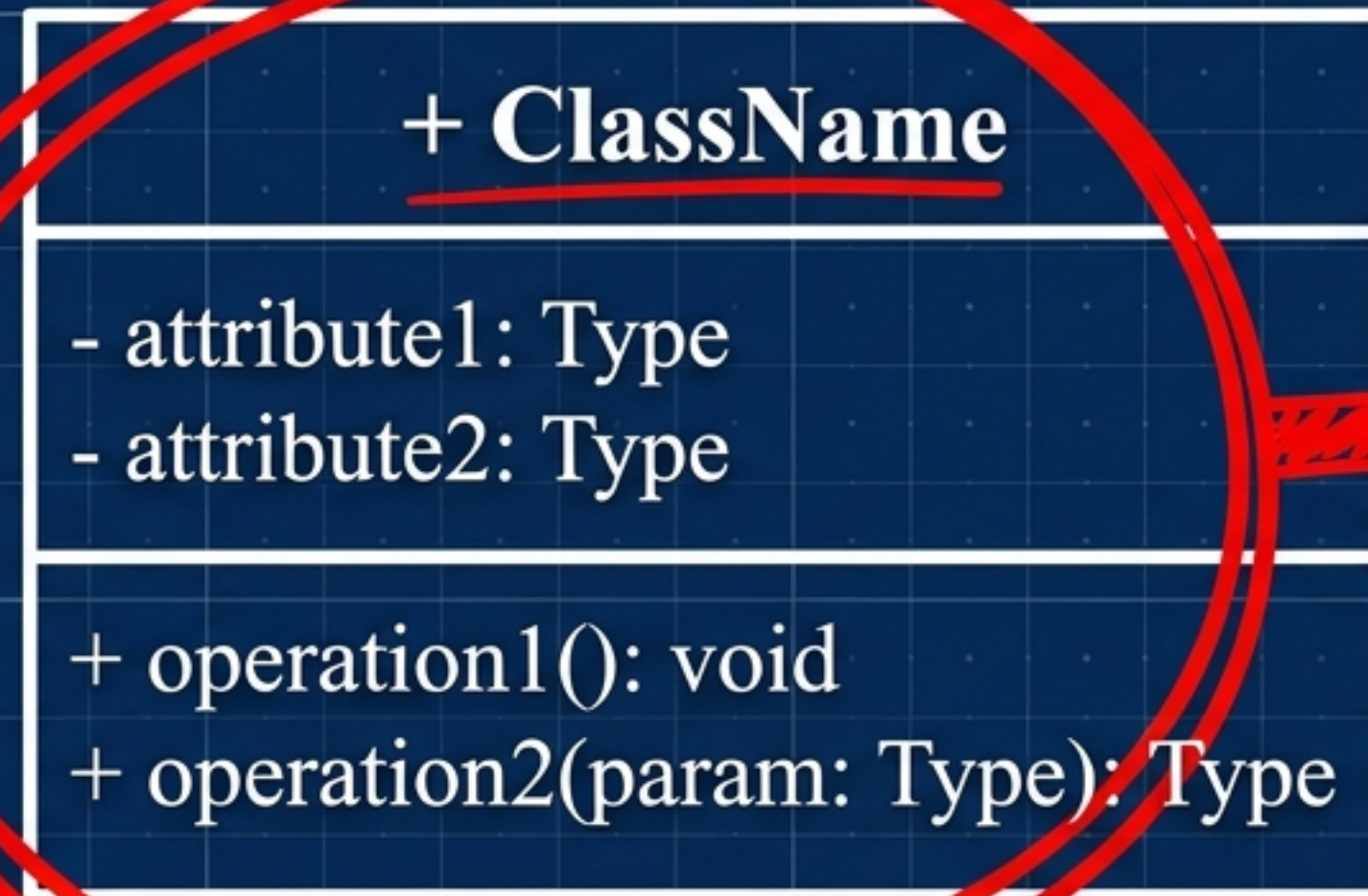


OOT Structure Modeling: The Professor's Masterclass

從零開始的物件導向技術與 UML 考試攻略 (From Zero to Exam Mastery) ✓



Focus on
Section A1 & B2!
考核重點!

Module Code: ITP4909
Object-Oriented Technology

THE JOURNEY

1. **基礎概念 (Core Concepts):**
Classes & Objects
2. **核心結構 (Structural Elements):**
Multiplicity & Relationships
3. **七步分析法**
(The 7-Step Domain Analysis)
4. **終極實戰 (Exam Synthesis):**
Section A vs. Section B

THE EXAM MAP

Section A1 (13 marks):
Detailed Class Diagram (詳細類別圖)

Section A2 (12 marks):
UML to Java Implementation (代碼實現)

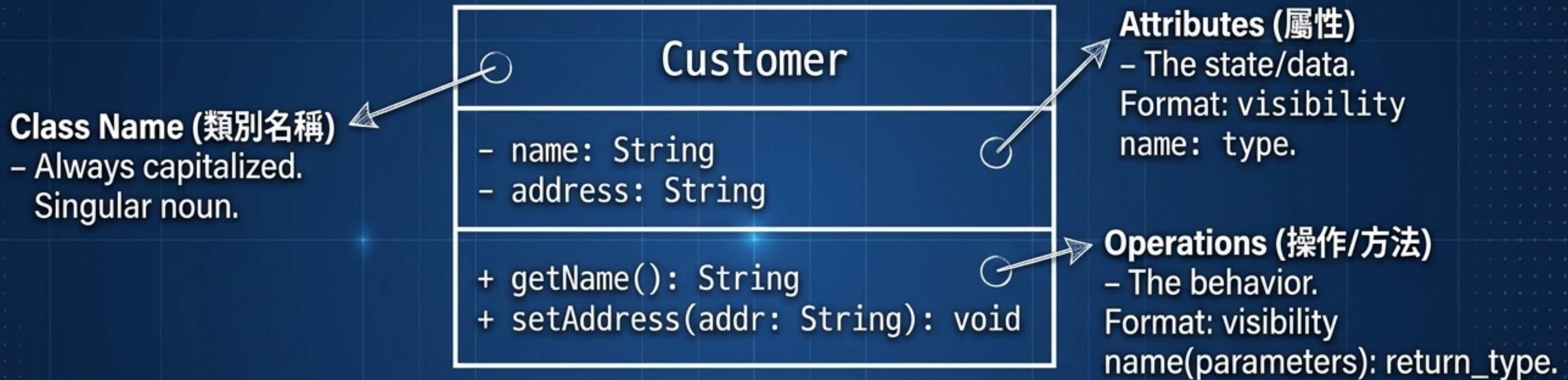
Section B2a (7-8 marks):
Textual Analysis (文本分析)

Section B2b (6-8 marks):
Initial Class Diagram (初步類別圖)

PROFESSOR'S TIP

這門課的秘訣在於：**不要盲目畫圖**。先看清楚題目是 **Section A** 還是 **B** !
(The secret to this course: **Don't just draw blindly**. Identify if it's **Section A** or **B** first!)

Anatomy of a UML Class



EXAM FOCUS BOX

Visibility Symbols (可見性符號): **必背!** (Must memorize!)

+ = public (公開)

- = private (私有 - most attributes should be private!)



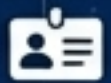




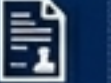

= protected (受保護 - used in inheritance)

The Multiplicity Map (多重性圖解)

1	Exactly one (剛好一個)	
0..1	Zero or one (零或一個)	
0..* or *	Zero to many (零至多個)	
1..*	One to many (一至多個 - At least one!)	

考核點: Section A1 explicitly requires you to show the multiplicity of **EACH** association. Missing these = instant **mark deduction!**
(Section A1 必須必須標明每一個關聯的多重性，漏掉直接扣分！)

Relationship Hierarchy Matrix (關聯關係層次)

Notation (符號)	Concept (概念)	English Keyword	Example (例子)
	Association (關聯) - Structural connection. Peer-to-peer.	uses, has a	  Customer → Account
	Aggregation (聚合) - “Has-a” relationship. The part can exist independently of the whole. (整體與部分的關係，部分可獨立存在).	has a, contains	  Department <-> Teacher
	Composition (組合) - Strong lifecycle dependency. The part dies with the whole. (強依賴的整體與部分，部分隨整體消亡).	consists of, is a part of	  Order ◆- LineItem

Exam trick: If the problem says ‘consists of’ or ‘is a part of’, think Aggregation/Composition. If it’s just ‘works for’ or ‘books’, use a simple Association.

The Inheritance Tree (繼承/泛化結構)

Generalization (泛化):
Extracting common
attributes and
operations into a
Superclass (父類別).



Specialization (特化):
Pushing specific
attributes down to
Subclasses (子類別).

題目看見 “is a kind of”, “types of” (例如: day drivers and night drivers), 立刻畫出繼承結構! 這是 Section A1 拿高分的關鍵! (When you see ‘types of, immediately draw an inheritance tree! This is key for high marks in A1!)

A Tale of Two Diagrams: Diagnostic Matrix

Section B2b - Initial Class Diagram

Marks: 6-8 marks

Goal: The 'Big Picture' skeleton. (宏觀骨架)

Requirement: Classes, Relationships, Association Names.

STRICT NO-NOs (畫了會浪費時間):

- ❌ NO Attributes
- ❌ NO Operations
- ❌ NO Multiplicities
- ❌ NO Navigability ✘

Section A1 - Detailed Class Diagram

Marks: 13 marks

Goal: The Complete Blueprint. (完整藍圖)

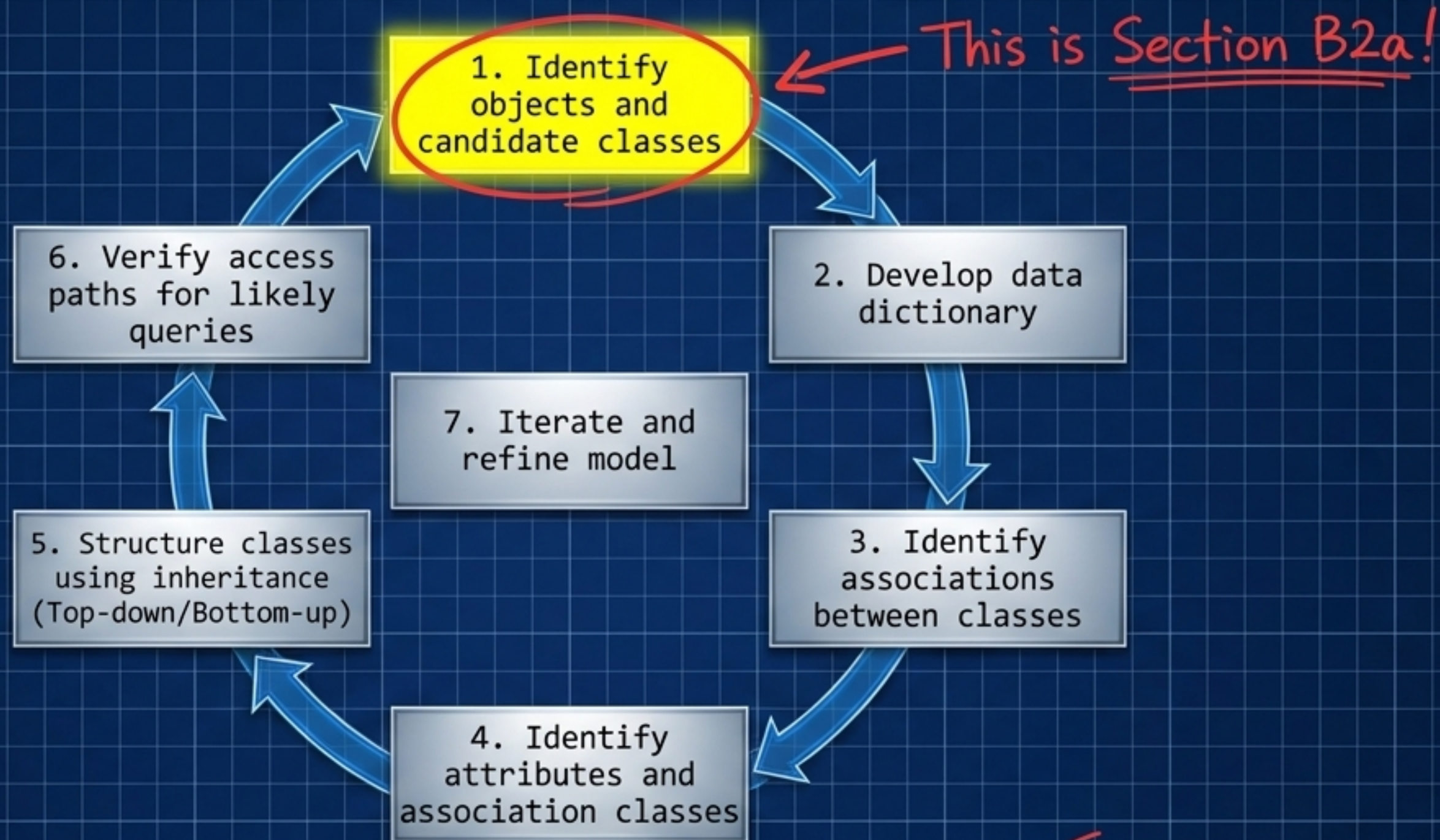
Requirement: Classes + All Attributes + All Multiplicities + Association Names + Inheritance (if applicable).

MUST INCLUDE (不畫會扣分):

- ✅ Exact Multiplicities (1..*) ✓
- ✅ Specific Attributes (-licenceNumber) ✓

Know your battlefield. Don't waste 10 minutes drawing attributes in Section B2b! (認清考題要求，不要在 B2b 浪費時間畫屬性！)

The 7-Step Domain Analysis Method (七步領域分析法)



This isn't just theory. Steps 1-3 give you the answer to Section B2. Steps 4-5 give you the answer to Section A1. (這七步是解題公式。前三步解 B 卷，後兩步解 A 卷。)

The Extraction Matrix (Section B2a: Textual Analysis)

Exam Excerpt

A sports club has an online booking system for members to book sports facilities. A member can book different facilities in a booking.



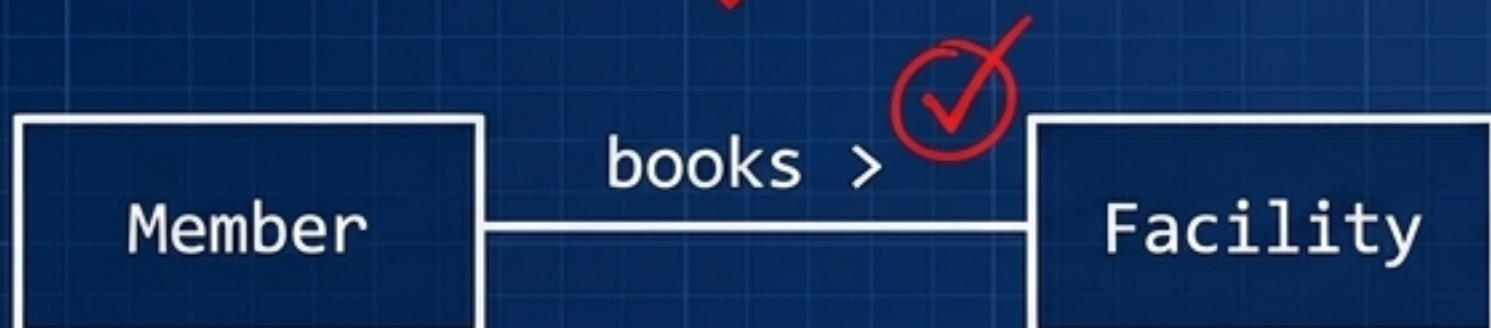
Noun Phrase (名詞)	Category (類別)	Verdict (結論)
member	Concept (概念)	Candidate Class (候選類別)
booking	Event (事件)	Candidate Class
facility	Tangible Object (實體)	Candidate Class
booking system	Irrelevant (無關系統本身)	Discard (捨棄)

Section B2a (8 marks):
必須列出名詞，寫出它是概念/實體/事件，並解釋為何入選！
(You must list the noun, state its category, and explain why it is chosen as a candidate class!)

Finding Associations & Attributes (尋找關聯與屬性)

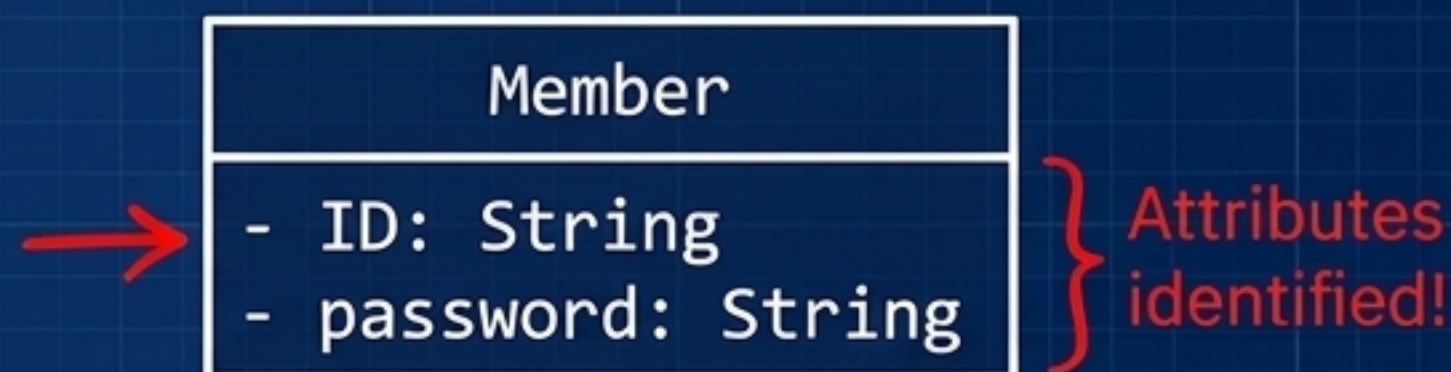
Panel 1: Finding Associations (Step 3)

The Clue: Look for verbs.
“A member can **book** different facilities.”



Panel 2: Finding Attributes (Step 4)

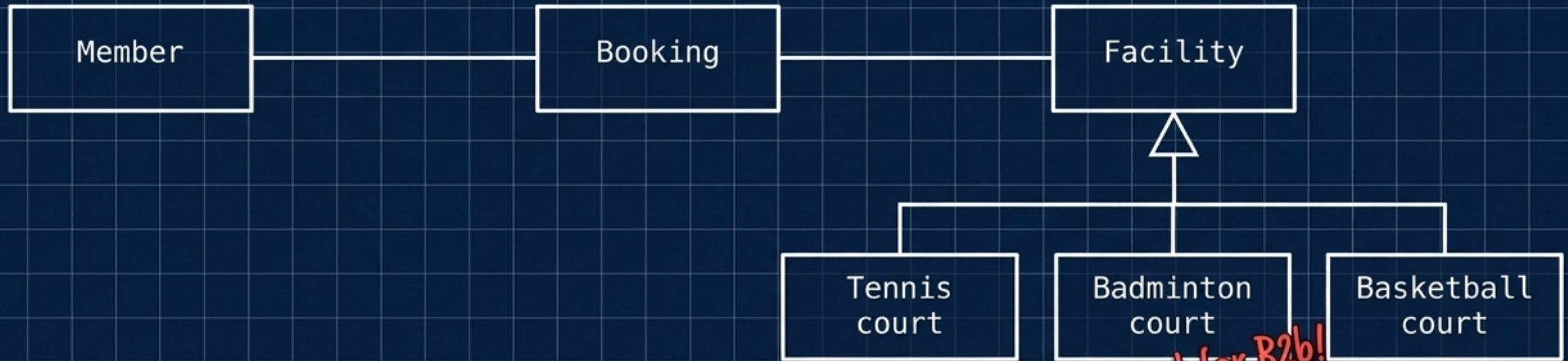
The Clue: Look for possessive phrases or adjectives.
“ID **of the** member”, “password **of the** member”, “closing time **of the** item.”



Professor's Tip

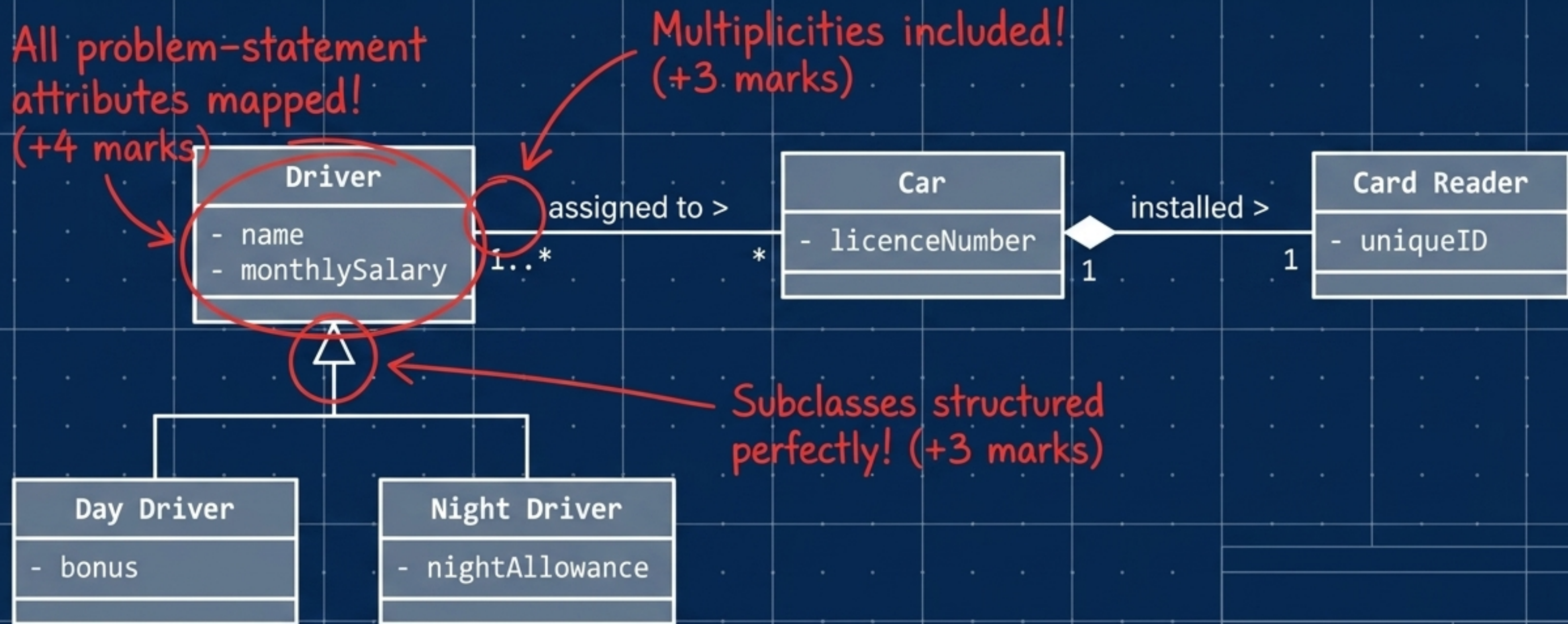
Implementation details like **‘linked lists’** or **‘arrays’** in the text should be IGNORED in class modeling. (考題如果出現 linked list 等底層代碼細節，在畫類別圖時請直接忽略！)

Synthesis: The Initial Class Diagram (Section B2b - 6 Marks)



100% Correct for B2b!
✓ Classes identified.
✓ Relationships drawn.
✓ Association names included.
✗ No attributes or multiplicities to waste time.
(這就是 B2b 的滿分答案，乾淨俐落！)

Synthesis: The Detailed Class Diagram (Section A1 - 13 Marks)



From Blueprint to Code (Section A2: Java Implementation)

UML Blueprint

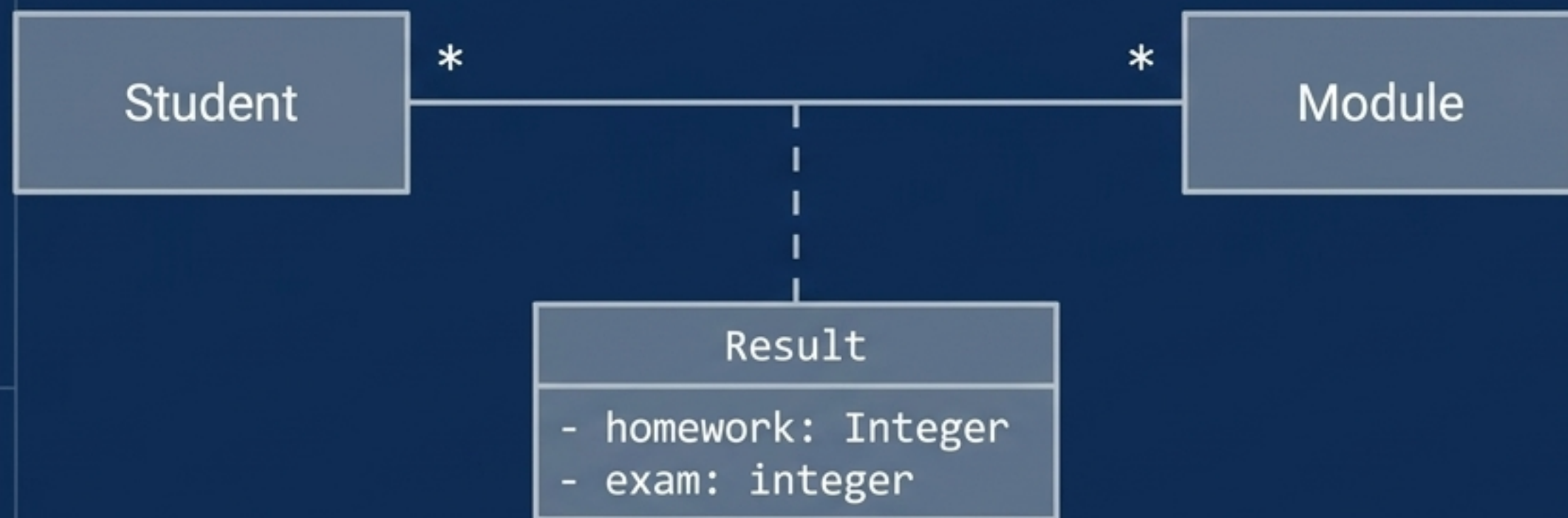


Java Implementation

```
1 public class Director {
2     private String name;
3     private String country;
4     // Professor's tip: 1..* means you need a List or Collection!
5     private Vector<Movie> movies;
6
7     public Director(String name, String country) {
8         this.name = name;
9         this.country = country;
10        this.movies = new Vector<Movie>();
11    }
12    public void addMovie(Movie m) {
13        movies.addElement(m);
14    }
15 }
```

✓ When you see a 1..* multiplicity in UML, it translates to an Array, Vector, or List in Java to hold multiple objects. (在 UML 看到一對多，Java 裡就要宣告陣列或集合！)

Advanced Tactic: The Association Class (關聯類別)



Sometimes the relationship itself holds data. A student's exam score doesn't belong *only* to the Student, nor *only* to the Module. It belongs to the *Study* association between them. (當關聯本身帶有屬性時, 例如學生的該科成績, 我們使用關聯類別。)

Exam Note: Read the prompt carefully. Many past papers explicitly say: "Note: You do not need to add association classes for this question." If you see this, keep it simple! (注意題目備註, 如果寫了不需要, 就不要畫蛇添足!)

The Professor's Final Checklist (交卷前最後檢查)

- Did I read the prompt? (Section A1 = Detailed, B2b = Initial).
- Are my **Class Names capitalized**? (Customer, not customer).
- Did I include **visibility symbols**? (+ for operations, - for attributes).
- Are my **multiplicities** correct? (Did I miss a $\theta..*$ or a 1?)
- Did I use **Inheritance** where asked? (Looking for "is a kind of").

Structure modeling is logical blueprinting. Follow the 7 steps, answer exactly what the section asks, and secure your marks. Good luck! (結構建模就是邏輯的藍圖。跟隨七步法，精準作答，祝各位考試順利！)