

《程式語言》

一、何謂程式語言 (programming language)？又有那些分類？(17 分)

試題評析	此題需回答出程式語言的定義，再說明程式語言分類。由於程式語言依照不同面向有多種分類方式，因此若能將分類加以結合說明會使答案更完整。此題為 101 年檢事官、102 年高考相似題，充分練習的同學應能心應手。
考點命中	《高點程式語言講義》，金乃傑編撰，第一回，頁 2；第六回，頁 56~63。(完全命中！)

【擬答】

- (一) 程式語言 (Programming Language) 是描述演算法 (Algorithm) 和資料結構 (Data Structure) 的系統化符號，其目的為指揮電腦做指定操作，如運算、儲存或傳輸資料的語言。
- (二) 根據機器執行平台的相依程度，可以將程式語言粗分平台相依的低階語言與平台獨立的高階語言。低階語言例如：機器語言 (Machine Language)、組合語言 (Assembly Language)，這些語言與 CPU 的指令直接對應，因此執行效率高，但撰寫困難，且不能移植到其他平台使用。而高階語言獨立於執行平台 (machine independent)，依照設計理念與執行方式，可以分為：命令型語言、函數型式語言、邏輯型式語言與物件導向語言。特性及內容分述如下：
1. 命令型語言 (Imperative Language)：由一連串順序性的敘述所組成，主要概念是透過指定運算改變變數的狀態以完成所要執行的工作，故稱為「命令式語言」。另外，再重複任務上，透過迴圈完成。由於程式設計師必須用敘述描述問題的解決步驟 (How to do)，故也屬於程序式語言 (Procedural Language)。主要實作的語言包括：Pascal、C、Basic、Fortran、COBOL 等。
 2. 函數型語言 (Functional Language)：由函數呼叫指令所形成的運算式所組成。運算式由函數呼叫指令所構成，成為程式的基本執行單位。在函數型語言中，變數的設定必須經由函數傳入參數來處理，不使用指定敘述；而控制與重複結構也都由特殊的函數呼叫完成。函數型語言還有一個特性是參考透明性 (Referential Transparency)，指的是函數給予相同的參數，就會有相同的結果，不具有邊際效應。在函數型語言中，我們不需要告訴電腦要「怎麼做」，只要描述「做什麼 (What to do)」，故屬於宣告式程式語言 (declarative language) 或非程序式語言 (non-procedural language)。主要實作的語言包括：LISP、APL、SNOBOL 等。
 3. 邏輯型語言 (Logical Language)：由一群宣告所組成。宣告是一種符號邏輯 (symbolic logic)，是一些推理的事實或規則，而不是指定敘述或控制流程，其主要目的為推論 (inference)，為人工智慧的高階語言。相較於命令型語言，邏輯語言的算術運算效率較低，專長為邏輯符號的處理。在邏輯型語言中，我們不需要告訴電腦要「怎麼做」，只要描述「做什麼」，故屬於宣告式程式語言或非程序式語言。主要實作為 Prolog 語言。
 4. 物件導向語言 (Object-oriented Language)：由類別 (class) 實作的物件 (objects) 所構成，物件間透過訊息 (message) 溝通操作。有三大特色：
 - (1) 封裝 (Encapsulation)：將資料及操作包藏在類別中，有助於資訊隱藏 (Information hiding) 及撰寫大型系統。
 - (2) 繼承 (Inheritance)：父類別的資料和操作可供子類別使用，增加再用性 (reusability)。
 - (3) 多型 (Polymorphism)：執行時根據實際的物件型態決定呼叫的方法，使相同程式碼有不同的執行模式。實際上，在當其他類型的語言包括物件導向的特性，就可以成為物件導向語言，主要實作語言有：C++、Java、C#、Smalltalk、Ada 等。

二、請說明下述文法是否為混淆的 (ambiguous)？又下列句子有那些可由此文法產生？(24 分)

$S \rightarrow aSbS \mid bSaS \mid \varepsilon$

- (a) abab
- (b) aababb
- (c) bababa

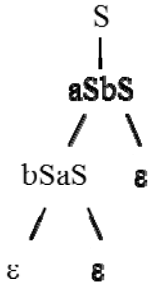
(d) aaaabbb

試題評析	此題為BNF與模糊文法考題，極相似於95年檢事官考古題。題型變化度不高，若平日多加練習，拿到滿分難度不高。
考點命中	《高點程式語言講義》，金乃傑編撰，第六回，頁 6~17。(完全命中！)

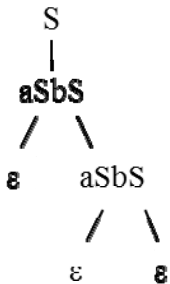
【擬答】

混淆 (ambiguous) 文法指的是若一語言的某一語句，依其 BNF 文法規則來推導，可繪出兩棵以上的不同剖析樹。以題目 BNF 而言，若語句為：abab，則產生兩棵剖析樹如下：

剖析樹一：



剖析樹二：

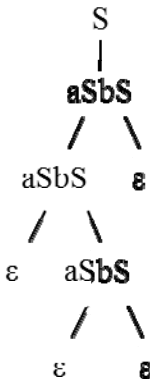


依照題意，可使用文法產生的語句如右：a、b、c

補充：

a 的剖析樹(如擬答)

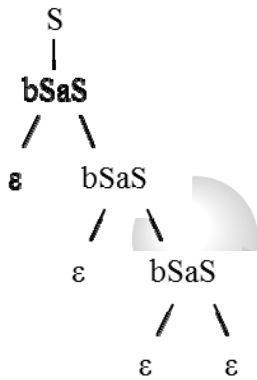
b 的剖析樹(其中一棵)



【高點法律專班】

版權所有，重製必究！

c 的剖析樹(其中一棵)



d 的剖析樹：無法繪出

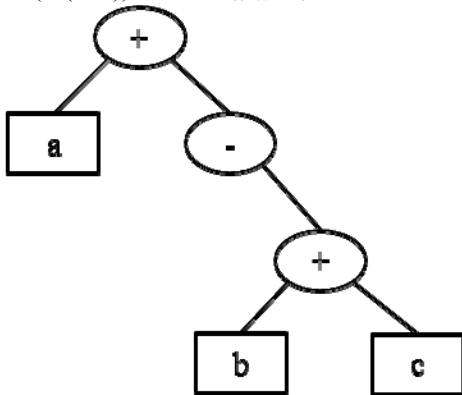
三、一個運算式很自然地可以畫成一棵樹，稱為運算式樹 (expression tree)，請畫出下列運算式的運算式樹，又有那些方法可以 traversal 此運算式樹。(20 分)

運算式： $a+(-(b+c))$

試題評析	此題跨到資料結構領域，在正課「編譯器詳細流程」章節亦有對運算式樹加以介紹。若同學能整合資料結構知識，亦應可寫出答案。
考點命中	《高點程式語言講義》，金乃傑編撰，第六回，頁 45~46 及上課筆記。

【擬答】

$a+(-b+c)$ 之運算式樹如下：



其追蹤 (traversal) 方法有前序 (Per-order)、中序 (In-order) 與後序 (Post-order) 三種，追蹤結果如下：

前序： $+a+bc$

中序： $a+-b+c$

後序： $abc++$

【高點法律專班】

版權所有，重製必究！

四、請寫出三種參數傳遞方法？並敘述其作法？(18 分)

試題評析	此題要說明參數傳遞的運作方式，與以往要求說明參數傳遞結果的題目不同，但若掌握計算方式，應該更為容易書寫掌握分數。
考點命中	《高點程式語言講義》，金乃傑編撰，第二回，頁 7~8。(完全命中！)

【擬答】

程式語言常見的參數傳遞有：傳值呼叫 (Call by Value)、傳址呼叫 (Call by Address)、傳名呼叫 (Call by Name)。

以下說明執行步驟：

參數傳遞方法	作法
傳值呼叫	1. 配置副程式形式參數記憶體空間。 2. 複製主程式實際參數的值到副程式形式參數的記憶體。 3. 執行副程式中的敘述。 4. 傳回結果，釋放副程式。
傳址呼叫	1. 將主程式實際參數的位址對應到副程式形式參數(使副程式形式參數與主程式實際參數互為別名)。 2. 執行副程式中的敘述。 3. 傳回結果，釋放副程式。
傳名呼叫	1. 將主程式實際參數的名稱，取代副程式中對應的形式參數名稱。 2. 將副程式中的形式參數參考到主程式實際參數的記憶體位址。 3. 執行副程式中的敘述。 4. 傳回結果，釋放副程式。

五、何謂繫結(binding)? 又發生繫結的時期種類有那些?(21分)

試題評析	此題為繫結的基本概念，與以往說明動態/靜態繫結或指出程式中有那些繫結處不同的是，此題與第四題一樣只要說明概念即可，亦應更能掌握分數。
考點命中	《高點程式語言講義》，金乃傑編撰，第三回頁 33。(完全命中！)

【擬答】

繫結(Binding)是程式實體(program entity)與其屬性(attributes)發生關聯(association)的動作稱之。依照繫結的時間點可分為四個時期：定義(Definition)時期、實作(Implementation)時期、編譯(Compilation)時期與執行(Execute)時期。以表格說明如下：

時期	內容	實例
定義	限制程式語言所提供的資料型態、符號、變數名稱寫法、函數……的功能與意義。	1. while 具有重複執行的功能。 2. C 語言陣列只有一維。 3. 等號(=)具有指派變數的作用。
實作	所有跟機器相關的特性及變數可能的值。在不同計算機中，即使相同的數學運算式，亦可能得到不同的結果；固定在某種機器上才有的特性，另一台機器可能就無法執行。	在 32 位元系統整數範圍為 -214783648 至 214783647。
編譯	由程式設計師定義資料變數與型態的關係。	宣告 int a，則將變數 a 繫結到整數型態。
	由編譯器定義變數所使用的方法(overload)。	x = x + 10，「+」執行何種型態的相加，由編譯器決定。
	載入時期，將靜態變數、全域變數繫結到指定的記憶體位址。	配置全域變數
執行	進入副程式時，在活動紀錄中配置區域變數空間。	<pre>foo(){ int x }</pre> 則呼叫 foo()時，系統在活動紀錄配置 x 變數的空間。
	執行動態記憶體配置語法，在 Heap 中配置物件空間。	Obj o1 = new Obj(); 執行此敘述時，配置 o1 物件空間。
	執行指定敘述時，將新的值與變數名稱產生關聯。	執行到 i = i + 3 敘述時，將 i 與新的變數值繫結。