

《程式設計概要》

試題評析

今年第一題考「程式語言」基本概念；第二題如預測考出「Javascript」有關event的函數用法；第三題與第四題延續往年的題型，考「鏈結串列」與「迴圈遞迴」的傳統問題。就考題趨勢來看，javascript的考題為未來主要題型，程度好的考生可獲得80分以上。

- 一、(一)程式語言有分直譯式與編譯式，請問有何差異？(9分)
 (二)也有程式語言的程式碼會先編譯成虛擬機器碼(virtual machine code)，然後在執行時直譯或編譯成機器語言來執行此程式，請問為何要這樣做？(8分)
 (三)下列何者為直譯式，編譯式，或間接編譯成虛擬機器碼之程式語言：Ada, C, C++, C#, Fortran, Java, JavaScript, Pascal, Perl, PHP, VBScript, Visual Basic, Visual Basic .NET。(8分)

答：

(一)

編譯程式	直譯程式
1.輸出是目的程式，較浪費記憶空間且不可交談。 2.執行時間較快，因可以產生最佳碼。 3.發展時間通常較慢。 4.具有彈性；可同時與其他語言接觸。 5.程式可以原始或目的形式儲存。	1.輸出是執行的結果，不產生目的程式。較節省空間，且可交談。 2.執行時間較慢，但可交談方式使原始程式易於找錯。 3.發展時間通常較快。 4.無彈性；通常不能與其他語言同時接觸。 5.程式只能以原始形式儲存。

(二)先編譯成中間碼(如Java byte code)可以做到跨平台的分散式處理。如可將Java原始程式編譯成byte code後成為class檔後可移植到其他平台直譯執行。

(三)直譯式：Javascript, Perl, PHP, VBScript, Visual Basic

編譯式：ADA, C, C++, C#, Fortran, Pascal

間接編譯成虛擬機器碼: Java, Visual Basic .NET

- 二、撰寫一個html網頁，其利用JavaScript之onMouseOver及onMouseOut功能來動態顯示如圖(a)(b)所示之"Over"及"Out"兩張圖像。當滑鼠移到虛線框內則顯示圖(a)之網頁，當滑鼠移出虛線框，則顯示圖(b)之網頁。額外功能是當滑鼠移到虛線框內，使用者點擊(click)圖像，則跳至考選部的主網頁(<http://www.moex.gov.tw/>)。請依照圖(c)的格式，填入正確的語法，完成此html網頁。
 Note：(I)字串"JavaScript Mouse Over/Out Example"是網頁中文字部分，不屬於圖像本身。(II)虛線框為說明用，不會顯示在網頁中。(III)假設兩張"Over"及"Out"圖像是儲存於與html網頁同一目錄中，其檔案名稱是out.jpg及over.jpg。(25分)



(a)當滑鼠移到虛線框內



(b)當滑鼠移出虛線框

```

<HTML><HEAD>
<title>JavaScript Mouse Over/Out Example</title>
<SCRIPT language=JavaScript>
[ ]
</SCRIPT></HEAD>
<BODY>
<h2>[ ]</h2>
<A HREF=[ ]
[ ]
<IMG SRC=" [ ]" NAME=" picture" BORDER=0>
</BODY>
</HTML>

```

答：

```

<HTML><HEAD>
<title>JavaScript Mouse Over/Out Example</title>
<SCRIPT language=JavaScript>
function mouseover(){
    picture.src="over.jpg";
}
function mouseout(){
    picture.src="out.jpg";
}
</SCRIPT></HEAD>
<BODY>
<h2>JavaScript Mouse Over/Out Example</h2>
<A HREF="http://www.moex.gov.tw/" onmouseover="mouseover()" onmouseout="mouseout()">
<IMG SRC="out.jpg" NAME="picture" BORDER=0>
</BODY>
</HTML>

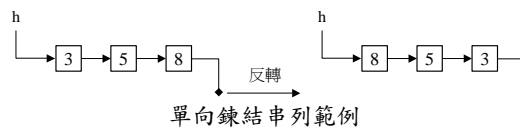
```





三、用C語言撰寫一個函式，能反轉一單向鍊結串列（singly linked list）：struct node* reverse(struct node *h)。Note：請勿複製節點。其中節點的資料結構為：

```
struct node{
    int d;
    struct node *next;
}
```



單向鍊結串列範例

答：

```
struct node *reverse(struct node *h){
    struct node *p, *q;

    if(h!=NULL && h->next!=NULL){ // 當鍊結串列至少一個節點以上
        p=h->next;
        h->next=NULL;
        while(p!=NULL){
            q=p;
            p=p->next;
            q->next=h;
            h=q;
        }
    }
    return h;
}
```

四、用C語言撰寫反覆結構（for-loop）及遞迴函式（recursive）2個版本的函式，能計算出費式數列

(Fibonacci Sequence) : int F(int n), 其數學定義如下 : $F_0=0$, $F_1=1$, and $F_n=F_{n-1}+F_{n-2}$ for $n>1$ 。
(25分)

例如：呼叫F(6)計算出8 and F(7)計算出13

答：

遞迴:

```
int F(int n){
if(n<0) return(-1); // input error
if(n==0 || n==1) return(n);
else return(F(n-1)+F(n-2));
}
```

非遞迴:

```
int F(int n){
int a,b,c,i;
if(n<0) return(-1); // input error
if(n==0 || n==1) return(n);
```

```
a=0;
b=1;
for(i=2;i<=n;i++){
c=a+b;
a=b;
b=c;
}
return(c);
}
```

```
void main(void){
```

```
printf("F(6)=%d\n", F(6));
printf("F(7)=%d\n", F(7));
}
```

【參考書目】

《程式設計概要》(高點文化)。