

Information and Communication Technology

තොරතුරු හා සන්නිවේදන තාක්ෂණය

2026 QUIZ NO 515 MARKING

2026/05/09
ictfromabc
Ravindu Bandaranayake

1. The time taken by the actuator arm of a hard disk to move the header to the sector containing the relevant data is called by which name?

දෘඪ තැටියක ඇක්ටුවේටර් බාහුව මගින් ශීර්ෂකය අදාළ දත්තය ඇතුළත් sector එක වෙතට යොමුවීමට ගතවන කාලය කුමන නමකින් හැඳින්වේද?

- 1) Seek time 2) Rotational delay 3) Access time 4) cluster time 5) Sector delay

Answer : 1)

Let's consider each of these applications related to hard disk.

දෘඪ තැටියට අදාළ මෙම එක් එක් යෙදුම් පිළිබඳව සලකා බලමු.

Seek Time:

The hard disk's data is stored in concentric circles known as tracks. When data is requested, the hard disk must move the head to the track containing that data. Seek time is the time it takes for the actuator arm of the hard disk to move the read/write head to the correct track where the data is located. Seek time is one of the most critical factors in determining the speed of a hard disk.

දෘඪ තැටියේ දත්ත, track ලෙස හඳුන්වන පටි තුළ ගබඩා කර ඇත. දත්ත ඉල්ලා සිටින විට, දෘඪ තැටිය එම දත්ත අඩංගු track එකට ශීර්ෂකය ගෙන යා යුතුය. Seek time යනු දෘඪ තැටියේ ඇක්ටුවේටර් බාහුවට කියවීමේ/ලිවීමේ හිස දත්ත ඇති නිවැරදි track එකට ගෙනයාමට ගතවන කාලයයි. දෘඪ තැටියක වේගය තීරණය කිරීමේදී seek time වඩාත් තීරණාත්මක සාධකයක් වේ.

Rotational Delay (Latency):

Rotational delay, also known as rotational latency, is the time it takes for the desired sector of the disk to rotate under the read/write head after the head is positioned over the correct track. Once the head is over the right track (after the seek time), the disk must rotate until the correct sector is under the head. The average rotational delay depends on the speed at which the disk spins (measured in RPM).

Rotational delay" rotational latency ලෙසද හැඳින්වේ, නිවැරදි track එකෙහි ශීර්ෂකය ස්ථානගත කිරීමෙන් පසු තැටියේ අපේක්ෂිත sector එක කියවීමේ/ලිවීමේ ශීර්ෂකය යටතට භ්‍රමණය වීමට ගතවන කාලයයි. ශීර්ෂකය නිවැරදි track එකට ගිය පසු (seek time එකෙන් පසු), නිවැරදි sector එක ශීර්ෂකයට යටින් ඇති තෙක් තැටිය භ්‍රමණය විය යුතුය. සාමාන්‍යයෙන් Rotational delay එක තැටිය භ්‍රමණය වන වේගය මත රඳා පවති (RPM) වලින් මනිනු ලැබේ.

Access Time:

Access time is the total time it takes to retrieve data from the disk. It is the sum of the seek time, the rotational delay and read/write time.

ප්‍රවේශ කාලය යනු තැටියෙන් දත්ත ලබාගැනීමට ගතවන මුළු කාලයයි. එය seek time, rotational delay සහ කියවීමේ/ලිවීමේ කාලයෙහි එකතුවයි.

Terms such as cluster time, sector delay mentioned in the answers here do not exist.

මෙහි පිළිතුරුවල සඳහන් cluster time, sector delay ලෙස යෙදුම් නොපවතී.

So, the correct answer number here is 1).

එමනිසා, මෙහි නිවැරදි පිළිතුරු අංකය 1) වේ.

2. What is the 8 bit 2's complement representation of -57?
-57 හි 8 bit 2 හි අනුපූරක නිරූපණය කුමක්ද?

- 1) 11000110₂ 2) 11000111₂ 3) 00111001₂ 4) 01000111₂ 5) 00111000₂

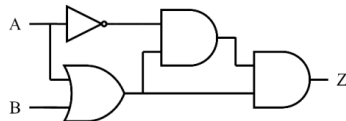
Answer : 2)

First Let's find the 8 bit binary representation of 57 and -57 මුලින්ම අපි 57 සහ -57 හි බිටු 8 ද්විමය නිරූපණය සොයා ගනිමු							
2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰
0	0	1	1	1	0	0	1
57 = 00111001 ₂							
-57 = 11000110 ₂							

Now Let's Find the 2's complement of -57 දැන් අපි -57 හි 2 හි අනුපූරකය සොයා ගනිමු							
1	1	0	0	0	1	1	0
						+	1
1	1	0	0	0	1	1	1
2's complement of -57 = 11000111 ₂							

Therefore, the correct answer is 2).
එබැවින් නිවැරදි පිළිතුර 2) වේ.

3. What is the Boolean expression that can be obtain from the below logic circuit?
පහත තාර්කික පරිපථයෙන් ලබාගත හැකි බුලිය ප්‍රකාශනය කුමක්ද?

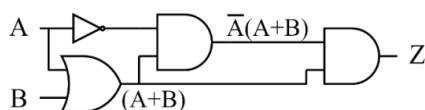


- 1) $\bar{A}B$ 2) 0 3) A 4) 1 5) $(\bar{A} + B)(A \cdot B)$

Answer: 1)

Explanation / පැහැදිලි කිරීම:

Let's obtained an expression from the circuit. / පරිපථයෙන් ප්‍රකාශනයක් ලබා ගනිමු.



Let's simplify the expression. / ප්‍රකාශනය සරල කර ගනිමු.

$Z = (A + B)(A + B)\bar{A}$	
$= (AA + AB + AB + BB)\bar{A}$	Distributive law / විකටන න්‍යාය
$= (A + AB + AB + B)\bar{A}$	Idempotent law / තදේවතාවී න්‍යාය
$= (A + AB + B)\bar{A}$	Idempotent law / තදේවතාවී න්‍යාය
$= (A + AB + B)\bar{A}$	Idempotent law / තදේවතාවී න්‍යාය
$= (A(1 + B) + B)\bar{A}$	Distributive law / විකටන න්‍යාය
$= (A(1) + B)\bar{A}$	Identity law / සර්වකාමය න්‍යාය
$= (A + B)\bar{A}$	Identity law / සර්වකාමය න්‍යාය
$= A\bar{A} + \bar{A}B$	Distributive law / විකටන න්‍යාය
$= 0 + \bar{A}B$	Inverse law / ප්‍රතිලෝම න්‍යාය
$= \bar{A}B$	Identity law / සර්වකාමය න්‍යාය

Therefore, the correct answer is 1). / එබැවින්, නිවැරදි පිළිතුර 1) වේ.

4. Consider a computer system where the CPU executes the following sequence of instructions.
CPU විසින් පහත උපදෙස් මාලාව ක්‍රියාත්මක කරන පරිගණක පද්ධතියක් සලකා බලන්න.

- Load data from memory address 1000 into register A.
1000 වන මතක ලිපිනය වෙතින් දත්ත A රෙජිස්තරය වෙත පුරණය කරයි.
- Add the value in register A to the value in register B and store the result in register C
A රෙජිස්තරයෙහි ඇති අගය B රෙජිස්තරයේ ඇති අගයට එකතු කර එහි ප්‍රතිඵලය C රෙජිස්තරයෙහි ගබඩා කරයි.
- Store the value in register C back to memory address 2000.
C රෙජිස්තරයෙහි අගය නැවත මතක ලිපිනය 2000 වෙත ගබඩා කරයි.

Which statement best describes the purpose of these instructions?

මෙම උපදෙස්වල අරමුණ වඩාත් හොඳින් විස්තර කරන ප්‍රකාශය කුමක්ද?

- The instructions are performing arithmetic operations and memory management.
උපදෙස් අංක ගණිතමය මෙහෙයුම් සහ මතක කළමනාකරණය සිදු කරයි.
- The instructions are performing input/output operations.
උපදෙස් ආදාන/ප්‍රතිදාන මෙහෙයුම් සිදු කරයි.
- The instructions are involved in network communication. / උපදෙස් ජාල සන්නිවේදනයට සම්බන්ධ වේ.
- The instructions are handling system security and access control.
උපදෙස් පද්ධතියේ ආරක්ෂාව සහ ප්‍රවේශ පාලනය හැසිරවීමයි.
- The instructions are executing program code for user interface updates.
උපදෙස් පරිශීලක අතුරුමුහුණත් යාවත්කාලීන සඳහා වැඩසටහන් කේතය ක්‍රියාත්මක කරයි.

Answer : 1)

The sequence of instructions in this example involves loading data from memory, performing arithmetic operation (addition), and storing the result back into memory.

මෙම උදාහරණයේ ඇති උපදෙස් අනුපිළිවෙලට අනුව මතකයෙන් දත්ත පුරණය කිරීම, ගණිතමය මෙහෙයවන (එකතු කිරීම) සිදු කිරීම සහ ප්‍රතිඵලය නැවත මතකයේ ගබඩා කිරීම ඇතුළත් වේ.

Let's go through the instructions step by step / පියවරෙන් පියවර උපදෙස් හරහා යමු.

A. Load data from memory address 1000 into register A.

1000 වන මතක ලිපිනය වෙතින් දත්ත A රෙජිස්තරය වෙත පුරණය කරයි.

This is fetching data from a specific memory address.

මෙය නිශ්චිත මතක ලිපිනයකින් දත්ත ලබා ගැනීමයි.

B. Add the value in register A to the value in register B and store the result in register C

A රෙජිස්තරයෙහි ඇති අගය B රෙජිස්තරයේ ඇති අගයට එකතු කර එහි ප්‍රතිඵලය C රෙජිස්තරයෙහි ගබඩා කරයි.

This is an arithmetic operation (addition).

මෙය අංක ගණිතමය මෙහෙයවනයකි (එකතු කිරීම).

C. Store the value in register C back to memory address 2000.

C රෙජිස්තරයෙහි අගය නැවත මතක ලිපිනය 2000 වෙත ගබඩා කරයි.

This stores the result of the arithmetic operation back to a different memory location.

මෙය අංක ගණිතමය මෙහෙයවනයේ ප්‍රතිඵලය වෙනත් මතක ස්ථානයකට නැවත ගබඩා කරයි.

5. Which SQL keyword is used to combine rows from two or more tables, based on a related column between them, and includes unmatched rows from the left table?

වගු දෙකකින් හෝ වැඩි ගණනකින් පේළි ඒකාබද්ධ කිරීමට භාවිත කරන SQL මූල පදය, ඒවා අතර අදාළ තීරුවක් මත පදනම්ව, සහ වම් වගුවෙන් නොගැලපෙන පේළි ඇතුළත් වේ ද?

- 1) INNER JOIN 2) LEFT JOIN 3) RIGHT JOIN 4) FULL OUTER JOIN 5) CROSS JOIN

Answer : 2)

Explanation of each option / එක් එක් විකල්පය පැහැදිලි කිරීම

1) INNER JOIN

Combines rows from two or more tables where there is a match in the related columns. Only matched rows are included in the result set (no unmatched rows are shown).

අදාළ තීරුවල ගැලපීමක් ඇති වගු දෙකකින් හෝ වැඩි ගණනකින් පේළි ඒකාබද්ධ කරයි. ප්‍රතිඵල කට්ටලයට ඇතුළත් වන්නේ ගැලපෙන පේළි පමණි (නොගැලපෙන පේළි නොපෙන්වයි).

2) LEFT JOIN (or LEFT OUTER JOIN)

Returns all rows from the left table, along with the matched rows from the right table. If there is no match, NULL values are returned for columns from the right table, ensuring unmatched rows from the left table are still included.

වම් වගුවෙන් සියලුම පේළි, දකුණු වගුවෙන් ගැලපෙන පේළි සමඟින් ආපසු ලබා දෙයි. ගැලපීමක් නොමැති නම්, වම් වගුවෙන් නොගැලපෙන පේළි තවමත් ඇතුළත් කර ඇති බව සහතික කරමින් දකුණු වගුවේ තීරු සඳහා NULL අගයන් ලබා දෙනු ඇත.

3) RIGHT JOIN (or RIGHT OUTER JOIN)

Similar to LEFT JOIN, but returns all rows from the right table, along with the matched rows from the left table. Unmatched rows from the right table will still be included with NULL values for columns from the left table.

LEFT JOIN ට සමාන, නමුත් වම් වගුවෙන් ගැලපෙන පේළි සමඟින් දකුණු වගුවෙන් සියලුම පේළි ආපසු ලබා දෙයි. දකුණු වගුවේ ඇති නොගැලපෙන පේළි තවමත් වම් වගුවේ තීරු සඳහා NULL අගයන් සමඟ ඇතුළත් වේ.

4) FULL OUTER JOIN

Returns all rows when there is a match in either table. If there is no match, NULL values are returned for the missing side in both directions (left and right).

එක් වගුවක ගැලපීමක් ඇති විට සියලුම පේළි ආපසු ලබා දෙයි. ගැලපීමක් නොමැති නම්, මඟ හැරුණු කොටස සඳහා NULL අගයන් දිශාවන් දෙකෙහිම (වම සහ දකුණ) ලබා දෙනු ලැබේ.

5) CROSS JOIN

Produces a Cartesian product of the two tables, meaning every row from the first table is combined with every row from the second table. There is no condition for matching rows.

වගු දෙකේ Cartesian නිෂ්පාදනයක් නිපදවයි, එනම් පළමු වගුවේ ඇති සෑම පේළියක්ම දෙවන වගුවේ ඇති සෑම පේළියක්ම සමඟ ඒකාබද්ධ වේ. පේළි ගැලපීම සඳහා කොන්දේසියක් නොමැත.

Therefore, the correct answer is 2).

එබැවින්, නිවැරදි පිළිතුර 2) වේ.

6. Consider the Python code and answer the following questions.
පහත පයිතන් කේතය සලකා පහත ප්‍රශ්නවලට පිළිතුරු සපයන්න.

```
rows = int(input("Enter a number as you wish : "))
for i in range(rows):
    for j in range(rows - i - 1):
        print(" ", end="")
    for j in range(i + 1):
        print(j + 1, end="")
    for j in range(i, 0, -1):
        print(j, end="")
    print()
```

What is the primary purpose of the above Python code?

ඉහත පයිතන් කේතයේ මූලික අරමුණ කුමක් ද?

- 1) To generate a square pattern with numbers / ඉලක්කම් සහිත හතරැස් රටාවක් ජනනය කිරීමට.
- 2) To generate a diamond shape with asterisks / තරු ලකුණු සහිත දියමන්ති හැඩයක් ජනනය කිරීමට.
- 3) To generate a pyramid pattern with numbers / ඉලක්කම් සහිත පිරමීඩ රටාවක් ජනනය කිරීමට.
- 4) To create a right triangle pattern with numbers.
ඉලක්කම් සහිත සෘජුකෝණාස්‍රාකාර ත්‍රිකෝණ රටාවක් නිර්මාණය කිරීම.
- 5) To print Fibonacci numbers in a pattern / Fibonacci අංක රටාවක් මුද්‍රණය කිරීමට.

Answer: 3)

The given Python code generates a pyramid pattern of numbers. Here's how it works:

ලබා දී ඇති පයිතන් කේතය සංඛ්‍යා පිරමීඩ රටාවක් ජනනය කරයි. එය ක්‍රියා කරන ආකාරය මෙන්න:

Outer loop / පිටත ඉපය:

The for i in range(rows) loop controls the number of rows in the pyramid.

for i in range(rows) ඉපය පිරමීඩයේ පේළි ගණන පාලනය කරයි.

First inner loop / පළමු අභ්‍යන්තර ඉපය:

for j in range(rows - i - 1) prints the spaces before the numbers, ensuring the alignment for the pyramid shape.

for j in range(rows - i - 1) පිරමීඩයේ හැඩය සඳහා පෙළගැස්ම සහතික කරමින්, සංඛ්‍යාවලට පෙර හිස්තැන් මුද්‍රණය කරයි.

Second inner loop / දෙවන අභ්‍යන්තර ඉපය:

for j in range(i + 1) prints the numbers in increasing order, starting from 1.

for j in range(i + 1) අංක 1 සිට වැඩි වන අනුපිළිවෙලින් මුද්‍රණය කරයි.

Third inner loop / තුන්වන අභ්‍යන්තර ඉපය:

for j in range(i, 0, -1) prints the numbers in decreasing order, creating the second half of the pyramid.

for j in range(i, 0, -1) සංඛ්‍යා අඩුවන අනුපිළිවෙලින් මුද්‍රණය කරයි, පිරමීඩයේ දෙවන භාගය නිර්මාණය කරයි.

print():

Moves to the next line after completing each row.

එක් එක් පේළිය සම්පූර්ණ කිරීමෙන් පසු ඊළඟ පේළියට ගමන් කරයි.

This pattern creates a symmetric pyramid with numbers increasing up to the middle of the row and then decreasing.

මෙම රටාව පේළියේ මැද දක්වා සංඛ්‍යා වැඩි වන අතර පසුව අඩු වන සමමිතික පිරමීඩයක් නිර්මාණය කරයි.

Therefore, the correct answer is, 3).

එබැවින් නිවැරදි පිළිතුර නම්, 3).

7. What will be the output of the following Python code snippet?

පහත පයිතන් කේත කොටසෙහි ප්‍රතිදානය කුමක් වේද?

```
numbers = [1, 2, 3, 4, 5]
total = 0
def calculate_sum():
    global total
    for num in numbers:
        total += num
def squared(n):
    return n*n
calculate_sum()
squared_numbers = [squared(x) for x in range(1, total-9)]
print(squared_numbers)
```

- 1) 0
- 2) 15
- 3) [1, 2, 3, 4, 5]
- 4) [1, 4, 9, 16, 25]
- 5) Error: total is not defined

Answer: 4)

Explanation / පැහැදිලි කිරීම:

The output of the code snippet is [0] because there is an issue with the provided answer. The code calculates the sum of numbers, which results in 15. It then creates a list of squared values from 1 to 15 - 9, which is from 1 to 6.

සපයා ඇති පිළිතුරේ ගැටලුවක් ඇති බැවින් කේත කොටසෙහි ප්‍රතිදානය [0] වේ. කේතය 15 හි ප්‍රතිඵලයක් වන සංඛ්‍යා එකතුව ගණනය කරයි. ඉන්පසු එය 1 සිට 15 - 9 දක්වා වර්ග අගයන් ලැයිස්තුවක් නිර්මාණය කරයි, එය 1 සිට 6 දක්වා වේ.

Therefore, the list comprehension generates squared values of [1, 2, 3, 4, 5], resulting in [1, 4, 9, 16, 25]. The answer of 0 is incorrect based on the code logic.

එබැවින්, ලැයිස්තු අවබෝධය [1, 2, 3, 4, 5], වර්ග අගයන් ජනනය කරයි, එහි ප්‍රතිඵලයක් ලෙස [1, 2, 3, 4, 5] කේත තර්කය මත පදනම්ව 0 හි පිළිතුර වැරදිය.

Therefore, the correct answer is 4).

එබැවින්, නිවැරදි පිළිතුර 4) වේ.

8. In which scenario would an Executive Support System (ESS) be the least useful?
විධායක සහායක පද්ධතියක් (ESS) අඩුවෙන්ම ප්‍රයෝජනවත් වන්නේ කුමන තත්ත්වයකදීද?
- 1) A CEO analyzing the financial performance of a company
සමාගමක මූල්‍ය කාර්ය සාධනය විශ්ලේෂණය කරන ප්‍රධාන විධායක නිලධාරියාට
 - 2) A manager planning the company's next marketing campaign
සමාගමේ මිළඟ අලෙවිකරණ ව්‍යාපාරය සැලසුම් කරන කළමනාකරුවෙකුට
 - 3) A government official setting long-term economic goals
දිගු කාලීන ආර්ථික ඉලක්ක සැලසුම් කරන රජයේ නිලධාරියෙකුට
 - 4) A board deciding on a new international merger.
නව ජාත්‍යන්තර ඒකාබද්ධ කිරීමක් තීරණය කරන මණ්ඩලයකට
 - 5) A supervisor overseeing daily production activities
දෛනික නිෂ්පාදන කටයුතු අධීක්ෂණය කරන අධීක්ෂකයෙකුට

Answer: 5)

Explanation / පැහැදිලි කිරීම:

An Executive Support System (ESS) is designed for high-level strategic decision-making and is used primarily by executives like CEOs or government officials to analyze data and make long-term decisions. It is least useful for day-to-day operational tasks, such as a supervisor managing daily production activities, as these tasks require more hands-on, immediate management rather than strategic analysis.

විධායක ආධාරක පද්ධතියක් (ESS) ඉහළ මට්ටමේ උපාය මාර්ගික තීරණ ගැනීම සඳහා නිර්මාණය කර ඇති අතර දත්ත විශ්ලේෂණය කිරීමට සහ දිගුකාලීන තීරණ ගැනීමට ප්‍රධාන විධායක නිලධාරීන් හෝ රාජ්‍ය නිලධාරීන් වැනි විධායකයින් විසින් මූලික වශයෙන් භාවිතා කරයි. දෛනික නිෂ්පාදන ක්‍රියාකාරකම් කළමනාකරණය කරන අධීක්ෂකවරයෙකු වැනි වදිනෙදා ක්‍රියාකාරී කාර්යයන් සඳහා එය අවම වශයෙන් ප්‍රයෝජනවත් වේ, මන්ද මෙම කාර්යයන් සඳහා උපාය මාර්ගික විශ්ලේෂණයට වඩා ක්ෂණික කළමනාකරණයක් අවශ්‍ය වේ.

Therefore, the answer is 5) A supervisor overseeing daily production activities.

එබැවින්, පිළිතුර 5) දෛනික නිෂ්පාදන කටයුතු අධීක්ෂණය කරන අධීක්ෂකවරයෙකු වේ.

9. Which PHP super global variable is used to access form data sent with the GET method?
GET ක්‍රමය සමඟ යවන ලද පෝරම දත්ත වෙත ප්‍රවේශ වීමට භාවිතා කරන PHP සුපිරි ගෝලීය විචල්‍යය කුමක්ද?
- 1) \$GLOBALS 2) \$_SERVER 3) \$_GET 4) \$_POST 5) \$_REQUEST

Answer: 3)

Let's consider each option one by one. / එක් එක් විකල්පය එකින් එක සලකා බලමු.

1) \$GLOBALS

According to this option, it's false. The \$GLOBALS super global variable is used to access global variables from anywhere in the PHP script, not specifically for accessing form data sent with the GET method.

මෙම විකල්පයට අනුව, එය අසත්‍ය වේ. \$GLOBALS සුපිරි ගෝලීය විචල්‍යය PHP ස්ක්‍රිප්ට් හි ඕනෑම තැනක සිට ගෝලීය විචල්‍ය වෙත ප්‍රවේශ වීමට භාවිතා කරයි, විශේෂයෙන් GET ක්‍රමය සමඟ යවන ලද පෝරම දත්ත වෙත ප්‍රවේශ වීම සඳහා නොවේ.

2) \$_SERVER

According to this option, it's false. The \$_SERVER super global variable contains information about headers, paths, and script locations, but it is not used for accessing form data sent with the GET method.

මෙම විකල්පයට අනුව, එය අසත්‍ය වේ. \$_SERVER සුපිරි ගෝලීය විචල්‍යයේ ශීර්ෂක, මාර්ග සහ ස්ක්‍රිප්ට් ස්ථාන පිළිබඳ තොරතුරු අඩංගු වේ, නමුත් එය GET ක්‍රමය සමඟ යවන ලද පෝරම දත්ත වෙත ප්‍රවේශ වීම සඳහා භාවිතා නොවේ.

3) \$_GET

According to this option, it's true. The \$_GET super global variable is specifically used to access form data sent with the GET method. It retrieves data from the URL query string when a form is submitted using the GET method.

මෙම විකල්පය අනුව, එය සත්‍ය වේ. GET ක්‍රමය සමඟ යවන ලද පෝරම දත්ත වෙත ප්‍රවේශ වීමට \$_GET සුපිරි ගෝලීය විචල්‍යය විශේෂයෙන් භාවිතා වේ. GET ක්‍රමය භාවිතයෙන් පෝරමයක් ඉදිරිපත් කළ විට එය URL විමසුම් තත්ත්වයෙන් දත්ත ලබා ගනී.

4) \$_POST

According to this option, it's false. The \$_POST super global variable is used to access form data sent with the POST method, not the GET method.

මෙම විකල්පයට අනුව, එය අසත්‍ය වේ. GET ක්‍රමය නොව POST ක්‍රමය සමඟ යවන ලද පෝරම දත්ත වෙත ප්‍රවේශ වීමට \$_POST සුපිරි ගෝලීය විචල්‍යය භාවිතා වේ.

5) \$_REQUEST

According to this option, it's false. The \$_REQUEST super global variable can be used to collect data sent with both GET and POST methods, but it is not specifically for GET method data only.

මෙම විකල්පයට අනුව, එය අසත්‍ය වේ. GET සහ POST යන ක්‍රම දෙකෙන්ම යවන ලද දත්ත රැස් කිරීමට \$_REQUEST සුපිරි ගෝලීය විචල්‍යය භාවිතා කළ හැක, නමුත් එය විශේෂයෙන් GET ක්‍රම දත්ත සඳහා පමණක් නොවේ.

So, the correct answer is 3).

ඒ අනුව නිවැරදි පිළිතුර වන්නේ 3) වේ'

10. What is the correct internal CSS code to make only the word 'from' red in the output of the HTML code below?

පහත HTML කේතයෙන් ප්‍රතිදානය වන 'from' යන වචනය පමණක් රතු පැහැ කිරීමට අදාළ නිවැරදි අභ්‍යන්තර CSS කේතය කුමක් ද?

```
<body>
  <h1 class="a">ict</h1>
  <h2 class="b">from</h2>
  <h3 class="c">abc</h3>
  <p class="a">hello</p>
  <h3 class="b">world</h3>
  <h2 class="c">2024</h2>
</body>
```

- 1) <style>.b{color="red"}</style>
- 2) <style>h2{color="red"}</style>
- 3) <style>b.h2{color: red;}</style>
- 4) <style>h2,b{color: red;}</style>
- 5) <style>h2.b{color: red;}</style>

Answer : 5)

Here we want to make only the word "from" red, which is wrapped inside an <h2> tag with the class "b". Here you can use CSS class selector. A class selector in CSS is used to apply styles to HTML elements that have a specific class attribute.

මෙහිදී CSS class selector භාවිතා කළ හැක. විශේෂිත පන්ති ගුණාංගයක් ඇති HTML මූලාංග සඳහා විලාස යෙදීමට CSS class selector භාවිතා කරයි. මෙහිදී අපට අවශ්‍ය වන්නේ "b" පන්තිය සහිත <h2> උපුලනයක් තුළ ඇති "from" යන වචනය පමණක් රතු කිරීමටයි.

• Selector Explanation

h2:

This part of the selector targets all <h2> elements in the document.

Selector එකෙහි මෙම කොටස ලේඛනයේ ඇති සියලුම <h2> මූලාංග ඉලක්ක කරයි.

.b:

This part of the selector targets any element with the class "b".

එකෙහි මෙම කොටස "b" පන්තිය සහිත ඕනෑම මූලාංගයක් ඉලක්ක කරයි.

• Combined Selector (h2.b):

h2.b:

When you combine these two parts (h2 and .b), you create a selector that specifically targets <h2> elements that also have the class "b".

ඔබ මෙම කොටස් දෙක (h2 සහ .b) එකතුවීම කළ විට, ඔබ "b" පන්තිය ඇති <h2> මූලාංග විශේෂයෙන් ඉලක්ක කරන selector එකක් සාදයි.

Therefore, <style>h2.b{color: red;}</style> targets an <h2> element with the class "b". Since only the word "from" is inside an <h2> tag with the class "b", this option will correctly apply the red color to that word only.

එබැවින්, <style>h2.b{color: red;}</style> "b" පන්තිය සමඟ <h2> මූලාංගයක් ඉලක්ක කරයි. "b" පන්තිය සහිත <h2> උපුලනය තුළ "from" යන වචනය පමණක් ඇති බැවින්, මෙම පිළිතුර එම වචනයට පමණක් රතු පැහැය නිවැරදිව යොදනු ඇත.

So, the correct answer is 5).

එමනිසා, නිවැරදි පිළිතුර 5) වේ.

1.(a)

- i. A student needs to calculate the volume of a solid cylinder. To obtain precise measurements of the diameter and the height, the student uses a vernier caliper. Assume that the zero errors (baseline errors) of the instrument are negligible.

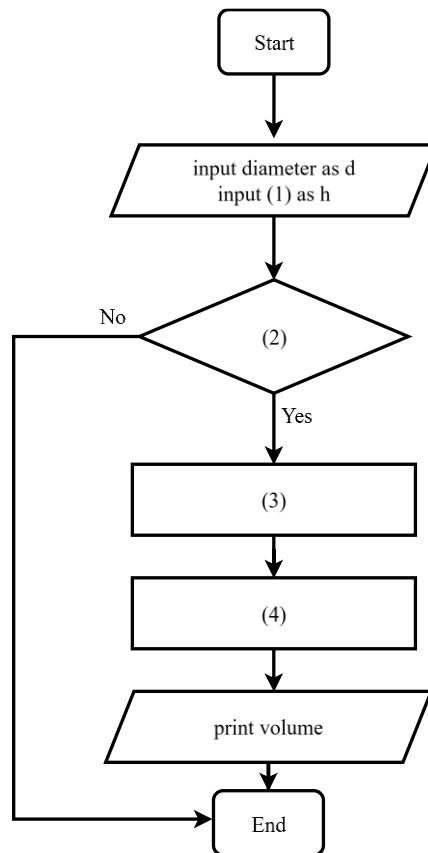
ශිෂ්‍යයෙකුට සහ සිලින්ඩරයක පරිමාව ගණනය කිරීමට අවශ්‍ය වී ඇත. එහි විෂ්කම්භය සහ උස නිවැරදිව මනන ගැනීම සඳහා ශිෂ්‍යයා වර්නියර් කැලිපරයක් භාවිතා කරයි. උපකරණයේ ශුන්‍ය දෝෂ නොගිනිය හැකි තරම් අවම යැයි උපකල්පනය කරන්න.

$$\text{Volume} = 3.14 \times (\text{Radius})^2 \times \text{Height}$$

$$\text{පරිමාව} = 3.14 \times (\text{අරය})^2 \times \text{උස}$$

Fill the flowchart to represent the algorithm that inputs the diameter (d) and height (h) of the cylinder, calculates the radius(r), determines the volume, and finally outputs the result.

සිලින්ඩරයේ විෂ්කම්භය d සහ උස (h) ඇතුළත්ය (input) කර, අරය (r) ගණනය කර, පරිමාව සොයා, අවසානයේ ප්‍රතිදානය ලබා දෙන ඇල්ගොරිතමයට අදාළ ගැලීම් සටහන සම්පූර්ණ කරන්න



(1) height
(3) r= d/2

(2) if d AND h > 0 ?
(4) volume= 3.14 x r x r x h

ii. What will be the output printed by the program?

මෙම වැඩසටහන මගින් ලබාදෙන ප්‍රතිදානය කුමක්ද?

`n = 2026`

`r = 0`

`while n != 0:`

`d = n % 10`

`r = r * 10 + d`

`n //= 10`

`print(r)`

6202

Iteration	n!=0	d=n(mod10)	r=(r×10)+d	n=n//10
Initial	-	-	0	2026
Step 1	True	2026 (mod 10) = 6	(0 × 10) + 6 = 6	2026 // 10 = 202
Step 2	True	202 (mod 10) = 2	(6 × 10) + 2 = 62	202 // 10 = 20
Step 3	True	20 (mod 10) = 0	(62 × 10) + 0 = 620	20 // 10 = 2
Step 4	True	2 (mod 10) = 2	620 × 10) + 2 = 6202	2 // 10 = 0
End	False	-	Final: 6202	-

iii. Consider the following Python code segment which is intended to calculate the sum of the digits of a given number n. Note that line numbers are provided on the left. Identify the line numbers containing logical or syntax errors that would prevent the code and correct the error with the line number and write it down.

දී ඇති n සංඛ්‍යාවක ඉලක්කම්වල එකතුව සෙවීම සඳහා නිර්මාණය කරන ලද පහත Python කේත ඛණ්ඩය සලකා බලන්න. වම් පසින් පේළි අංක දක්වා ඇත.මෙම කේතය බලාපොරොත්තු වූ පරිදි ක්‍රියාත්මක වීම වළක්වන තර්ක දෝෂ (logical errors) හෝ වාක්‍ය වින්‍යාස දෝෂ (syntax errors) පවතින පේළි අංක හඳුනාගෙන පේළි අංකය සමග දෝශය නිවැරදි කර ලියන්න.

1 `n = 456`

2 `total = 0`

3 `while n > 0:`

4 `digit = n % 10`

5 `total + digit`

6 `n = n // 10`

7 `print(total)`

Line 4 & 6 - IndentationError

Line 5 - total = total + digit

Explanation / පැහැදිලි කිරීම:

This code contains multiple errors related to Python syntax and logic:

- Line 4 & 6 (Indentation Error):** In Python, statements inside a while loop must be indented. Lines 4, 5, and 6 are not indented, which will cause an IndentationError.
4 සහ 6 පේළි (Indentation Error): Python හි while ඉපයක් තුළ ඇති ප්‍රකාශන indentation ලක් කළ යුතුය. 4, 5, සහ 6 පේළි indent එකකට ලක් කර නොමැති බැවින් IndentationError දෝෂය ඇති වේ.
- Line 5 (Logical/Assignment Error):** The expression `total + digit` calculates the sum but does not store it. To update the variable, it must be `total = total + digit` or `total += digit`.
5 වන පේළිය (Logical/Assignment Error): `total + digit` මගින් එකතුව ගණනය කළද එය නැවත විචල්‍යය තුළ තැන්පත් නොකරයි. විචල්‍යය යාවත්කාලීන වීමට නම් එය `total = total + digit` හෝ `total += digit` ලෙස ලිවිය යුතුය.

- iv. Write down two characteristics of high level computer programming languages in comparison with low level compute programming languages.

පහළ මට්ටමේ පරිගණක ක්‍රමලේඛන භාෂා හා සැසඳීමේදී ඉහළ මට්ටමේ පරිගණක ක්‍රමලේඛන භාෂාවල ලක්ෂණ දෙකක් ලියන්න.

Readability and Maintainability: HLLs use human-readable syntax (e.g., if, while, print). This makes the code easier to write, read, and maintain over long periods.

කියවීමේ හැකියාව සහ නඩත්තු කිරීමේ හැකියාව: HLLs මිනිසුන්ට කියවිය හැකි වාක්‍ය ඛණ්ඩය භාවිතා කරයි (උදා: if, while, print). මෙය කේතය දිගු කාලයක් පුරා ලිවීමට, කියවීමට සහ නඩත්තු කිරීමට පහසු කරයි.

Machine Independence (Portability): Programs written in a high-level language can run on different hardware architectures (e.g., Intel or ARM) with little to no modification, provided a compatible compiler or interpreter is available.

යන්ත්‍ර ස්වාධීනත්වය : අනුකූල සම්පාදකයක් හෝ පරිවර්තකයක් තිබේ නම්, ඉහළ මට්ටමේ භාෂාවකින් ලියා ඇති වැඩසටහන් විවිධ දෘඩාංග ගෘහ නිර්මාණ ශිල්පයන් (උදා: Intel හෝ ARM) මත යුළ හෝ කිසිදු වෙනස් කිරීමකින් තොරව ක්‍රියාත්මක විය හැකිය.

Built-in Functions and Libraries: They come with extensive pre-written modules that handle complex tasks like string manipulation, file I/O, and mathematical operations, saving developers significant time.

බිල්ට්-ඉන් කාර්යයන් සහ ප්‍රස්තකාල: ඒවා සංවර්ධකයින්ට සැලකිය යුතු කාලයක් ඉතිරි කරමින්, string හැසිරවීම, ගොනු (I/O) සහ ගණිතමය මෙහෙයුම් වැනි සංකීර්ණ කාර්යයන් හසුරුවන පුළුල් පුර්ව-ලිඛිත මොඩියුල සමඟ පැමිණේ.

Automated Memory Management: Most modern HLLs (like Java or Python) which automatically frees up memory that is no longer in use, preventing memory leaks without manual intervention.

ස්වයංක්‍රීය මතක කළමනාකරණය: බොහෝ නවීන HLLs (ජාවා හෝ පයිතන් වැනි), එය තවදුරටත් භාවිතයේ නොමැති මතකය ස්වයංක්‍රීයව නිදහස් කරයි, අතින් මැදිහත් වීමකින් තොරව මතක කාන්දු වීම වළක්වයි.

Requirement of a Translator: Since CPUs only understand binary, HLLs must be converted into machine code using a Compiler or an Interpreter.

පරිවර්තකයෙකුගේ අවශ්‍යතාවය: CPUs ද්විමය පමණක් තේරුම් ගන්නා බැවින්, HLLs සම්පාදකයක් හෝ පරිවර්තකයක් භාවිතයෙන් යන්ත්‍ර කේතයක් බවට පරිවර්තනය කළ යුතුය.