

## 3.2 Data Abstraction

Name: \_\_\_\_\_ Class: \_\_\_\_\_ Date: \_\_\_\_\_

**Total: 8 marks**

**KEY WORDS** abstraction 抽象 • list 列表 • data abstraction 数据抽象 • general 通用  
• modeling 建模

### Objective

Build the skills to answer exam questions on **data abstraction**.

**You must be able to:**

- explain how **data abstraction** 数据抽象 hides detail to manage complexity
- describe how a **list** 列表 groups many values under one name
- explain how using a list makes a program more **general** 通用
- relate data abstraction to real-world **modeling** 建模

### 1 Worked examples

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

#### ■ Data abstraction

**Data abstraction** hides detail so the programmer can manage complexity. A **list** is an abstract data type that groups many values under **one name**.

#### ■ More general

Storing scores as one list — `scores = [85, 90, 78]` — instead of three separate variables lets a single loop handle **any** number of values.

#### ■ Modeling

Like real-world modeling, abstraction simplifies detail to focus on the essentials.

### 2 Practice

**2.1** State what data abstraction does. [1]

**2.2** State how a list makes a program more general. [1]

**2.3** State one benefit of using a list instead of many separate variables. [1]

### 3 Exam-style questions

**3.1** A list is an example of [1]

- **A** a data abstraction
- **B** a syntax error
- **C** a Boolean
- **D** a byte

**3.2** Grouping many values under a single name is [1]

- **A** iteration
- **B** data abstraction
- **C** an event
- **D** compression

**3.3** A program must store 100 test scores.

(a) State whether to use 100 separate variables or one list. [1]

(b) State one benefit of the choice in (a). [1]

(c) Name the general idea of hiding detail to manage complexity. [1]

## Solutions

---

**2.1** it hides detail to manage complexity in a program.

**2.2** one list and one loop can handle any number of values.

**2.3** it is easier to store, name, and process many values together.

**3.1 A.**

**3.2 B.**

**3.3** (a) one list. (b) all scores can be processed with a single loop. (c) (data) abstraction.