

4.7 Wrapper Classes

Name: _____ Class: _____ Date: _____

Total: 9 marks

Objective

Build the skills to answer exam questions on **wrapper classes**.

You must be able to:

- understand that a **wrapper class** 包装类 lets a primitive be treated as an object
- use the `Integer` and `Double` classes
- describe **autoboxing** 自动装箱 and **unboxing** 拆箱

1 Worked examples

Study these first. Each one shows the method for a question type used later.

■ Wrapper classes

A **wrapper class** lets a primitive value be treated as an **object**: `Integer` wraps an `int`, and `Double` wraps a `double`. This is needed because some structures (like `ArrayList`) store only objects.

■ Autoboxing and unboxing

```
Integer n = 5;    // autoboxing: int -> Integer
int m = n;       // unboxing: Integer -> int
```

2 Practice

2.1 State what a wrapper class does. [1]

2.2 Name the wrapper class for `int`. [1]

2.3 State the difference between autoboxing and unboxing. [2]

3 Exam-style questions

3.1 The wrapper class for `double` is [1]

- A `Double`
 - B `double`
 - C `Number`
 - D `Float`
-

3.2 Converting an `int` to an `Integer` automatically is [1]

- A unboxing
 - B autoboxing
 - C casting
 - D overloading
-

3.3 `Integer x = 7;`

(a) Name what happened to the 7. [1]

(b) Name the wrapper class used. [1]

(c) Name the reverse process (`Integer` to `int`). [1]

4 Go further

- work through the **4.7 Wrapper Classes** lesson on the **Learn** page;
- read the **ArrayList** section of the AP Computer Science A handout on the **Know** page.

Solutions

2.1 it lets a primitive value be treated as an object.

2.2 Integer.

2.3 autoboxing converts a primitive to its wrapper; unboxing converts a wrapper back to a primitive.

3.1 A.

3.2 B.

3.3 (a) it was autoboxed into an Integer. (b) Integer. (c) unboxing.