

《程式語言》

試題評析	綜合來說，今年基本分數題為第二、三、五題。第一、四題是涉獵廣泛領域的考生才較易拿到分數。一般程度學生可以得到 60~70 分；詳細研讀上課內容，且有廣泛知識的考生拿到 70 分以上並不難。
	第一題：系統分析範圍，SOA 屬於相當新的開發模式，須要有實務經驗或有廣泛涉獵相關知識才能解答完整，而 SaaS 則是系統分析第六章範圍。
	第二、三題：講義第三章文法範圍，屬於基本題型。
	第四題：偏於網路，系統安全範疇。
	第五題：遞迴為補充講義及課堂講解的內容範疇，也屬基本題型。

- 一、(一)服務導向架構 (Service-oriented architecture, SOA) 是一種事件驅動 (Event-driven) 的程式結構 (Programming paradigms)，試說明之。(10 分)
- (二)雲端運算 (Cloud computing) 包括以下幾個層次的服務：基礎設施即服務 (IaaS)，平台即服務 (PaaS) 和軟體即服務 (SaaS)。試舉例並說明何謂軟體即服務 (SaaS)？(10 分)

【擬答】

(一)SOA

一個企業業務的應用程式稱為一個獨立的「邏輯單位」，而對企業營運層面而言，則可稱為是一項「服務」，在企業的整體運算環境中就存在著多個「獨立邏輯／業務服務」，且需要對其進行妥善設計、開發、佈建、管理等，也因此需要採用服務導向架構。

SOA 的核心概念底層在將軟體轉化為可重複使用與分享的服務，因此，架構的建立，首先在分析各個應用系統中，有哪些功能是必要開放成為服務的；一般直覺的方法就是看看有哪些功能是重複出現在不同的系統中，而據此決定哪些功能是要包成服務。而另一種方式，則是從流程中著手，藉由 IT 與 Business 部門的合作，進行流程內涵的審視與分析，一方面由流程實務需求上，確定哪些功能需要被建構成服務，另一方面可藉由審視流程的機會，進行流程的改進。

(二)SaaS

軟體即服務是基於網際網路技術興起的一種完全創新的軟體應用模式。

軟體商將應用軟體統一部署在自己的伺服器上，用戶根據自己實際需求，通過網際網路向廠商訂購所需的應用軟體服務，按訂購的服務多少或時間長短向廠商支付費用，並通過網際網路獲得廠商提供的服務。

最大的優點是用戶不用再購買軟體，而改用向提供商租用軟體，來管理企業經營活動，且無需對軟體進行維護，服務提供商會全權管理和維護軟體。

以雲端運算為例，目前許多企業使用的企業入口系統(Enterprise portal)，可以由軟體供應商提供系統及伺服器，讓企業用戶透過網路連結此系統台在企業內部交換資訊。

- 二、下述文法會產生二進位數字，試剖析其產生之值為那一數字之倍數。(20 分)

$\langle C \rangle \rightarrow \langle C \rangle 0 \mid \langle A \rangle 1 \mid 0$

$\langle A \rangle \rightarrow \langle B \rangle 0 \mid \langle C \rangle 1 \mid 1$

$\langle B \rangle \rightarrow \langle A \rangle 0 \mid \langle B \rangle 1$

【擬答】

三的倍數

$\langle 3n \rangle \rightarrow 0 \mid \langle 3n \rangle 0 \mid \langle 3n+1 \rangle 1$

$\langle 3n+1 \rangle \rightarrow \langle 3n+2 \rangle 0 \mid \langle 3n \rangle 1 \mid 1$

$\langle 3n+2 \rangle \rightarrow \langle 3n+1 \rangle 0 \mid \langle 3n+2 \rangle 1$

將 $\langle 3n \rangle$ 用 $\langle C \rangle$ 取代， $\langle 3n+1 \rangle$ 用 $\langle A \rangle$ 取代， $\langle 3n+2 \rangle$ 用 $\langle B \rangle$ 取代，則得題目文法，故此題為三的倍數之文法。



三、舉一例說明什麼是混淆文法 (ambiguous grammar) ? (20 分)

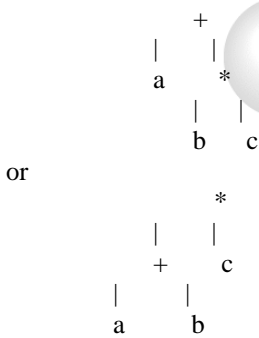
【擬答】

混淆文法(ambiguous grammar)是指一句子(sentence)可以由此文法推導出兩個或兩個以上的剖析樹(parse tree)，則稱此文法為混淆文法。

Ex.以下文法為混淆文法

$$\begin{aligned} \langle \text{expr} \rangle &\rightarrow \langle \text{expr} \rangle + \langle \text{expr} \rangle \\ &\quad | \langle \text{expr} \rangle * \langle \text{expr} \rangle \\ &\quad | \langle \text{id} \rangle \\ \langle \text{id} \rangle &\rightarrow a | b | c \end{aligned}$$

則句子 $a + b * c$ 可以推導出以下兩顆剖析樹



四、試從應用程式安全 (application security) 角度說明弱點偵測 (vulnerability detection)、警訊分析 (alert analysis) 及資訊鑑識 (Information Forensics) 之目的及作法。(20 分)

【擬答】

(一)弱點偵測

[目的]：找出應用程式已經存在或是潛在的安全性問題。

[作法]：安全弱點的偵測以一個偵察階段開始，進行蒐集目標系統與資源的重要資料。這個階段將引導進入系統準備階段，於目標系統上徹底地檢查所有已知的安全性弱點。再進入回報階段，將所有的發現都分類為高、中與低度危險的類別，最後提出改進安全性（或降低安全弱點所可能帶來之風險）的方法。

(二)警訊分析

[目的]：找出入侵系統偵測工具所記錄的資料，分析真正造成系統威脅的對象。

[作法]：通常利用資料探勘(Data mining)技術，配合過去發生的歷史紀錄去比對，找出真正有意義的警訊。

(三)資訊鑑識

[目的]：蒐集分析造成系統問題的原因和事證

[作法]：準備工作(申請取得相關資訊的許可)→收集資訊(收集所有可能的事證文件資料)→驗證(驗證資料的正確可靠性)→分析(分析所收集的資料導出具體結論)→保護產出結果。

五、試比較使用遞迴副程式 (recursive call，即副程式呼叫本身) 與不使用遞迴副程式之時間複雜度 (time complexity) 高低，並舉例說明之。(20 分)

【擬答】

通常迴圈版本副程式的效率會高於同樣問題遞迴版本的副程式，原因如下：

(一)同樣的時間複雜度下，遞迴版本的副程式需要額外時間處理參數傳遞及函數堆疊。

(二)遞迴的解法須要先將問題拆解，再組合回去，若在拆解問題過程中，子問題重覆拆解，則遞迴版本副程式效能就會較低。

Ex. Fibonacci 數列，若以遞迴版本解，複雜度為 $O(2^n)$ 的 n 次方，而迴圈版本為 $O(n)$ 。



