G.C.E. O/L ICT · 11 ශ්‍රේණිය · පාඩම 1

පාඩම 1 O/L විභාග පුහුණුව: ප්‍රශ්න සහ පිළිතුරු

මේ ප්‍රශ්න O/L ICT විභාගයේ ආකාරයටයි හදලා තියෙන්නේ. Paper I එකේ බහුවරණ ප්‍රශ්න ඇත. Paper II එකේ කොටස්, ආදර්ශ පිළිතුරු සහ ලකුණු දීමේ කරුණු සහිත ව්‍යුහගත ප්‍රශ්න ඇත. ඒවා පාඩම 1 සම්පූර්ණයෙන්ම (1.1 සිට 1.11) ආවරණය කරයි: ගැටලු විශ්ලේෂණය, ඇල්ගොරිතම, ගැලීම් සටහන් සහ ව්‍යාජ කේත, විචල්‍ය සහ දත්ත වර්ග, කාරක, වරණය, පුනරාවර්තනය, කැදැලි, අරා, උප වැඩසටහන්, භාෂා සහ පරිවර්තක, පරීක්ෂාව සහ දෝෂ නිරාකරණය. සෑම වැඩසටහන් පිළිතුරක්ම Pascal (Free Pascal) සහ Python වලින් සැබවින්ම ධාවනය කර ඇත. මුලින් උත්සාහ කර පසුව පිළිතුරු පරීක්ෂා කරන්න.

Paper I – බහුවරණ ප්‍රශ්න (25)

1. බිල් වැඩසටහනක IPO අනුව මිල සහ ප්‍රමාණය

A. සැකසීම්
B. ආදාන
C. නියත
D. ප්‍රතිදාන

2. ගැටලුවකට පියවරෙන් පියවර විසඳුම

A. අරාවයි
B. කාරකයයි
C. ඇල්ගොරිතමයයි
D. හඳුන්වනයයි

3. කොන්දේසියක් මගින් මාර්ග දෙකක් අතර තේරීම

A. පුනරාවර්තනයයි
B. කැදැලියයි
C. අනුක්‍රමයයි
D. වරණයයි

4. තීරණයක් සඳහා ගැලීම් සටහන් සංකේතය

A. ඕවලයයි
B. දියමන්තියයි
C. සමාන්තරාස්‍රයයි
D. සෘජුකෝණාස්‍රයයි

5. ආදාන සහ ප්‍රතිදාන පෙන්වන්නේ

A. සමාන්තරාස්‍රයකිනි
B. සෘජුකෝණාස්‍රයකිනි
C. කවයකිනි
D. දියමන්තියකිනි

6. වලංගු හඳුන්වනයක් කුමක්ද?

A. 2ndMark
B. Total Marks
C. Total_Marks
D. begin

7. 71.67 වැනි සාමාන්‍යයකට හොඳම දත්ත වර්ගය

A. real
B. integer
C. char
D. Boolean

8. 17 mod 5 සමාන වන්නේ

A. 3.4
B. 3
C. 12
D. 2

9. 2 + 3 * 4 හි අගය

A. 24
B. 14
C. 20
D. 9

10. Mark = 120 සඳහා (Mark >= 35) AND (Mark <= 100)

A. True
B. 120
C. False
D. දෝෂයකි

11. O/L සීමා අනුව ලකුණු 66 ට ශ්‍රේණිය

A. A
B. S
C. C
D. B

12. Pascal හි else ට පෙර නොලිය යුත්තේ

A. semicolon එකක්
B. හිස්තැනක්
C. begin
D. then

13. 1–7 අංකයකින් දින නම තේරීමට වඩාත් සුදුසු

A. FOR
B. අරා දෝෂයක්
C. CASE
D. REPEAT

14. සැමවිටම අවම වශයෙන් එක් වරක් ධාවනය වන ලූපය

A. IF
B. REPEAT … UNTIL
C. FOR
D. WHILE

15. for i in range(1, 6) ඇතුළත ධාවනය වන්නේ

A. 7 වරක්
B. 6 වරක්
C. 5 වරක්
D. 1 වරක්

16. සංඛ්‍යා කියවීම නවත්වන්නට 0 ඇතුළත් කිරීම 0 භාවිත කරන්නේ

A. වෙන් කළ වචනයක් ලෙස
B. දර්ශකයක් ලෙස
C. නියතයක් ලෙස
D. sentinel අගයක් ලෙස

17. ක්‍රමාරෝපිත වැඩසටහන ගණනය කරන 5! අගය

A. 25
B. 60
C. 15
D. 120

18. පිටත ලූපය 4 වරක්, ඇතුළු ලූපය සෑම විටම 3 වරක්: ඇතුළත ධාවනය වන්නේ

A. 4 වරක්
B. 12 වරක්
C. 1 වරක්
D. 7 වරක්

19. අරා අවයව ගබඩා වන්නේ

A. එක ළඟ පිහිටි මතකයේ
B. වෙනම ගොනුවල
C. CPU එකේ පමණි
D. අහඹු ස්ථානවල

20. Python හි marks හි තුන්වන අවයවය

A. marks[3]
B. marks[2]
C. marks(3)
D. marks[1]

21. අගයක් ආපසු දෙන උප වැඩසටහනක්

A. ශ්‍රිතයකි
B. ලූපයකි
C. ක්‍රියා පටිපාටියකි
D. අරාවකි

22. කාර්යයක් එක් තැනක පමණක් වෙනස් කිරීම පෙන්වන උප වැඩසටහන් වාසිය

A. නඩත්තුවේ පහසුව
B. සෘජු ප්‍රවේශය
C. පුනරාවර්තනය
D. ප්‍රමුඛතාව

23. සම්පූර්ණ වැඩසටහනම එකවර පරිවර්තනය කරන පරිවර්තකය

A. සංස්කාරකයකි
B. අර්ථවිනයකයකි
C. සම්පාදකයකි
D. ඇසෙම්බ්ලරයකි

24. ධාවනය වුවත් වැරදි පිළිතුරක් දෙන වැඩසටහනක ඇත්තේ

A. දෝෂයක් නැත
B. ධාවන කාල දෝෂයකි
C. වාක්‍ය රීති දෝෂයකි
D. තාර්කික දෝෂයකි

25. AI කේත සහායකයක් වගකීමෙන් භාවිතය

A. එය යෝජනා කරන කේතය තේරුම් ගෙන පරීක්ෂා කිරීම
B. ඔබේ මුරපද එය සමඟ බෙදා ගැනීම
C. එහි කේතය ඔබේ විභාග පිළිතුර ලෙස ඉදිරිපත් කිරීම
D. එහි කේතය කිසිදා නොකියවීම

Paper II – ව්‍යුහගත ප්‍රශ්න

1. ගැටලු විශ්ලේෂණය සහ ඇල්ගොරිතම. සාප්පුවක් අයිතම විකුණයි. මුදල් අයකැමි මිල සහ ප්‍රමාණය ඇතුළත් කරයි, වැඩසටහන මුළු මුදල පෙන්වයි. *(ලකුණු 9)*

(i) ආදානය, සැකසීම සහ ප්‍රතිදානය හඳුනාගන්න. *(ලකුණු 3)*
(ii) මූලික පාලන ව්‍යුහ තුන නම් කරන්න. *(ලකුණු 3)*
(iii) බිල් වැඩසටහන සඳහා ගැලීම් සටහනක් අඳින්න. *(ලකුණු 3)*

2. ගැලීම් සටහන්, ව්‍යාජ කේත සහ අනුරේඛන වගු. *(ලකුණු 9)*

(i) 1 සිට N දක්වා එකතුව සොයන ව්‍යාජ කේතය ලියන්න. *(ලකුණු 3)*
(ii) N = 5 සඳහා අනුරේඛන වගුවක් සම්පූර්ණ කරන්න. *(ලකුණු 3)*
(iii) ආරම්භය, සැකසීම සහ තීරණය සඳහා සංකේත නම් කරන්න. *(ලකුණු 3)*

3. විචල්‍ය, දත්ත වර්ග සහ කාරක. *(ලකුණු 9)*

(i) (a) සිසුන් ගණන (b) මිල (c) නම (d) සමත්ද නැද්ද සඳහා සුදුසු දත්ත වර්ගයක් දෙන්න. *(ලකුණු 4)*
(ii) 17 div 5, 17 mod 5 සහ (2 + 3) * 4 අගයන් සොයන්න. *(ලකුණු 3)*
(iii) හඳුන්වන නම් කිරීමේ රීති දෙකක් දෙන්න. *(ලකුණු 2)*

4. වරණය. ශ්‍රේණි: A ≥ 75, B ≥ 65, C ≥ 50, S ≥ 35, නැතහොත් W. *(ලකුණු 8)*

(i) ලකුණක් ආදානය කර ශ්‍රේණිය පෙන්වන වැඩසටහනක් (Pascal හෝ Python) ලියන්න. *(ලකුණු 5)*
(ii) 82, 66 සහ 47 සඳහා පෙන්වන ශ්‍රේණි මොනවාද? *(ලකුණු 3)*

5. පුනරාවර්තනය සහ කැදැලි. *(ලකුණු 8)*

(i) ලූපයක් භාවිතයෙන් 5! ගණනය කර පෙන්වන වැඩසටහනක් ලියන්න. *(ලකුණු 3)*
(ii) WHILE සහ REPEAT … UNTIL අතර එක් වෙනසක් දෙන්න. *(ලකුණු 2)*
(iii) පේළි 4 තරු ත්‍රිකෝණය සඳහා අනුරේඛන වගුවක් සම්පූර්ණ කරන්න. *(ලකුණු 3)*

6. අරා සහ උප වැඩසටහන්. Nums = 12, 45, 7, 30, 18. *(ලකුණු 9)*

(i) Nums[2] (Pascal) සහ nums[2] (Python) මොනවාද? *(ලකුණු 2)*
(ii) 30 සඳහා රේඛීය සෙවීමක් අනුරේඛනය කර ප්‍රතිදානය දෙන්න. *(ලකුණු 3)*
(iii) ශ්‍රිතයක් ක්‍රියා පටිපාටියකින් වෙන්කර උප වැඩසටහන්වල වාසි දෙකක් දෙන්න. *(ලකුණු 4)*

7. භාෂා, පරිවර්තක සහ දෝෂ නිරාකරණය. *(ලකුණු 7)*

(i) සම්පාදකයක් සහ අර්ථවිනයකයක් අතර වෙනස් දෙකක් දෙන්න. *(ලකුණු 2)*
(ii) 75, 68, 72 හි සාමාන්‍යය 107.50 ලෙස පෙන්වයි. දෝෂය නම් කර නිවැරදි කරන්න. *(ලකුණු 3)*
(iii) Python IDE එකක් නම් කර AI සහායක වගකීමෙන් භාවිතයට එක් රීතියක් දෙන්න. *(ලකුණු 2)*

විභාග උපදෙස්

- IPO: ආදානය → සැකසීම → ප්‍රතිදානය.
- ඇල්ගොරිතමය: පැහැදිලි, පිළිවෙළ, සීමිත පියවර.
- ඕවලය, සමාන්තරාස්‍රය, සෘජුකෝණාස්‍රය, දියමන්තිය.
- අනුරේඛන වගු: එක් වෙනසකට එක් පේළියක්.
- හඳුන්වන: මුලින් අකුර, හිස්තැන් නැත, වෙන් කළ වචන නැත.
- div = පූර්ණ කොටස; mod = ශේෂය; මුලින් වරහන්.
- Pascal හි else ට පෙර semicolon නැත.
- FOR ස්ථාවර; WHILE මුලින් පරීක්ෂාව; REPEAT අවසානයේ.
- පිටත සෑම ධාවනයකටම ඇතුළු ලූපය සම්පූර්ණයෙන්.
- Pascal array[1..5]; Python list දර්ශකය 0.
- ශ්‍රිතය අගයක් ආපසු දෙයි; ක්‍රියා පටිපාටිය නොදෙයි.
- සම්පාදකය එදිරිව අර්ථවිනයකය; syntax, runtime, logic දෝෂ.

පිළිතුරු සහ පැහැදිලි කිරීම්

Paper I

1. **B** – ඇතුළත් කරයි.
2. **C** – ඇල්ගොරිතමය.
3. **D** – වරණය.
4. **B** – තීරණය.
5. **A** – I/O.
6. **C** – මුලින් අකුර, හිස්තැන් නැත.
7. **A** – දශම.
8. **D** – ශේෂය.
9. **B** – * මුලින්.
10. **C** – දෙවන කොටස False.
11. **D** – 65–74.
12. **A** – else ට පෙර ; නැත.
13. **C** – අගය අනුව.
14. **B** – අවසානයේ පරීක්ෂාව.
15. **C** – 1 සිට 5.
16. **D** – Sentinel.
17. **D** – 1×2×3×4×5.
18. **B** – 4 × 3.
19. **A** – එක ළඟ.
20. **B** – 0 සිට.
21. **A** – ශ්‍රිතය.
22. **A** – නඩත්තුව.
23. **C** – සම්පාදකය.
24. **D** – තාර්කික.
25. **A** – වගකීම්.

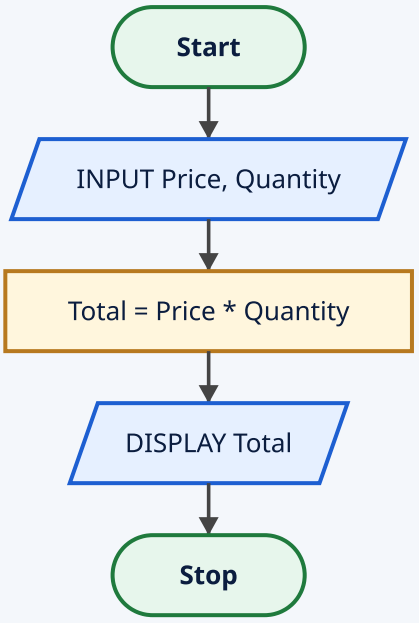
Paper II – ආදර්ශ පිළිතුරු සහ ලකුණු දීමේ කරුණු

1. *(ලකුණු 9)*

(i) (ලකුණු 3)
ආදානය – මිල, ප්‍රමාණය (1); සැකසීම – total = price × quantity (1); ප්‍රතිදානය – total (1).

(ii) (ලකුණු 3)
අනුක්‍රමය, වරණය, පුනරාවර්තනය (එකට 1).

(iii) (ලකුණු 3)
Start; price, qty ආදානය; total = price × qty; total පෙන්වීම; Stop (3).



ගැලීම් සටහන – කඩ බිල: මිල × ප්‍රමාණය

🎯 පරීක්ෂක උපදෙස: ආදාන ඇතුළට, ප්‍රතිදාන පිටතට – සැකසීම මැද.

2. *(ලකුණු 9)*

(i) (ලකුණු 3)
INPUT N; Sum = 0; FOR i = 1 TO N, Sum = Sum + i; OUTPUT Sum (3).
ව්‍යාජ කේතය (Pseudocode)

```
BEGIN
  INPUT N
  Total = 0
  i = 1
  WHILE i <= N DO
    Total = Total + i
    i = i + 1
  ENDWHILE
  DISPLAY Total
END
```

(ii) (ලකුණු 3)
Sum: 1, 3, 6, 10, 15; ප්‍රතිදානය 15 (3).

පියවර	i	Total	i <= 5 ?
start	1	0	–
loop	1	1	ඔව්
loop	2	3	ඔව්
loop	3	6	ඔව්
loop	4	10	ඔව්
loop	5	15	ඔව්
check	6	15	නැත

(iii) (ලකුණු 3)
ඕවලය, සෘජුකෝණාස්‍රය, දියමන්තිය (එකට 1).

🎯 පරීක්ෂක උපදෙස්: අනුරේඛන වගුවේ විචල්‍යයක් වෙනස් වන සෑම විටම නව පේළියක් ලියන්න.

3. *(ලකුණු 9)*

(i) (ලකුණු 4)
(a) integer (b) real (c) string (d) Boolean (එකට 1).

(ii) (ලකුණු 3)
3, 2, 20 (එකට 1).

අංක ගණිත කාරක

Pascal

```
program Operators;
var
  a, b: integer;
begin
  a := 17;
  b := 5;
  writeln('Sum = ', a + b);
  writeln('Difference = ', a - b);
  writeln('Product = ', a * b);
  writeln('Division = ', a / b:0:2);
  writeln('Quotient = ', a div b);
  writeln('Remainder = ', a mod b);
end.
```

Python

```
a = 17
b = 5
print("Sum =", a + b)
print("Difference =", a - b)
print("Product =", a * b)
print(f"Division = {a / b:.2f}")
print("Quotient =", a // b)
print("Remainder =", a % b)
```

නියැදි ධාවනය

```
Sum = 22
Difference = 12
Product = 85
Division = 3.40
Quotient = 3
Remainder = 2
```

(iii) (ලකුණු 2)
අකුරකින් ආරම්භ; හිස්තැන් නැත; වෙන් කළ වචනයක් නොවේ (ඕනෑම දෙකක්).

🎯 පරීක්ෂක උපදෙස්: div පූර්ණ කොටස, mod ශේෂය දෙයි.

4. *(ලකුණු 8)*

(i) (ලකුණු 5)

ආදානය (1); IF ≥ 75 A (1); ELSE IF ≥ 65 B, ≥ 50 C (2); ≥ 35 S නැතහොත් W (1).

ලකුණු ශ්‍රේණියට (A, B, C, S, W)

Pascal

```
program MarksToGrade;
var
  marks: integer;
  grade: char;
begin
  write('Enter marks: ');
  readln(marks);
  if marks >= 75 then
    grade := 'A'
  else if marks >= 65 then
    grade := 'B'
  else if marks >= 50 then
    grade := 'C'
  else if marks >= 35 then
    grade := 'S'
  else
    grade := 'W';
  writeln('Grade: ', grade);
end.
```

Python

```
marks = int(input("Enter marks: "))
if marks >= 75:
    grade = "A"
elif marks >= 65:
    grade = "B"
elif marks >= 50:
    grade = "C"
elif marks >= 35:
    grade = "S"
else:
    grade = "W"
print("Grade:", grade)
```

නියැදි ධාවනය 1

```
Enter marks: 82
Grade: A
```

නියැදි ධාවනය 2

```
Enter marks: 58
Grade: C
```

නියැදි ධාවනය 3

```
Enter marks: 30
Grade: W
```

(ii) (ලකුණු 3)
A, B, S (එකට 1).

🎯 පරීක්ෂක උපදෙස: මායිම් ලකුණු 75, 65, 50 සහ 35 පරීක්ෂා කරන්න.

5. *(ලකුණු 8)*

(i) (ලකුණු 3)
Fact = 1 (1); FOR i = 1 TO 5, Fact = Fact × i (1); 120 පෙන්වීම (1).

FOR loop මගින් ක්‍රමාරෝපිතය (factorial)

Pascal

```
program Factorial;
var
  n, i, f: integer;
begin
  write('Enter N: ');
  readln(n);
  f := 1;
  for i := 1 to n do
    f := f * i;
  writeln('Factorial = ', f);
end.
```

Python

```
n = int(input("Enter N: "))
f = 1
for i in range(1, n + 1):
    f = f * i
print("Factorial =", f)
```

නියැදි ධාවනය

```
Enter N: 5
Factorial = 120
```

i	F = F * i
1	1
2	2
3	6
4	24
5	120

(ii) (ලකුණු 2)
WHILE ආරම්භයේ පරීක්ෂා කර ශූන්‍ය වරක් විය හැක; REPEAT අවසානයේ පරීක්ෂා කර අවම වශයෙන් එක් වරක් ධාවනය වේ (2).

(iii) (ලකුණු 3)
පේළි *, , *, **** මුද්‍රණය කරයි (3).

පුනරාවර්තනය තුළ පුනරාවර්තනය: තරු ත්‍රිකෝණය

Pascal

```
program Stars;
var
  row, col: integer;
begin
  for row := 1 to 4 do
  begin
    for col := 1 to row do
      write('*');
    writeln;
  end;
end.
```

Python

```
for row in range(1, 5):
    for col in range(1, row + 1):
        print("*", end="")
    print()
```

නියැදි ධාවනය

```
*
**
***
****
```

Row	ඇතුළු ලූපය ධාවනය (වාර)	මුද්‍රණය වූ දේ
1	1	*
2	2	**
3	3	***
4	4	****

🎯 පරීක්ෂක උපදෙස්: කැදැලි ලූපයක පිටත ලූපය ඉදිරියට යාමට පෙර ඇතුළු ලූපය සම්පූර්ණයෙන් අවසන් වේ.

6. *(ලකුණු 9)*

(i) (ලකුණු 2)
45 (1); 7 (1).

(ii) (ලකුණු 3)
12, 45, 7, 30 පරීක්ෂා කරයි → Found at position 4 (3).

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

(iii) (ලකුණු 4)
ශ්‍රිතය අගයක් ආපසු දෙයි, ක්‍රියා පටිපාටිය නොදෙයි (2); නැවත භාවිතය, කියවීමේ පහසුව, නඩත්තුව (ඕනෑම දෙකක්) (2).

🎯 පරීක්ෂක උපදෙස්: මෙහි Pascal අරා 1 න්, Python list 0 න් ආරම්භ වේ.

7. *(ලකුණු 7)*

(i) (ලකුණු 2)
සම්පූර්ණ වැඩසටහන එදිරිව පේළියෙන් පේළිය; ක්‍රියාත්මක ගොනුව එදිරිව නැත; වේගවත් එදිරිව මන්දගාමී (ඕනෑම දෙකක්).

(ii) (ලකුණු 3)
තාර්කික දෝෂය (1); එකතුව 2 න් බෙදා ඇත (1); 3 න් බෙදන්න → 71.67 (1).

තාර්කික දෝෂය: වැරදි සංඛ්‍යාවෙන් බෙදීම

Pascal

```
program AverageBug;
var
  m1, m2, m3, total: integer;
  avg: real;
begin
  write('Enter mark 1: ');
  readln(m1);
  write('Enter mark 2: ');
  readln(m2);
  write('Enter mark 3: ');
  readln(m3);
  total := m1 + m2 + m3;
  avg := total / 2;            { BUG: should be / 3 }
  writeln('Average = ', avg:0:2);
end.
```

Python

```
m1 = int(input("Enter mark 1: "))
m2 = int(input("Enter mark 2: "))
m3 = int(input("Enter mark 3: "))
total = m1 + m2 + m3
avg = total / 2               # BUG: should be / 3
print(f"Average = {avg:.2f}")
```

නියැදි ධාවනය

```
Enter mark 1: 75
Enter mark 2: 68
Enter mark 3: 72
Average = 107.50
```

(iii) (ලකුණු 2)
IDLE / Thonny / VS Code (1); කේතය තේරුම් ගෙන පරීක්ෂා කරන්න, ඔබේ ලෙස ඉදිරිපත් නොකරන්න (1).

🎯 පරීක්ෂක උපදෙස්: syntax = ව්‍යාකරණ, runtime = බිඳ වැටීම, logic = වැරදි පිළිතුර.

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