

KK05 — Characteristics of data structures

Programming · Comprehensive exam study guide

STUDY DESIGN — YOU MUST KNOW

- one-dimensional arrays
- two-dimensional arrays
- records (varying data types, field index)

Data structures group related data. You must know how a **one-dimensional array**, **two-dimensional array** and **record** organise data, how each element is accessed, and when each is appropriate.

The three structures

Structure	Holds	Access	Good for
1-D array	A list of items of the same type	By a single index, e.g. <code>scores[0]</code>	A list of marks, a row of names.
2-D array	A grid/table of items of the same type	By row and column, e.g. <code>grid[2][3]</code>	A timetable, a seating grid, an image.
Record	A group of fields of varying types describing one thing	By field name/index, e.g. <code>student.name</code>	One student's id (int), name (string), paid (Boolean).

Arrays vs records

An **array** stores many items of the **same** type and is accessed by numeric index. A **record** stores several **different** fields about a single entity and is accessed by **field name** (or field index). Combining them (an array of records) is the common way to store many entities.

REMEMBER

Array index usually starts at **0**, so an array of length 5 has valid indexes 0–4. Accessing index 5 causes an 'index out of range' runtime error.

Key definitions to memorise

One-dimensional array	An ordered collection of elements of the same data type, accessed by a single index.
Two-dimensional array	A grid of elements of the same type, accessed by row and column indexes.
Record	A structure grouping related fields that may be of different data types, accessed by field name/index.
Field index	A number identifying a particular field within a record.

Exam technique

EXAM TIP

- When choosing a structure: same type, many items → array; different fields about one thing → record.
- Mention zero-based indexing if a question involves accessing elements.

COMMON MISTAKES

- Storing mixed data types in one array (use a record instead).
- Forgetting arrays are zero-indexed and accessing one past the end.

Quick revision checklist

- I can describe 1-D, 2-D arrays and records and how each is accessed.
- I can choose the right structure for a scenario and justify it.

Exam practice — questions & model answers

Q1. Distinguish between a one-dimensional array and a record.

Distinguish · 2 marks

Model answer

- A 1-D array stores multiple elements of the same type accessed by index (1);
- a record stores related fields that may be different types accessed by field name (1).

Q2. A program stores the daily temperatures for one week. Name the most suitable data structure and justify it.

Name/Justify · 2 marks

Model answer

- A one-dimensional array (1),
- because it holds seven values of the same type (numbers) accessed by index (1).

Q3. Describe a situation where a two-dimensional array is more suitable than a one-dimensional array.

Describe · 2 marks

Model answer

- When data forms a grid/table, e.g. a class timetable with rows (days) and columns (periods) (1),
- a 2-D array stores each cell and is accessed by row and column (1).