

# Arrays: Data Structure & Pre-Release Materials

## Arrays

Arrays are data structure used to store multiple data items of same data type under one identifier name.

Arrays are considered to be fixed-length structures of elements of identical data type, accessible by consecutive index (subscript) numbers. It is good practice to explicitly state what the lower bound of the array (i.e. the index of the first element) is because this defaults to either 0 or 1 in different systems. **Generally, a lower bound of 1 will be used.**

Square brackets are used to indicate the array indices.

Each element in the array is identified using its **subscript** or **index number**. The largest and smallest index numbers are called the *upper bound* and *lower bound* of the array.

### Example

```
StudentName[1:30]
```

For illustration, let's take array declaration to store marks of 10 students.

```
Marks[1:10]
```

After storing values in array

|          |    |    |    |    |    |    |    |    |    |    |
|----------|----|----|----|----|----|----|----|----|----|----|
| elements | 35 | 33 | 42 | 10 | 14 | 19 | 27 | 44 | 26 | 31 |
| index    | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10 |

Size :10

As per the above illustration, following are the important points to be considered.

- Index starts with 1.
- Array length is 10 which means it can store 10 elements.
- Each element can be accessed via its index. For example, we can fetch an element at index 6 as 19.

### The terms associated with Arrays

**Name:** The identifier of the array is called Array Name. E.g. StudentName[]

**Element:** Each data item stored in array is called element. Array can store only single types of elements.

**Size:** The number elements the array can store. E.g. StudentName[1:30] can store 30 names while StudentName[30] can store 31 names as by default it is 0 to 30.

**Index:** The position of each element is referred as Index Number. Index of Abdullah in array example is 1.

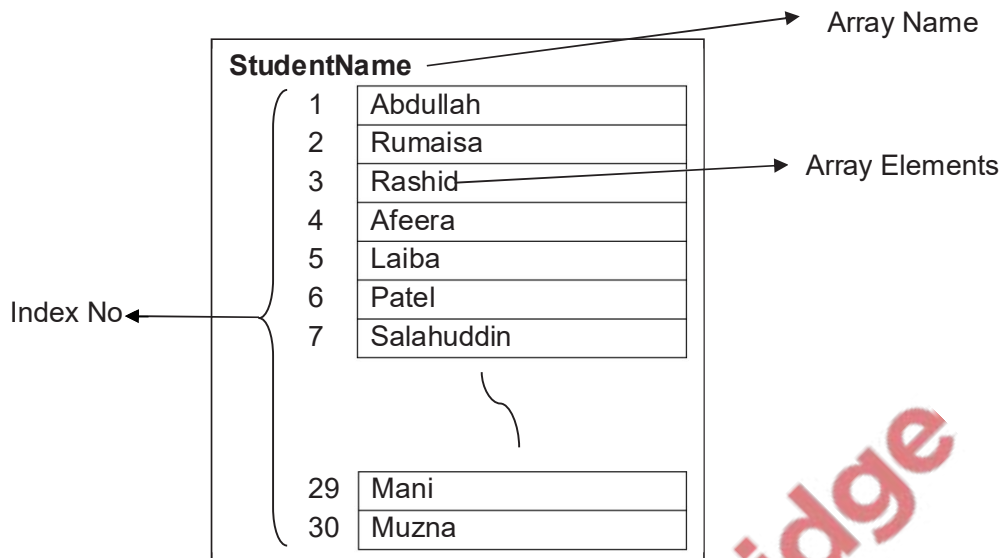
**Type:** Data type of all elements in a single array have same data types.

**Dimension:** Dimension is the organisational structure of array. It may be 1D that has single column or 2D that have multiple columns.

### Example

```
DECLARE StudentName[1:30] : STRING
```

```
StudentName[1] ← "Abdullah"
```



### Declaring an array

It is important to declare the arrays before assigning values in it so that the program can reserve that amount of space in its memory; otherwise, there may not be enough space when the program uses the data.

Declaration consists of telling the computer program:

- the identifier name of the array
- the sort of data that is going to be stored in the array, i.e. its data type
- How many items of data are going to be stored, so that it knows how much space to reserve.

Different programming languages have different statements for initialising the array but they all do the same thing. In Visual Basic, the statement is:

```
Dim Name(20) As String
```

This Dim statement declares:

- the identifier name: Name
- the upper bound: 20
- the data type: String.

***The upper bound of 20 specifies that there can be a maximum of 21 data items, since Visual Basic starts with a subscript of zero. We do not have to fill the array; the upper bound of 20 indicates the maximum size.***

The array that has been described in one dimension array so far is really only a list of single data items. It is possible to have an array which can be visualised as a two-dimensional table with rows and columns and a data value in each cell.

**Reading data into an array**

To assign data values to the elements of the array, we do this with assignment statements such as:

    Name(6) = "Patel"

    This places the string "Patel" at index position 6 in the array.

Similarly, the following statement places the string "Rashid" at index position 3 in the array.

    Name(19) = "Mani"

**Quick Revision Questions**

Q 17.1) Explain the following terms regarding arrays:

Array: .....

.....

Size of Array: .....

.....

Element: .....

.....

Index: .....

.....

Type: .....

.....

Dimension: .....

.....

Q 17.2) Explain with the help of examples when arrays are used in programming.

.....

.....

.....

.....

.....

.....

.....

.....

Q 17.3) Declare arrays to Explain with the help of examples when arrays are used in programming

.

a) Declare arrays to store name of 30 students

.....

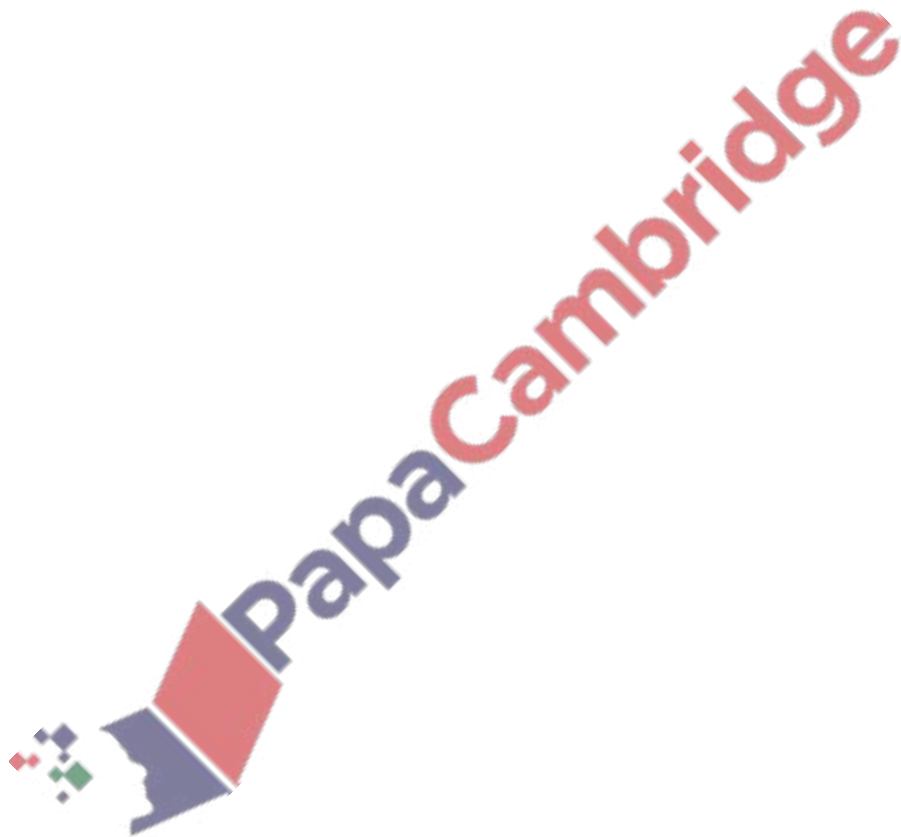
.....

b) Declare arrays to store basic pay of 50 Employees.

.....  
.....  
c) Declare arrays to input and store status of 50 employee that they are permanent or not.

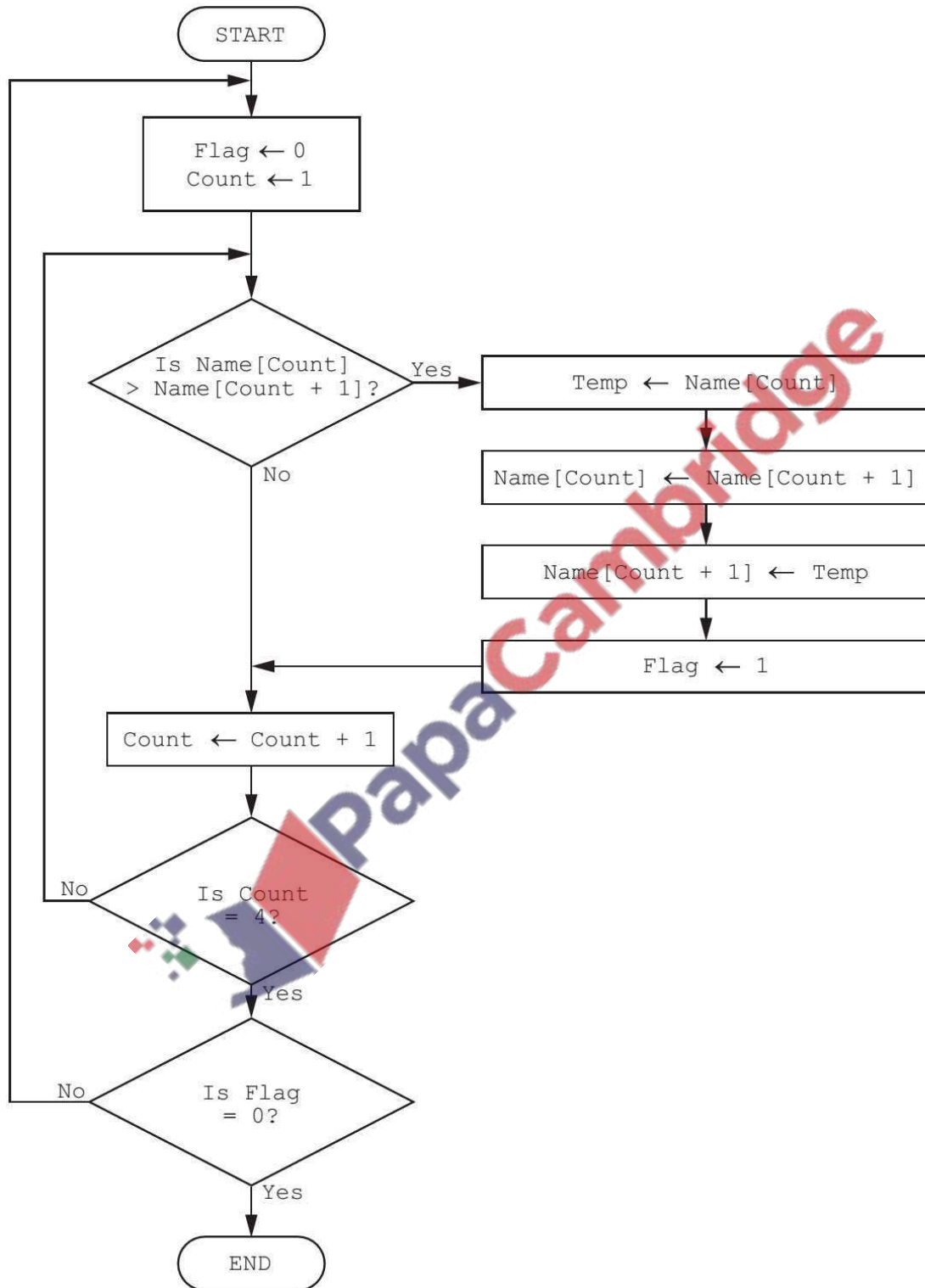
.....  
.....  
**Q 17.4) Draw a flowchart that**

- Inputs name of 10 students in a class and store them in one dimension array
- Display list of names of students



Past paper flowchart for same type of question in Winter 2017 P21 Q5

The flowchart below represents a program routine.



(a) The array used in the flowchart contains the following data:

|                |                |                |                |
|----------------|----------------|----------------|----------------|
| <b>Name[1]</b> | <b>Name[2]</b> | <b>Name[3]</b> | <b>Name[4]</b> |
|----------------|----------------|----------------|----------------|

|       |      |     |      |
|-------|------|-----|------|
| Jamal | Amir | Eve | Tara |
|-------|------|-----|------|

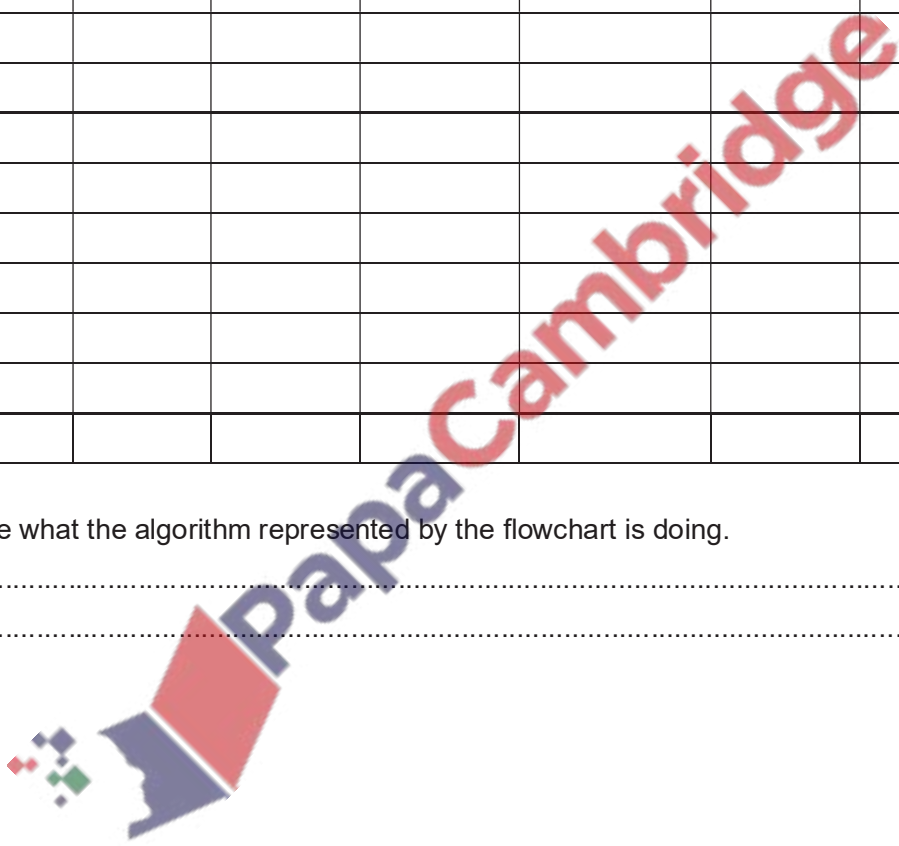
Complete the trace table using the data given in the array.

[5]

| Flag | Count | Name[1] | Name[2] | Name[3] | Name[4] | Temp |
|------|-------|---------|---------|---------|---------|------|
|      |       | Jamal   | Amir    | Eve     | Tara    |      |
|      |       |         |         |         |         |      |
|      |       |         |         |         |         |      |
|      |       |         |         |         |         |      |
|      |       |         |         |         |         |      |
|      |       |         |         |         |         |      |
|      |       |         |         |         |         |      |
|      |       |         |         |         |         |      |
|      |       |         |         |         |         |      |
|      |       |         |         |         |         |      |
|      |       |         |         |         |         |      |
|      |       |         |         |         |         |      |
|      |       |         |         |         |         |      |

(b) Describe what the algorithm represented by the flowchart is doing.

.....  
 ..... [2]





```
i = 1
REPEAT
    Coins(i) = 0
    i = i + 1
UNTIL i = 10
```

(i) State what is meant by a logic error.

..... [1]

(ii) Explain why the algorithm above contains a logic error.

..... [2]

- (i) •The program is written to do something other than what the programmer intended
- (ii) •It will only reset the first 9 elements / will not reset the 10<sup>th</sup> element
- After setting Coins(9) = 0, i will become 10...
- ... and the loop will stop
- It should be UNTIL i > 10 / or other working correction

**Q 17.8 Summer 2015 P22**

5 (a) Write an algorithm, using pseudo code and a FOR ... TO ... NEXT loop structure, to input 1000 numbers into an array.

..... [2]

(b) Rewrite your algorithm using another loop structure.

..... [4]

Examiner's comments on Question 5

- (a) Most candidates attempted the loop structure, better candidates also showed the skill of being able to use the loop counter as the array index. Some candidates misread the question and incorrectly provided program code rather than pseudo code.
- (b) Better candidates correctly used REPEAT ... UNTIL or WHILE ... DO ... ENDWHILE structures.



The most challenging aspect was the correct management of the loop counter.

**Q 17.9 Summer 2016 P22**

5 A programmer writes a program to store a patient's temperature every hour for a day. State the data structure that would be most suitable to use and give the reason for your choice.

Data structure .....  
Reason .....  
.....[2]

5 (a) Describe the purpose of each statement in this algorithm.

```
FOR I ← 1 to 300
  INPUT Name[I]
NEXT I
```

.....  
.....  
.....  
.....  
.....[2]

(b) Identify, using pseudocode, another loop structure that the algorithm in part (a) could have used.

.....  
.....[1]

(c) Write an algorithm, using pseudocode, to input a number between 0 and 100 inclusive. The algorithm should prompt for the input and output an error message if the number is outside this range.

.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....[3]

**Q 17.10 Winter 2017 P22**

3 The following diagram shows **four** data structures and **four** descriptions. [3]

Draw a line to connect each data structure to the correct description.

**Data structure**

**Description**

Constant

A collection of related data

Array

A value that can change whilst a program is running

Table

A value that never changes whilst a program is running

Variable

A series of elements of the same data type

A restaurant table will have its data stored in its own booking record. Alessio decides to use an array of records.

Write **program code** to declare the array TableBookings for the 12 table records.

Programming language.....  
Code .....  
.....[1]

**Summer 2016 P21 &P23**

(ii) The swimming club has 50 members.

State the data structure that would be most suitable to use and give a reason for your choice.

Data structure .....  
Reason .....  
.....[2]

