

國立臺北商業大學附設空中進修學院 114 學年度第一學期期末考參考答案

考試科目：二專 413 資料處理

考試日期：114 年 12 月 21 日 節次：1

一、選擇題：每題 3 分，共 60 分

C	1.	在資料處理中，「向量化」的主要優勢為？(A)增加程式碼複雜度 (B)減少記憶體使用 (C)提高執行效率 (D)美化圖表
C	2.	向量化的程式碼通常會取代哪種傳統結構？(A)函數定義 (B)if 條件式 (C)for 迴圈 (D)try except
B	3.	向量化的特性可用於那些資料別？(A)字串 (B)numpy 陣列與 pandas 結構 (C)函數定義 (D)字典
C	4.	在向量化概念中，哪一項可以避免冗長的迴圈並提升效率？(A)使用 if-else 巢狀判斷 (B)將資料轉為文字 (C)將資料以矩陣方式一次運算 (D)單純增加記憶體容量
C	5.	Python 中的縮排在控制流程中的作用？(A)顯示區塊顏色 (B)分隔區塊結束 (C)決定程式區塊範圍 (D)增加可讀性但非必要
C	6.	若想要在 for 迴圈中提前跳出迴圈，使用的關鍵字？(A)skip (B)continue (C)break (D)stop
C	7.	哪一種情境適合用於「不知道幾次，但持續條件成立就會重複」的情境？(A)for (B)if (C)while (D)def
C	8.	函數的參數列表必須放的位置為？(A)大括號中 (B)方括號中 (C)小括號中 (D)雙引號中
B	9.	若函數定義為 'def green(name="user"):'，則稱 'name' 為？(A)不定參數 (B)預設參數 (C)關鍵參數 (D)匿名參數
C	10.	有關函數命名的敘述正確為？(A)可以使用數字開頭 (B)可以包含空格 (C)建議使用有意義的名稱 (D)可與內建函數同名以覆蓋系統
B	11.	NumPy 陣列的主要優勢之一？(A)支援鍵值對結構 (B)高效的效量化運算 (C)提供 GUI 操作介面 (D)完全不需要安裝
C	12.	'ndarray' 屬性會回傳什麼資訊？(A)元素型別 (B)陣列的值 (C)陣列的形狀(維度大小) (D)陣列中的最大值
C	13.	在 Python 中，物件推導最常用來建立何種類型的資料型態？(A)函數 (B)類別 (C)容器物件(如 list、set、dict) (D)模組
D	14.	哪種不是 comprehension 支援的資料型別？(A)list (B)set (C)dict (D)tuple(不可變)
D	15.	隱式迴圈與傳統明確的 for 迴圈相比的主要優勢為？(A)可處理大量數據 (B)減少縮排錯誤 (C)可直接搭配 GUI 工具 (D)可精簡的語法與效能提升。
C	16.	使用隱式迴圈可在一行中完成的操作為？(A)條件判斷語字典建立 (B)資料載入與函數定義 (C)資料迭代與轉換 (D)模組匯入與異常處理
C	17.	若使用 '[f(x) for x in data]'，其中 'f(x)' 是一個？(A)固定常數 (B)字串 (C)函數運算 (D)資料框架
B	18.	何者關於隱式迴圈敘述是錯誤的？(A)能用在 list、set、dict 建構 (B)比 for 迴圈慢 (C)語法更簡潔 (D)能與條件語句結合
B	19.	'ndarray' 屬性可用來查詢？(A)陣列維度 (B)資料型別 (C)最大值 (D)元素個數
C	20.	若 'df["age"].mean()'，其功能為？(A)計算欄位最大值 (B)計算欄位總和 (C)計算欄位平均值 (D)回傳欄位型別

二、簡答題：每題 10 分，共 40 分。

1. 簡述推導式的優勢為何？

(答)1. 簡潔性：推導式能夠將傳統循環和條件語句簡化為一型，所以使用推導式可以減少程式碼行數，使代碼更易於閱讀和維護。

2. 可讀性：直觀地顯示資料轉換的過程，使代碼更具表達力，幫助更快理解程式意圖。

3. 效率：推導式通常比使用傳統迴圈方式更具效率，因為其底層最佳化和更少的解釋器開銷，處理大量資料時，能顯著縮短時間。

2. 為何要學 NumPy？

(答)1. 高效的陣列運算

(1) NumPy 提供多維陣列物件 ndarray，效率遠勝過 Python 的 list。

(2) 資料儲存的方式為連續記憶體區塊，處理速度超快。

2. 豐富的數學函式庫

(1) 可直接對整個陣列做加減乘除運算。

(2) 提供統計、矩陣算式、傅立葉轉換、線性代數等工具。

3. 強大的資料處理功能

(1) 支援切片(slicing)、索引(indexing)、reshape、合併(concatenate)、分割(split)等操作。

(2) 能快速做排序、過濾、聚合。

4. 與其他套件相容效高

(1) 是 Pandas、Matplotlib、SciPy 等工具的基礎。

(2) 機器學習套件(如 TensorFlow、PyTorch)等底層也用 NumPy 陣列當作核心資料結構。

5. 記憶體效率與執行速度超優

(1) 用 C 語言寫成的底層運算，大大提升數值運算效能。

(2) 記憶體占用低，運算速度快，非常適合處理大數據集。

3. 不使用隱式迴圈的時機？

(1) 當需要進行複雜的邏輯操作或有條件的處理時，顯式迴圈更靈活。

(2) 需要進行逐步調試或追蹤每一運算過程時，顯示迴圈能提供更細緻的控制。

4. 除了 NumPy 外，還有可支援向量化的？

(1) Pandas：DataFrame 中可以直接做加減乘除，篩選等操作。

(2) TensorFlow/PyTorch：做 AI 時處理“張量(tensor)”。

(3) Dask：支援平行運算的大型資料套件。