G.C.E. O/L Grade 11 ICT · Lesson 1

1.9 One-Dimensional Arrays

Syllabus 10.9. Imagine storing the marks of 40 students in 40 separate variables – Mark1, Mark2 ... Mark40. Very messy! An **array** stores many values of the **same data type** under **one name**. Each value is an **element**, and we reach it with its **index** (position number), like Marks[3]. Arrays are kept in **contiguous** (side-by-side) memory, so any element can be reached **directly** (random access).

1. Why do we need arrays?

- **One name** for many values – easy to read.
- Use a **loop** to process all elements with a few lines.
- Easy to find the total, average, highest and to search or sort.
- To change 5 students to 40, change one number only.

2. Index, elements and memory

Think of a row of numbered lockers in a school corridor. All lockers are **next to each other** (contiguous), and you can open locker 4 **directly** without opening 1, 2 and 3 first (random access). The array Marks with 5 elements:

Pascal index	1	2	3	4	5
Python index	0	1	2	3	4
Value	65	80	45	90	70

💡 **Index starts:** in Pascal we choose it (array[1..5] starts at 1). In **Python** a list always starts at **0**, so the first mark is marks[0] and the fifth is marks[4]. Pseudocode in O/L normally starts at 1.

3. Declare, assign and access

Action	Pascal	Python
Declare (5 integers)	var Marks: array[1..5] of integer;	marks = [0, 0, 0, 0, 0]
Assign a value	Marks[3] := 45;	marks[2] = 45
Access (read) a value	writeln(Marks[3]);	print(marks[2])
Fill with a loop	for i := 1 to 5 do readln(Marks[i]);	for i in range(5): marks[i] = int(input())

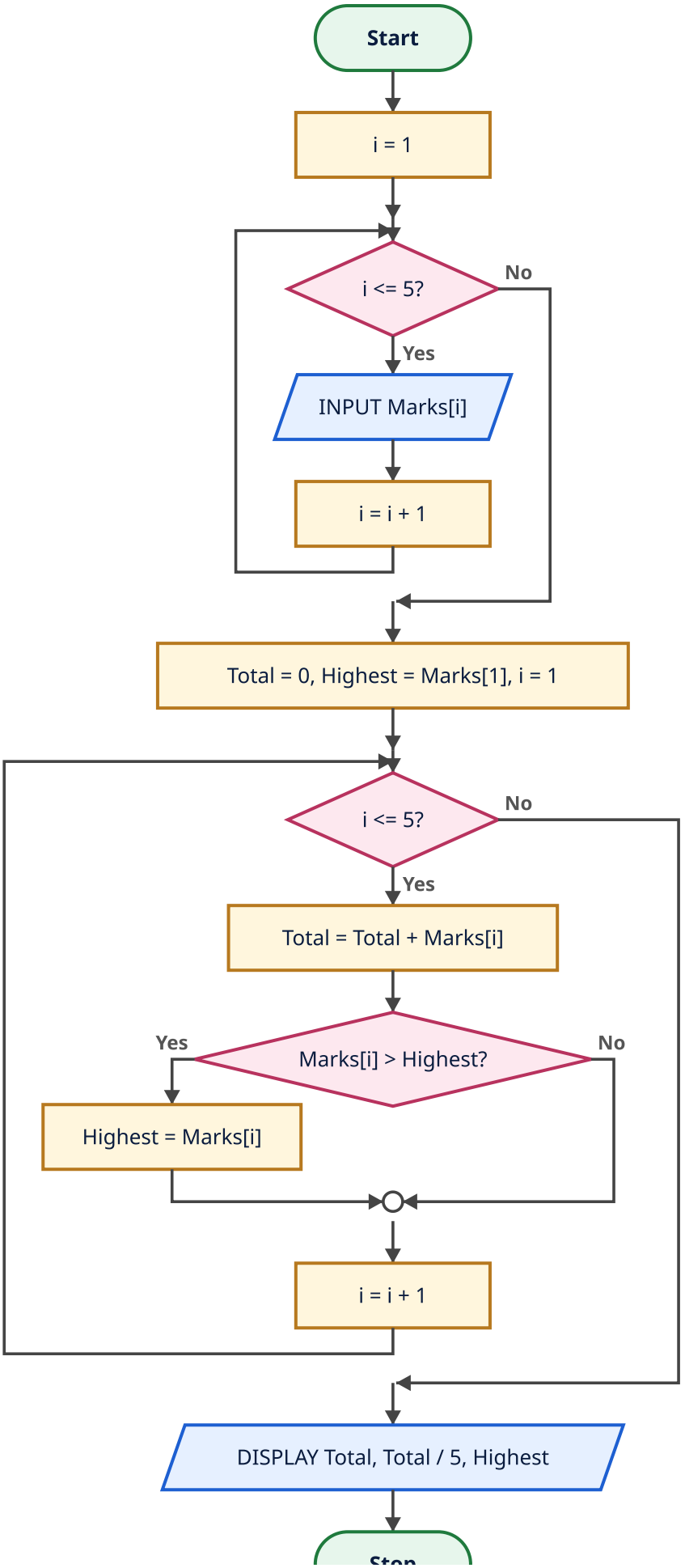
4. Example: total, average and highest

Pseudocode

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

```
BEGIN
  FOR i = 1 TO 5
    INPUT Marks[i]
  ENDFOR
  Total = 0
  Highest = Marks[1]
  FOR i = 1 TO 5
    Total = Total + Marks[i]
    IF Marks[i] > Highest THEN
      Highest = Marks[i]
    ENDIF
  ENDFOR
  DISPLAY Total, Total / 5, Highest
END
```

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved



© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

Stop

Flowchart – Array: store 5 marks, then total, average and highest

Array: store 5 marks, then total, average and highest

Pascal

```
program MarksArray;
var
  marks: array[1..5] of integer;
  i, total, highest: integer;
begin
  for i := 1 to 5 do
  begin
    write('Enter mark ', i, ': ');
    readln(marks[i]);
  end;
  total := 0;
  highest := marks[1];
  for i := 1 to 5 do
  begin
    total := total + marks[i];
    if marks[i] > highest then
      highest := marks[i];
  end;
  writeln('Total = ', total);
  writeln('Average = ', total / 5:0:2);
  writeln('Highest = ', highest);
end.
```

Python

```
marks = [0, 0, 0, 0, 0]          # 5 elements, index 0 to 4
for i in range(5):
    marks[i] = int(input(f"Enter mark {i + 1}: "))
total = 0
highest = marks[0]
for i in range(5):
    total = total + marks[i]
    if marks[i] > highest:
        highest = marks[i]
print("Total =", total)
print(f"Average = {total / 5:.2f}")
print("Highest =", highest)
```

Sample run

```
Enter mark 1: 65
Enter mark 2: 80
Enter mark 3: 45
Enter mark 4: 90
Enter mark 5: 70
Total = 350
Average = 70.00
Highest = 90
```

i	Marks[i]	Total	Marks[i] > Highest ?	Highest

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

1	65	65	No	65
2	80	145	Yes	80
3	45	190	No	80
4	90	280	Yes	90
5	70	350	No	90

Total = 350 and Highest = 90, the same as the real program output. Average = 350 ÷ 5 = 70.00.

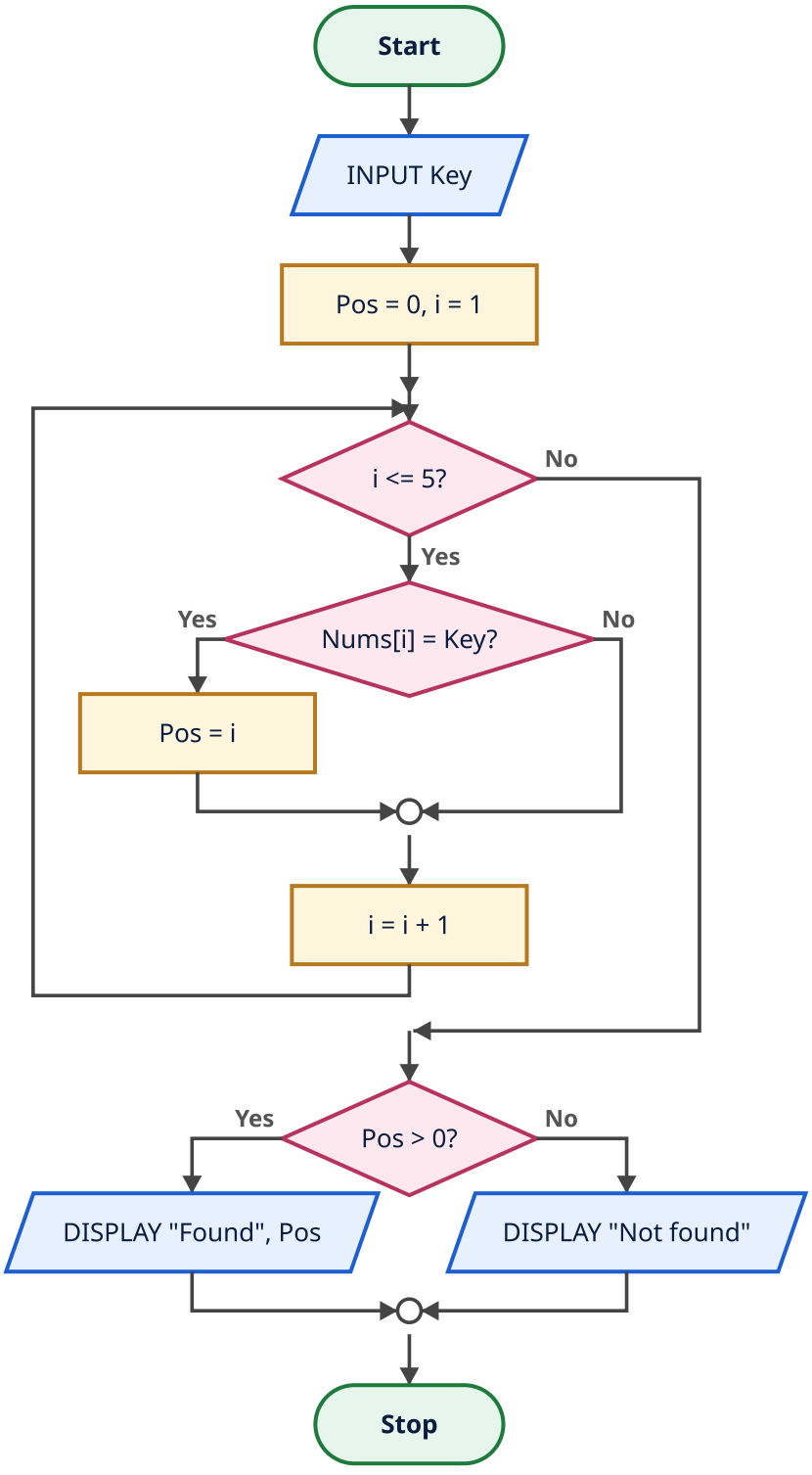
5. Example: searching an array (linear search)

The array Nums holds 12, 45, 7, 30, 18. To find a number, check each element **from the first to the last**. This is called a **linear search**.

Pseudocode

```
BEGIN
  Nums = [12, 45, 7, 30, 18]
  INPUT Key
  Pos = 0
  FOR i = 1 TO 5
    IF Nums[i] = Key THEN
      Pos = i
    ENDIF
  ENDFOR
  IF Pos > 0 THEN
    DISPLAY "Found at position ", Pos
  ELSE
    DISPLAY "Not found"
  ENDIF
END
```

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved



Flowchart – Array: search for a number

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

Array: search for a number

Pascal

```
program SearchArray;
var
  nums: array[1..5] of integer;
  i, key, pos: integer;
begin
  nums[1] := 12;
  nums[2] := 45;
  nums[3] := 7;
  nums[4] := 30;
  nums[5] := 18;
  write('Enter number to find: ');
  readln(key);
  pos := 0;
  for i := 1 to 5 do
    if nums[i] = key then
      pos := i;
  if pos > 0 then
    writeln('Found at position ', pos)
  else
    writeln('Not found');
end.
```

Python

```
nums = [12, 45, 7, 30, 18]
key = int(input("Enter number to find: "))
pos = 0
for i in range(5):
    if nums[i] == key:
        pos = i + 1      # position counted from 1
if pos > 0:
    print("Found at position", pos)
else:
    print("Not found")
```

Sample run 1

```
Enter number to find: 30
Found at position 4
```

Sample run 2

```
Enter number to find: 50
Not found
```

i	Nums[i]	Nums[i] = 30 ?	Pos
1	12	No	0
2	45	No	0
3	7	No	0
4	30	Yes	4
5	18	No	4

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

6. Finding a position

Temperatures (°C) in Colombo for a week are stored in an array. The program keeps the **index** of the highest value, not just the value, so it can tell us **which day** was hottest.

Position of the highest temperature

Pascal

```
program HottestDay;
var
  t: array[1..7] of integer;
  i, best: integer;
begin
  t[1] := 31; t[2] := 33; t[3] := 30; t[4] := 34;
  t[5] := 32; t[6] := 29; t[7] := 33;
  best := 1;
  for i := 2 to 7 do
    if t[i] > t[best] then
      best := i;
  writeln('Day ', best, ': ', t[best]);
end.
```

Python

```
t = [31, 33, 30, 34, 32, 29, 33]
best = 0
for i in range(1, 7):
    if t[i] > t[best]:
        best = i
print("Day " + str(best + 1) + ":", t[best])
```

Sample run

```
Day 4: 34
```

Key Words

Array – A collection of values of the same type under one name.

One-dimensional array – An array that is a single row (list) of elements.

Element – One value stored in an array.

Index – The position number of an element, e.g. 3 in Marks[3].

Contiguous memory – Elements stored side by side in memory.

Random (direct) access – Reaching any element at once using its index.

Declare – Creating the array with its name, size and type.

Assign – Storing a value in an element, Marks[3] := 45.

Access – Reading the value of an element.

List (Python) – Python's array-like type; index starts at 0.

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

Linear search – Checking elements one by one from the start.

Array size – The number of elements in an array.

Sample Q&A

1. What is an array?

> **Answer:** A group of values of the same data type stored under one name.

2. What is an index?

> **Answer:** The position number used to reach an element.

3. What does “random access” mean for arrays?

> **Answer:** Any element can be reached directly with its index.

4. Declare an integer array of 5 elements in Pascal.

> **Answer:** var Marks: array[1..5] of integer;

5. In Python, what is the index of the first element?

> **Answer:** 0.

6. For marks 65, 80, 45, 90, 70 what is the highest?

> **Answer:** 90.

MCQ

1. An array stores many values of...

A. the same data type
B. only text
C. only Boolean
D. different data types always

2. One value inside an array is called an...

A. operator
B. element
C. index
D. identifier

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

3. The position number of an element is its…

A. index
B. value
C. type
D. size

4. Array elements are stored in…

A. separate files
B. random places on the disk
C. contiguous memory
D. the printer

5. Reaching Marks[4] without reading Marks[1..3] is called…

A. nesting
B. sequential access
C. random (direct) access
D. compiling

6. Correct Pascal declaration of an array of 5 integers:

A. A = [5]
B. array A of 5;
C. var A: integer[5];
D. var A: array[1..5] of integer;

7. In Python, the first element of marks is…

A. marks(1)
B. marks[0]
C. marks[1]
D. marks{0}

8. A Python list of 5 elements has its last index…

A. 4
B. 1
C. 6
D. 5

9. Marks = 65, 80, 45, 90, 70 (Pascal). Marks[3] is…

A. 90
B. 45
C. 3
D. 80

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

10. The total of 65, 80, 45, 90 and 70 is...

A. 90
B. 340
C. 350
D. 70

11. Nums = 12, 45, 7, 30, 18. Searching for 30 gives...

A. Found at position 4
B. Not found
C. Found at position 3
D. Found at position 30

12. Searching for 50 in the same array gives...

A. Found at position 1
B. An error
C. Not found
D. Found at position 5

13. Checking elements one by one from the start is a...

A. random search
B. nested IF
C. constant
D. linear search

14. Which statement assigns 45 to the third element in Python?

A. 45 = marks[2]
B. marks = 45
C. marks[3] = 45
D. marks[2] = 45

15. Temperatures 31, 33, 30, 34, 32, 29, 33. The hottest day is...

A. Day 6: 29
B. Day 4: 34
C. Day 7: 33
D. Day 2: 33

Structured Questions

1. (i) What is a one-dimensional array? (1 mark)
(ii) Give two advantages of using an array instead of separate variables. (2 marks) *(3 marks)*

> (i) A single row of elements of the same type under one name. (ii) One name for many values; can process all with a loop; easy to change size (any two).

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

2. Explain the terms (a) index (b) contiguous memory (c) random access. (3 marks) *(3 marks)*

> (a) Position number of an element. (b) Elements are stored side by side in memory. (c) Any element can be reached directly using its index.

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

3. Write a program (Pascal or Python) to read 5 marks into an array and display the total, average and highest mark. (6 marks) *(6 marks)*

Declare array (1); loop to read (1); total = 0, highest = first element (1); loop adding and comparing (2); display total, average, highest (1).

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

Array: store 5 marks, then total, average and highest

Pascal

```
program MarksArray;
var
  marks: array[1..5] of integer;
  i, total, highest: integer;
begin
  for i := 1 to 5 do
  begin
    write('Enter mark ', i, ': ');
    readln(marks[i]);
  end;
  total := 0;
  highest := marks[1];
  for i := 1 to 5 do
  begin
    total := total + marks[i];
    if marks[i] > highest then
      highest := marks[i];
  end;
  writeln('Total = ', total);
  writeln('Average = ', total / 5:0:2);
  writeln('Highest = ', highest);
end.
```

Python

```
marks = [0, 0, 0, 0, 0]          # 5 elements, index 0 to 4
for i in range(5):
    marks[i] = int(input(f"Enter mark {i + 1}: "))
total = 0
highest = marks[0]
for i in range(5):
    total = total + marks[i]
    if marks[i] > highest:
        highest = marks[i]
print("Total =", total)
print(f"Average = {total / 5:.2f}")
print("Highest =", highest)
```

Sample run

```
Enter mark 1: 65
Enter mark 2: 80
Enter mark 3: 45
Enter mark 4: 90
Enter mark 5: 70
Total = 350
Average = 70.00
Highest = 90
```

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

4. Nums = 12, 45, 7, 30, 18. Complete a trace table for a linear search for 30 and give the output. (4 marks) *(4 marks)*

> i = 1…5 with Nums[i] = 30? No, No, No, Yes, No; Pos = 0, 0, 0, 4, 4 (3); output Found at position 4 (1).

i	Nums[i]	Nums[i] = 30 ?	Pos
1	12	No	0
2	45	No	0
3	7	No	0
4	30	Yes	4
5	18	No	4

5. Write Pascal and Python statements to (a) store 45 in the third element of Marks (b) display the first element. (4 marks) *(4 marks)*

> (a) Pascal Marks[3] := 45; Python marks[2] = 45. (b) Pascal writeln(Marks[1]); Python print(marks[0]) (1 each).

Summary

- Array: many values, same type, one name.
- Element = one value; index = its position.
- Contiguous memory → random (direct) access.
- Pascal: array[1..5] of integer.
- Python list index starts at 0.
- Use a loop to fill and process an array.
- Total, average, highest with one loop.
- Linear search checks elements one by one.

MCQ answer key – 1.9

1. **A** – Same type.
2. **B** – Element.
3. **A** – Index.
4. **C** – Side by side.
5. **C** – Direct.
6. **D** – Pascal syntax.
7. **B** – Index from 0.
8. **A** – 0 to 4.
9. **B** – Third element.
10. **C** – Sum.

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

11. **A** – Fourth element.
12. **C** – Not in the array.
13. **D** – Linear.
14. **D** – Index 2 = third.
15. **B** – 34 is 4th.

© 2026 schoolict.net (Ruwan Mangala Suraweera). All rights reserved.