Arrays and records

Algorithms and programming (Paper 2) · revision notes

Arrays store many items of the same data type under one name, using an index to pick out each item; 2D arrays store data in rows and columns. Records group related fields about one item, which can have different data types. Expect code that loops through arrays and questions about records and fields in a table.

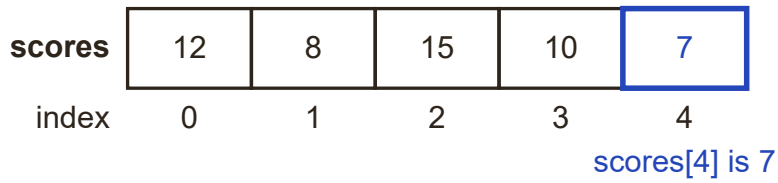
Key facts

- Array: a fixed number of items of the same data type, under one identifier
- Access: names[2]; assign: names[2] = "Jo"
- 2D array: two indexes, e.g. grid[2, 3]; use nested loops
- Database table: row = record, column = field
- Indexes start at 0: array names[5] has names[0] to names[4]
- Loop through an array: for i = 0 to length − 1
- Record: related fields about one item; fields can be different data types

Notes

1D arrays

- An **array** stores a fixed number of items of the same data type under one identifier. It is a **static** structure: its size is fixed and cannot change while the program runs.
- Each item (element) is picked out by its **index**. Indexes start at **0**.
- `array scores[5]` declares an array of 5 elements: `scores[0]` to `scores[4]`.



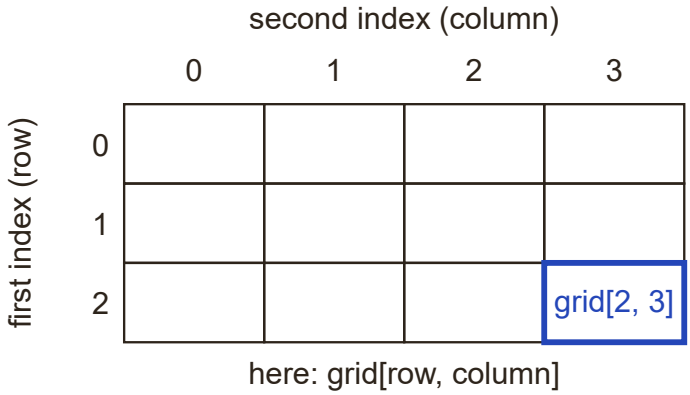
A 1D array: 5 elements, indexes 0 to 4

- `scores[0] = 12` stores a value; `print(scores[4])` outputs the last element.
- Loop through every element:

```
total = 0
for i = 0 to 4
   total = total + scores[i]
next i
```

2D arrays

- A **2D array** is like a table, with two indexes: `array grid[3, 4]` holds 3 × 4 = 12 elements. A 2D array can emulate a database table: one row per record, one column per field.
- An element is accessed with both indexes, for example `grid[2, 3]`. The question tells you which index is the row and which is the column.



A 2D array: 3 × 4 = 12 elements, each picked out by two indexes

- To visit every element, use two nested loops: one for each index.

Records

- A **record** is a collection of related data items, called **fields**, about one thing.
- The fields can have different data types. For example, a record for a film could have Title (string), Year (integer), Rating (real) and Subtitled (Boolean).
- In a database table, each row is a record and each column is a field.
- Difference: all elements of an array have the same data type, but the fields of a record can have different data types.

How to answer each type of question

State values or indexes in an array 1 mark each **Grade 4**

1. Number the elements from 0.
2. Read off the element at the index given, or give the index of the value asked for.

Example. The array `temps` holds the values 14, 17, 9, 21, 12, in that order.
(a) State the value of temps[3]. (b) State the index of the value 9. (c) State the value of temps[0] + temps[4]. [3]

Model answer.

(a) 21 (1)
(b) 2 (1)
(c) 26 (1)

Write code to process a 1D array

4 marks | Grade 6

1. Set any totals or counters to 0.
2. Loop from index 0 to the last index.
3. Inside the loop, update the total or counter using array[i].
4. Output the results after the loop.

Example. The array `temps` holds 7 daily temperatures as real numbers. Write code that outputs the mean temperature and the number of days above 20. [4]

Model answer.

```
total = 0
hot = 0
for i = 0 to 6
   total = total + temps[i]
   if temps[i] > 20 then
      hot = hot + 1
   endif
next i
print(total / 7)
print(hot)
```

Loop through all 7 elements, index 0 to 6 (1); running total of the temperatures (1); counts the values above 20 (1); outputs the mean and the count after the loop (1).

Write code to process a 2D array

4 marks | Grade 8

1. Use an outer loop for the first index and an inner loop for the second.
2. Reset a total at the start of the outer loop if you need one per row.
3. Output inside the outer loop but after the inner loop for a result per row.

Example. The 2D array `sales` stores the sales of 4 shops over 3 months. The first index (0 to 3) is the shop and the second index (0 to 2) is the month. Write code that outputs the total sales for each shop. [4]

Model answer.

```
for shop = 0 to 3
   total = 0
   for month = 0 to 2
      total = total + sales[shop, month]
   next month
   print(total)
next shop
```

Outer loop for the 4 shops (1); total reset to 0 for each shop (1); inner loop for the 3 months adding sales[shop, month] (1); outputs each shop's total after the inner loop (1).

Explain records and fields 2 marks **Grade 5**

1. Say that a record groups related data about one item.
2. Say that the fields can have different data types, unlike an array.

Example. A club stores each member's ID, name, age and whether they have paid. Explain why a record is a suitable way to store this data. [2]

Model answer. A record groups all the related data about one member together (1). Its fields can have different data types, such as a string for the name, an integer for the age and a Boolean for whether they have paid, which an array cannot do (1).

Don't lose marks

- Starting at index 1, or looping to n instead of n − 1, which goes past the end of the array.
- Resetting a total inside the inner loop instead of the outer loop (or not resetting it at all).
- Saying an array can hold different data types.
- Mixing up records (rows) and fields (columns).
- Using the wrong index order in a 2D array: check what the question says each index means.

More tips

Memory tricks

- Array of size n: indexes 0 to n − 1. 'Size 5, last index 4.'
- Nested loops for 2D arrays: the inner loop finishes all its passes for each single pass of the outer loop.
- Record = one row about one thing; field = one column, one piece of data.

Exam technique

- Write the index numbers above an array given in a question before answering.
- Use the array name and index meanings exactly as the question defines them.
- When asked for 'each' shop, row or player, you almost always need nested loops with a reset.

What each grade needs

What you need to be able to do, from the first marks up to the top grade.

Grade 4 **Access an array element by its index**
Remember the first element is at index 0, so names[2] is the third item.

Grade 5 **Declare and assign values in an array**
Use array scores[5] to declare it, then scores[0] = 12 to assign a value.

Grade 5 **Explain what a record is**
A group of related fields about one item, which can be of different data types.

Grade 6 **Loop through a 1D array**
Use for i = 0 to length − 1 to total, count or search the items.

Grade 7

Use a 2D array

Access an element with two indexes, such as grid[2, 3].

Grade 8

Process a 2D array with nested loops

Use an outer loop for one index and an inner loop for the other.

Free notes, Full packs, practice papers and mock exams at chhetriacademy.com. Written for this site; not taken from past papers. Stuck? 1-to-1 tutoring with a free 30-minute trial lesson.