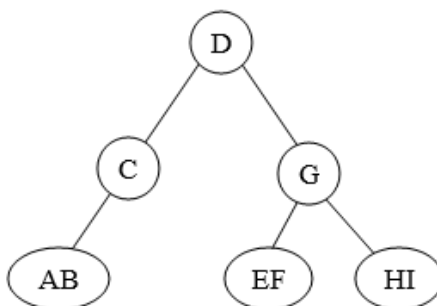


由上可知C一定是第二層的樹根。根據中序追蹤，在節點C左方的字母(AB)一定都位在左子樹，在C右方的字母一定都位在右子樹。可是在這個階段，中序追蹤時C右方並沒有字母，表示節點C沒有右子樹，因此畫出來的圖形就會如同上面所示。

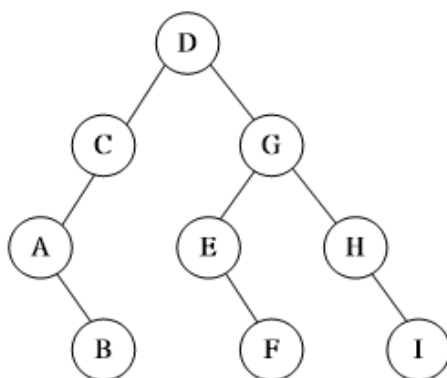
Sep 3 : 針對 EFGHI 再分

中序 : E F G H I

後序 : F E I H G



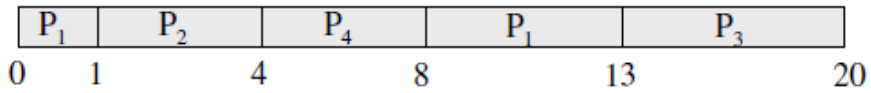
由上可知G一定是第三層的樹根。根據中序追蹤，在節點G左方的字母(EF)一定都位在左子樹。在G右方的字母(HI)一定都位在右子樹。依照同樣的做法以此類推，最後可以得到下圖：



二、某一作業系統之 CPU 排程為最短剩餘時間優先。請畫出甘特圖(Gant chart)及平均等待時間。

行程	CPU 暴衝時間 (毫秒)	到達時間
P1	6	0
P2	3	1
P3	7	2
P4	4	3

【答】：



平均等待時間： $((8 - 1) + (1 - 1) + (13 - 2) + (4 - 3))/4 = 4.75$ 毫秒

三、請寫出下列 C 語言程式碼執行的結果，請寫出說明。

```
int func(int n) {
    if (n < 10)
        return n;
    return func(n / 10) + n % 10;
}

int main() {
    printf("%d", func(12345));
    return 0;
}
```

【答】：

輸出：15

```
func(12345)
= func(1234) + 5
= func(123) + 4 + 5
= func(12) + 3 + 4 + 5
= func(1) + 2 + 3 + 4 + 5
= 1 + 2 + 3 + 4 + 5
= 15
```