G.C.E. O/L ICT · 11 ශ්‍රේණිය · පාඩම 1

1.9 ඒක-මාන අරා

විෂය නිර්දේශය 10.9. සිසුන් 40 දෙනෙකුගේ ලකුණු වෙනම විචල්‍ය 40 ක – Mark1, Mark2 … Mark40 – ගබඩා කිරීම ගැන සිතන්න. ඉතා අවුල්! **අරාවක්** (array) **එකම දත්ත වර්ගයේ** අගයන් බොහොමයක් **එක් නමක්** යටතේ ගබඩා කරයි. සෑම අගයක්ම **අවයවයකි** (element), අප එයට එහි **දර්ශකය** (index – ස්ථාන අංකය) මගින් ළඟා වෙමු, Marks[3] වැනි. අරා **එක ළඟ පිහිටි** (contiguous) මතකයේ තබන නිසා ඕනෑම අවයවයකට **කෙළින්ම** ළඟා විය හැක (random access).

1. අරා අවශ්‍ය ඇයි?

- බොහෝ අගයන්ට **එක් නමක්** – කියවීමට පහසුයි.
- පේළි කිහිපයකින් සියලු අවයව සැකසීමට **ලූපයක්** භාවිත කරන්න.
- එකතුව, සාමාන්‍යය, ඉහළම අගය සොයා ගැනීමට, සෙවීමට හෝ පිළිවෙළට සැකසීමට පහසුයි.
- සිසුන් 5 සිට 40 දක්වා වෙනස් කිරීමට එක් අංකයක් පමණක් වෙනස් කරන්න.

2. දර්ශකය, අවයව සහ මතකය

පාසල් කොරිඩෝවක අංක කළ ලොකර් පේළියක් ගැන සිතන්න. සියලු ලොකර් **එක ළඟ** (contiguous) ඇති අතර 1, 2, 3 මුලින් නොවිවෘත කර ලොකර් 4 **කෙළින්ම** විවෘත කළ හැක (random access). අවයව 5ක් ඇති Marks අරාව:

Pascal දර්ශකය	**1**	**2**	**3**	**4**	**5**
Python දර්ශකය	0	1	2	3	4
අගය	65	80	45	90	70

💡 **දර්ශකය ආරම්භය:** Pascal හි අප එය තෝරමු (array[1..5] 1 න් ආරම්භ වේ). **Python** හි list එකක් සැමවිටම **0** න් ආරම්භ වේ, එබැවින් පළමු ලකුණ marks[0] ද පස්වැන්න marks[4] ද වේ. O/L ව්‍යාජ කේතය සාමාන්‍යයෙන් 1 න් ආරම්භ වේ.

3. ප්‍රකාශ කිරීම, පැවරීම සහ ප්‍රවේශය

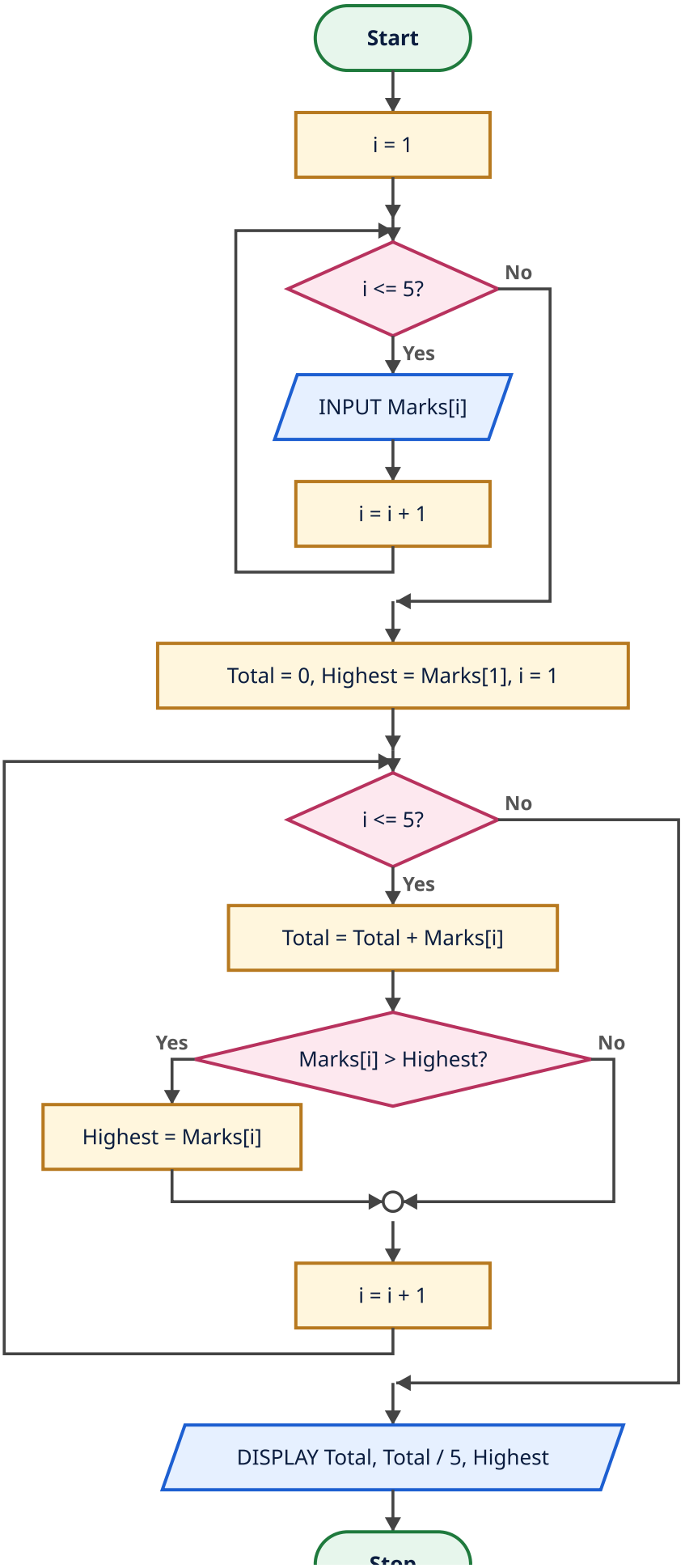
ක්‍රියාව	**Pascal**	**Python**
ප්‍රකාශ කිරීම (පූර්ණ සංඛ්‍යා 5)	var Marks: array[1..5] of integer;	marks = [0, 0, 0, 0, 0]
අගයක් පැවරීම	Marks[3] := 45;	marks[2] = 45
අගයක් කියවීම	writeln(Marks[3]);	print(marks[2])

ලූපයකින් පිරවීම	for i := 1 to 5 do readln(Marks[i]);	for i in range(5): marks[i] = int(input())

4. උදාහරණය: එකතුව, සාමාන්‍යය සහ ඉහළම

ව්‍යාජ කේතය (Pseudocode)

```
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
```


Start
i = 1
i <= 5?
No
Yes
INPUT Marks[i]
i = i + 1
Total = 0, Highest = Marks[1], i = 1
i <= 5?
No
Yes
Total = Total + Marks[i]
Yes
Marks[i] > Highest?
No
Highest = Marks[i]
i = i + 1
DISPLAY Total, Total / 5, Highest

Stop

ගැලීම් සටහන – අරාව: ලකුණු 5ක් ගබඩා කර එකතුව, සාමාන්‍යය, ඉහළම

අරාව: ලකුණු 5ක් ගබඩා කර එකතුව, සාමාන්‍යය, ඉහළම

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)
```

නියැදි ධාවනය

```
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

1	65	65	නැත	65
2	80	145	ඔව්	80
3	45	190	නැත	80
4	90	280	ඔව්	90
5	70	350	නැත	90

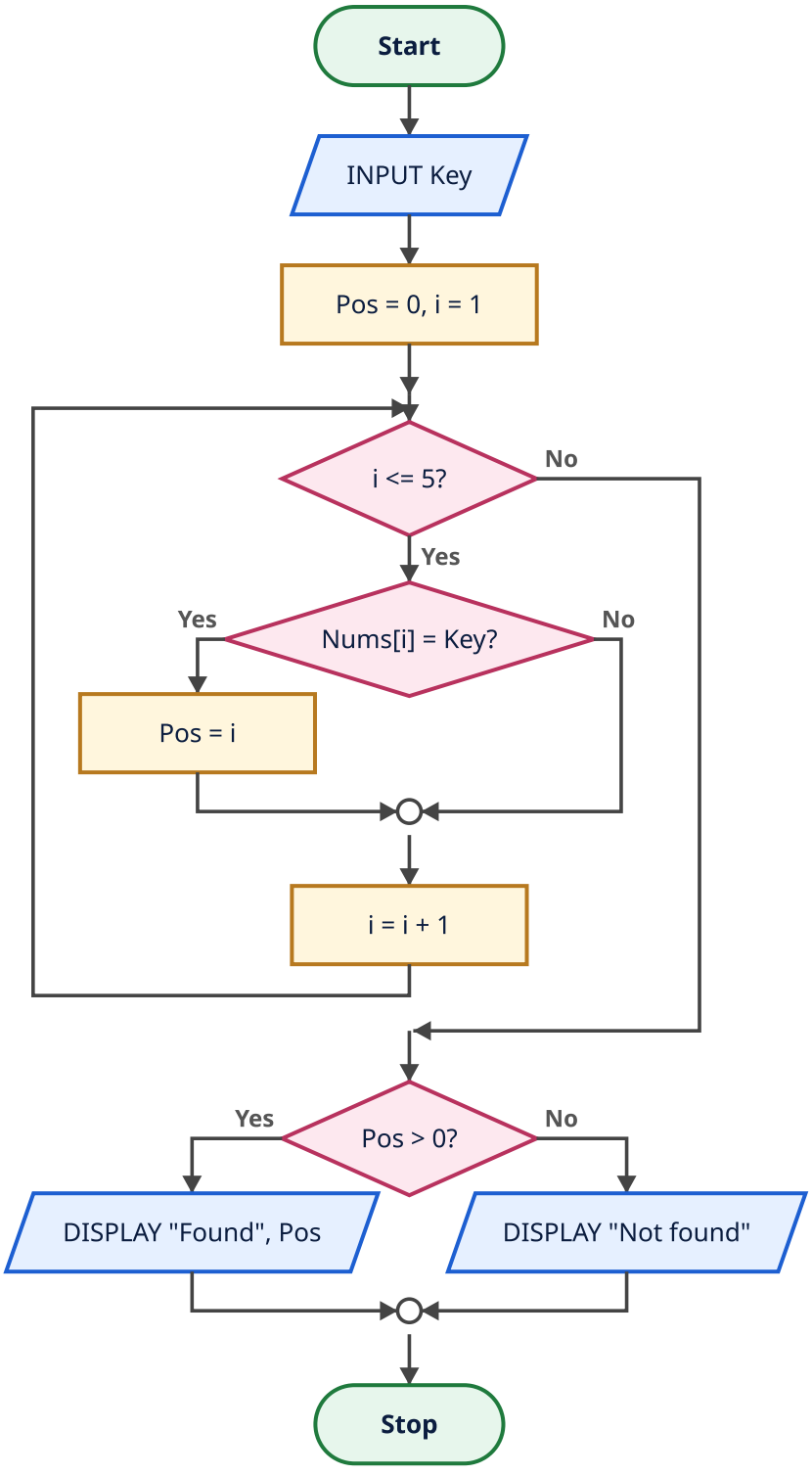
Total = 350 සහ Highest = 90, සැබෑ වැඩසටහන් ප්‍රතිදානයට සමානයි. සාමාන්‍යය = 350 ÷ 5 = 70.00.

5. උදාහරණය: අරාවක සෙවීම (රේඛීය සෙවීම)

Nums අරාවේ 12, 45, 7, 30, 18 ඇත. සංඛ්‍යාවක් සොයා ගැනීමට, සෑම අවයවයක්ම **පළමුවැන්නේ සිට අවසානය දක්වා** පරීක්ෂා කරන්න. මෙය **රේඛීය සෙවීම** (linear search) ලෙස හැඳින්වේ.

ව්‍යාජ කේතය (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
```



ගැලීම් සටහන - අරාව: සංඛ්‍යාවක් සෙවීම

අරාව: සංඛ්‍යාවක් සෙවීම

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")
```

නියැදි ධාවනය 1

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

නියැදි ධාවනය 2

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

i	Nums[i]	Nums[i] = 30 ?	Pos
1	12	නැත	0
2	45	නැත	0
3	7	නැත	0
4	30	ඔව්	4
5	18	නැත	4

6. ස්ථානයක් සොයා ගැනීම

සතියක කොළඹ උෂ්ණත්ව (°C) අරාවක ගබඩා කර ඇත. වැඩසටහන ඉහළම අගය පමණක් නොව එහි **දර්ශකයද** තබා ගන්නා නිසා **කුමන දිනය** උණුසුම්ම වූයේදැයි කිව හැක.

ඉහළම උෂ්ණත්වයේ ස්ථානය

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])
```

නියැදි ධාවනය

```
Day 4: 34
```

ප්‍රධාන පද

අරාව – එක් නමක් යටතේ එකම වර්ගයේ අගයන් එකතුවක්.

ඒක-මාන අරාව – අවයවවල තනි පේළියක් (ලැයිස්තුවක්) වන අරාව.

අවයවය – අරාවක ගබඩා කළ එක් අගයක්.

දර්ශකය – අවයවයක ස්ථාන අංකය, උදා Marks[3] හි 3.

එක ළඟ පිහිටි මතකය – මතකයේ එක ළඟ ගබඩා කළ අවයව.

සෘජු ප්‍රවේශය – දර්ශකය භාවිතයෙන් ඕනෑම අවයවයකට වහාම ළඟා වීම.

ප්‍රකාශ කිරීම – නම, ප්‍රමාණය සහ වර්ගය සමඟ අරාව සෑදීම.

පැවරීම – අවයවයක අගයක් ගබඩා කිරීම, Marks[3] := 45.

ප්‍රවේශය – අවයවයක අගය කියවීම.

List (Python) – Python හි අරා වැනි වර්ගය; දර්ශකය 0 න් ආරම්භ වේ.

රේඛීය සෙවීම – ආරම්භයේ සිට එකින් එක අවයව පරීක්ෂා කිරීම.

අරා ප්‍රමාණය – අරාවක ඇති අවයව ගණන.

ආදර්ශ ප්‍රශ්න සහ පිළිතුරු

1. අරාවක් යනු කුමක්ද?

> **පිළිතුර:** එක් නමක් යටතේ ගබඩා කළ එකම දත්ත වර්ගයේ අගයන් කණ්ඩායමකි.

2. දර්ශකයක් යනු කුමක්ද?

> **පිළිතුර:** අවයවයකට ළඟා වීමට භාවිත කරන ස්ථාන අංකයයි.

3. අරා සඳහා “random access” යන්නෙහි තේරුම කුමක්ද?

> **පිළිතුර:** දර්ශකය මගින් ඕනෑම අවයවයකට කෙළින්ම ළඟා විය හැක.

4. Pascal හි අවයව 5ක පූර්ණ සංඛ්‍යා අරාවක් ප්‍රකාශ කරන්න.

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

5. Python හි පළමු අවයවයේ දර්ශකය කුමක්ද?

> **පිළිතුර:** 0.

6. ලකුණු 65, 80, 45, 90, 70 සඳහා ඉහළම කුමක්ද?

> **පිළිතුර:** 90.

බහුවරණ ප්‍රශ්න

1. අරාවක් බොහෝ අගයන් ගබඩා කරන්නේ...

A. එකම දත්ත වර්ගයේ
B. පාඨ පමණි
C. Boolean පමණි
D. සෑමවිටම වෙනස් දත්ත වර්ගවල

2. අරාවක් තුළ ඇති එක් අගයක් හඳුන්වන්නේ...

A. කාරකය
B. අවයවය
C. දර්ශකය
D. හඳුන්වනය

3. අවයවයක ස්ථාන අංකය එහි...

A. දර්ශකයයි
B. අගයයි
C. වර්ගයයි
D. ප්‍රමාණයයි

4. අරා අවයව ගබඩා වන්නේ...

A. වෙනම ගොනුවල
B. තැටියේ අහඹු ස්ථානවල
C. එක ළඟ පිහිටි මතකයේ
D. මුද්‍රකයේ

5. Marks[1..3] නොකියවා Marks[4] ට ළඟා වීම හඳුන්වන්නේ...

A. කැදැලි කිරීම
B. අනුක්‍රමික ප්‍රවේශය
C. සෘජු ප්‍රවේශය
D. සම්පාදනය

6. පූර්ණ සංඛ්‍යා 5ක අරාවක නිවැරදි Pascal ප්‍රකාශනය:

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

7. Python හි marks හි පළමු අවයවය...

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

8. අවයව 5ක Python list එකක අවසාන දර්ශකය...

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

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

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

10. 65, 80, 45, 90 සහ 70 හි එකතුව...

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

11. Nums = 12, 45, 7, 30, 18. 30 සෙවීමෙන් ලැබෙන්නේ...

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

12. එම අරාවේම 50 සෙවීමෙන් ලැබෙන්නේ...

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

13. ආරම්භයේ සිට එකින් එක අවයව පරීක්ෂා කිරීම...

A. අහඹු සෙවීමකි
B. කැදැලි IF එකකි
C. නියතයකි
D. රේඛීය සෙවීමකි

14. Python හි තුන්වන අවයවයට 45 පවරන ප්‍රකාශය කුමක්ද?

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

15. උෂ්ණත්ව 31, 33, 30, 34, 32, 29, 33. උණුසුම්ම දිනය...

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

ව්‍යුහගත ප්‍රශ්න

1. (i) ඒක-මාන අරාවක් යනු කුමක්ද? (ලකුණු 1)
(ii) වෙනම විචල්‍ය වෙනුවට අරාවක් භාවිතයේ වාසි දෙකක් දෙන්න. (ලකුණු 2) *(ලකුණු 3)*

> (i) එක් නමක් යටතේ එකම වර්ගයේ අවයවවල තනි පේළියක්. (ii) බොහෝ අගයන්ට එක් නමක්; ලූපයකින් සියල්ල සැකසිය හැක; ප්‍රමාණය වෙනස් කිරීම පහසුයි (ඕනෑම දෙකක්).

2. (a) දර්ශකය (b) එක ළඟ පිහිටි මතකය (c) සෘජු ප්‍රවේශය යන පද පහදන්න. (ලකුණු 3) *(ලකුණු 3)*

(a) අවයවයක ස්ථාන අංකය. (b) අවයව මතකයේ එක ළඟ ගබඩා වේ. (c) දර්ශකය භාවිතයෙන් ඕනෑම අවයවයකට කෙළින්ම ළඟා විය හැක.

3. ලකුණු 5ක් අරාවකට කියවා එකතුව, සාමාන්‍යය සහ ඉහළම ලකුණ පෙන්වන වැඩසටහනක් (Pascal හෝ Python) ලියන්න. (ලකුණු 6) *(ලකුණු 6)*

අරාව ප්‍රකාශ කිරීම (1); කියවීමට ලූපය (1); total = 0, highest = පළමු අවයවය (1); එකතු කිරීම සහ සැසඳීමේ ලූපය (2); total, average, highest පෙන්වීම (1).

අරාව: ලකුණු 5ක් ගබඩා කර එකතුව, සාමාන්‍යය, ඉහළම

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)
```

නියැදි ධාවනය

```
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
```

4. Nums = 12, 45, 7, 30, 18. 30 සඳහා රේඛීය සෙවීමක අනුරේඛන වගුවක් සම්පූර්ණ කර ප්‍රතිදානය දෙන්න. (ලකුණු 4) *(ලකුණු 4)*

i = 1…5 සමඟ Nums[i] = 30? නැත, නැත, නැත, ඔව්, නැත; Pos = 0, 0, 0, 4, 4 (3); ප්‍රතිදානය Found at position 4 (1).

i	Nums[i]	Nums[i] = 30 ?	Pos
1	12	නැත	0
2	45	නැත	0
3	7	නැත	0
4	30	ඔව්	4
5	18	නැත	4

5. (a) Marks හි තුන්වන අවයවයේ 45 ගබඩා කිරීමට (b) පළමු අවයවය පෙන්වීමට Pascal සහ Python ප්‍රකාශ ලියන්න. (ලකුණු 4) *(ලකුණු 4)*

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

සාරාංශය

- අරාව: බොහෝ අගයන්, එකම වර්ගය, එක් නමක්.
- අවයවය = එක් අගයක්; දර්ශකය = එහි ස්ථානය.
- එක ළඟ මතකය → සෘජු ප්‍රවේශය.
- Pascal: array[1..5] of integer.
- Python list දර්ශකය 0 න් ආරම්භ වේ.
- අරාවක් පිරවීමට සහ සැකසීමට ලූපයක් භාවිත කරන්න.
- එක් ලූපයකින් එකතුව, සාමාන්‍යය, ඉහළම.
- රේඛීය සෙවීම අවයව එකින් එක පරීක්ෂා කරයි.

MCQ පිළිතුරු – 1.9

1. **A** – එකම වර්ගය.
2. **B** – අවයවය.
3. **A** – දර්ශකය.
4. **C** – එක ළඟ.
5. **C** – සෘජු.
6. **D** – Pascal වාක්‍ය රීතිය.
7. **B** – 0 සිට දර්ශකය.
8. **A** – 0 සිට 4.
9. **B** – තුන්වන අවයවය.
10. **C** – එකතුව.

11. **A** – හතරවන අවයවය.
12. **C** – අරාවේ නැත.
13. **D** – රේඛීය.
14. **D** – දර්ශකය 2 = තුන්වන.
15. **B** – 34 හතරවැන්නයි.

© 2026 schoolict.net (Ruwan Mangala Suraweera). සියලු හිමිකම් ඇවිරිණි.