සියලු ම හිමිකම් ඇවිරිණි /All Rights Reserved]

Ravindu Bandaranayake - ictfromabc

Information and Communication Technology
තොරතුරු හා සන්නිවේදන තාක්ෂණය

2026 QUIZ NO 401 MARKING

2026/01/15
ictfromabc
Ravindu Bandaranayake

1. Modern computers use three levels of cache memory, L1, L2, and L3. Correct answer regarding the data capacity of these?
නවීන පරිගණකවල වාරක මතක වර්ග මට්ටම් තුනක් L1, L2 සහ L3 ලෙස යොදා ගනු ලැබේ. මේවායේ දත්ත ධාරිතාව සම්බන්ධයෙන් නිවැරදිව දක්වා ඇති පිළිතුර කුමක්ද?

1) L1 > L2 > L3 2) L1 = L2 = L3 3) L1 < L2 < L3 4) L1 < L2 = L3 5) L1 = L2 < L3

Answer : 3)

L1 Cache
This is the smallest and fastest type of cache memory, located closest to the CPU cores. It typically has the smallest capacity, measured in kilobytes (KB).
ප්‍රමය CPU මධ්‍යයට ආසන්නව පිහිටා ඇති කුඩාම සහ වේගවත්ම වාරක මතකය වේ. එය සාමාන්‍යයෙන් කුඩාම ධාරිතාවය, කිලෝබයිට් (KB) වලින් මනිනු ලැබේ.

L2 Cache
This cache is larger than L1 but smaller than L3. It is also relatively fast but not as fast as L1. It is measured in megabytes (MB) or large kilobytes.
මෙම වාරකය L1 ට වඩා විශාල නමුත් L3 ට වඩා කුඩා වේ. එය ද සාපේක්ෂව වේගවත් නමුත් L1 තරම් වේගවත් නොවේ. එය මෙගාබයිට් (MB) හෝ විශාල කිලෝබයිට් වලින් මනිනු ලැබේ.

L3 Cache
This is the largest cache in terms of capacity but the slowest compared to L1 and L2. It is shared among multiple cores in a multi-core processor and is typically measured in megabytes (MB).
මෙය ධාරිතාව අනුව විශාලතම වාරකය වන නමුත් L1 සහ L2 වලට සාපේක්ෂව මන්දගාමී වේ. එය බහු හර සකසනයක බහු හරයන් අතර බෙදාගෙන ඇති අතර සාමාන්‍යයෙන් මෙගාබයිට් (MB) වලින් මනිනු ලැබේ.

Therefore, the correct answer is 3) L1 < L2 < L3.
එබැවින්, නිවැරදි පිළිතුර නම් 3) L1 < L2 < L3.

2. Convert the decimal value 96.3125_{10} to binary.
96.3125_{10} දශමය අගය ද්විමය බවට පරිවර්තනය කරන්න.

1) 110000.011_2 2) 1100000.0101_2 3) 1011111.0101_2 4) 1100000.101_2 5) 1100000.1101_2

Answer: 2)

First, let's convert the non-fractional part to binary.
පළමුව දශම රහිත කොටස ද්විමය බවට පරිවර්තනය කරමු.

96_{10}							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1
0	1	1	0	0	0	0	0
1100000_2							

Now let's convert the fractional part to binary.
දැන් දශම සහිත කොටස ද්විමය බවට පරිවර්තනය කරමු.

	.3125	x 2
0	.6250	x 2
1	.2500	x 2
0	.5000	x 2
1	.0000	

$0.3125_{10} = 0101_2$

Accordingly / ඒ අනුව,
$96.3125_{10} = 1100000.0101_2$

So, the correct answer number is 2.
එමනිසා, නිවැරදි පිළිතුරු අංකය 2 වේ.

3. What is the simplest form of (X + Y)(X + Z)?
(X + Y)(X + Z) හි සරලම ආකාරය කුමක් ද?

1) X 2) X+Y.Z 3) X.(Y+Z) 4) X+Y+Z 5) X.Y.Z

Answer : 2)

This boolean expression can be simplfied as follows.
මෙම බූලියානු ප්‍රකාශනය පහත ආකාරයට සුළු කළ හැකිය.

(X + Y)(X + Z)	
XX + XZ + (X + Y)Z	Distribution law විඝටන න්‍යාය
X + XZ + (X + Y)Z	Idempotant law තද්වභාවී න්‍යාය
X + (X + Y)Z	Redundancy Law සමතිරික්ත න්‍යාය
X + YX + YZ	Distribution law විඝටන න්‍යාය
X + YZ	Redundancy Law සමතිරික්ත න්‍යාය

Therefore the correct answer is **2)**
එබැවින් නිවැරදි පිළිතුර **2)** වේ

4. Which of the following statements about SQL JOIN types is correct?
SQL JOIN වර්ග පිළිබඳ පහත සඳහන් කුමන ප්‍රකාශය නිවැරදිද?

A. An INNER JOIN only returns rows when there is a match in both tables.
INNER JOIN එකක් වගු දෙකෙහිම ගැලපීමක් ඇති විට පමණක් පේළි ලබා දෙයි.

B. A LEFT JOIN returns all rows from the left table, even if there is no match in the right table.
LEFT JOIN එකක් දකුණු වගුවේ ගැළපීමක් නොමැති වුවද, වම් වගුවෙන් සියලුම පේළි ආපසු ලබා දෙයි.

C. A FULL JOIN returns rows when there is a match in either table.
FULL JOIN එක වගු දෙකෙහිම ගැලපීමක් ඇති විට පේළි ලබා දෙයි.

1) A only. A පමණි. 2) B only. B පමණි. 3) C only. C පමණි. 4) A and B only. A හා B පමණි. 5) All of the above. ඉහත සියල්ලම.

Answer: 5)

Each of the above statement accurately describes a different type of SQL join.
ඉහත සෑම ප්‍රකාශයක්ම විවිධ වර්ගයේ SQL join කිරීමක් නිවැරදිව විස්තර කරයි.

Statement / ප්‍රකාශය A
This statement is correct. An INNER JOIN will only include rows with matches in both tables.
මෙම ප්‍රකාශය නිවැරදියි. INNER JOIN එකකට දකුණු වගුවේ ගැළපෙන පේළි නොමැති වුවද, වම් වගුවේ ඇති සියලුම පේළි ඇතුළත් වේ.

Statement / ප්‍රකාශය B
This statement is also correct. A LEFT JOIN includes all rows from the left table, even if they don't have matching rows in the right table.
මේ ප්‍රකාශයත් නිවැරදි වේ. LEFT JOIN එකකට දකුණු වගුවේ ගැළපෙන පේළි නොමැති වුවද, වම් වගුවේ ඇති සියලුම පේළි ඇතුළත් වේ.

Statement / ප්‍රකාශය C
This statement is correct. A FULL JOIN returns all rows where there are matches in either the left or right table, including unmatched rows from both tables.
මෙම ප්‍රකාශය නිවැරදියි. FULL JOIN එකක් වගු දෙකෙන්ම නොගැලපෙන පේළි ඇතුළුව වම් හෝ දකුණු වගුවේ ගැලපීම් ඇති සියලුම පේළි ලබා දෙයි.

All the above statements become true.
ඉහත ප්‍රකාශ සියල්ල සත්‍ය වේ.

So, the correct answer is 5).
එබැවින්, නිවැරදි පිළිතුර 5) වේ.

5. Which SQL query is used to retrieve all records from the Customers table where the customer's name starts with the letter "A"?
පාරිභෝගිකයාගේ නම A අකුරෙන් ආරම්භ වන Customers වගුවෙන් සියලුම වාර්තා ලබා ගැනීමට භාවිතා කරන SQL විමසුම වන්නේ කුමක්ද?

1) `SELECT * FROM Customers WHERE Name LIKE 'A%';`
2) `SELECT * FROM Customers WHERE Name LIKE 'A';`
3) `SELECT * FROM Customers WHERE Name = 'A';`
4) `SELECT * FROM Customers WHERE Name LIKE '%A';`
5) `SELECT * FROM Customers WHERE Name LIKE '%A%';`

Answer: 1)

The correct answer to this question is / මෙම ප්‍රශ්නයට නිවැරදි පිළිතුර වන්නේ:
1) `SELECT * FROM Customers WHERE Name LIKE 'A%';`

Explanation / පැහැදිලි කිරීම:

The LIKE operator is used in SQL for pattern matching.
රටාව ගැලපීම සඳහා SQL හි LIKE ක්‍රියාකරු භාවිතා කරයි.

The % wildcard in `LIKE 'A%'` means:
`LIKE 'A%'` හි % wildcard යන්නෙන් අදහස් වන්නේ:

- The string must start with 'A'.
  A සමඟ ආරම්භ විය යුතුය.

- Any number of characters may follow.
  ඕනෑම අක්ෂර ගණනක් අනුගමනය කළ හැකිය.

This will retrieve all customers whose names start with the letter 'A'
මෙය A අකුරෙන් ආරම්භ නම් ඇති සියලුම ගනුදෙනුකරුවන් ලබා ගනී

Explanation of incorrect options / වැරදි විකල්ප පැහැදිලි කිරීම

Option / විකල්ප 2)
The LIKE operator is used for pattern matching, but "A" without wildcards means only exact matches. This query will return customers whose name is exactly "A", which is incorrect for finding names starting with "A".
රටා ගැලපීම සඳහා LIKE ක්‍රියාකරු භාවිතා කරයි, නමුත් wildcards නොමැතිව "A" යන්නෙන් අදහස් කරන්නේ හරියටම ගැලපීම් පමණි. මෙම විමසුම හරියටම "A" නම ඇති customers ආපසු ලබා දෙනු ඇත, එය "A" වලින් ආරම්භ වන නම් සොයා ගැනීම සඳහා වැරදිය.

Option / විකල්ප 3)
The = operator is for exact matches, meaning only records where the entire name is "A" will be returned. This is incorrect because need names starting with "A", not just "A".
= ක්‍රියාකරු නිශ්චිත ගැලපීම් සඳහා වන අතර, එනම් සම්පූර්ණ නම "A" වන වාර්තා පමණක් ආපසු ලබා දෙනු ඇත. මෙය වැරදියි, මන්ද අපට "A" වලින් පටන් ගන්නා නම් අවශ්‍ය වන අතර A වලින් පමණක් නොවේ.

Option / විකල්ප 4)
The "%A" pattern means names that end with "A", not start with "A". This is incorrect since we need names starting with "A".
"%A" රටාව යන්නෙන් අදහස් කරන්නේ "A" සමඟ අවසන් වන නම් මිස "A" සමඟ ආරම්භ නොවන බවයි. A සමඟ ආරම්භ වන නම් අපට අවශ්‍ය බැවින් මෙය වැරදියි.

6. Which of the following SQL statements is used to insert a new record into a table students with columns name and age?
නම සහ වයස තිරු සහිත student වගුවකට නව වාර්තාවක් ඇතුළු කිරීමට පහත සඳහන් SQL ප්‍රකාශනවලින් කුමන එක භාවිතා කරයිද?

1) `INSERT INTO students VALUES ('John', 25);`
2) <u>`INSERT INTO students (name, age) VALUES ('John', 25);`</u>
3) `INSERT NEW INTO students (name, age) VALUES ('John', 25);`
4) `INSERT INTO students ('name', 'age') VALUES ('John', 25);`
5) `INSERT INTO students VALUES (name = 'John', age = 25);`

Answer : 2)

Explanation / පැහැදිලි කිරීම

The correct SQL statement to insert a new record into the `students` table with specific columns `name` and `age` is:
නිශ්චිත තිරු `name` සහ `age` සහිත `students` වගුවට නව වාර්තාවක් ඇතුළත් කිරීම සඳහා නිවැරදි SQL ප්‍රකාශය වන්නේ:

```
INSERT INTO students (name, age) VALUES ('John', 25);
```

This statement explicitly specifies the columns into which values should be inserted, ensuring the data aligns correctly with the table structure. The columns `name` and `age` are listed in parentheses after the table name, followed by their respective values in the `VALUES` clause. This approach avoids ambiguity and is preferred when not all columns in the table are being populated.
මෙම ප්‍රකාශය මඟින් අගයන් ඇතුළත් කළ යුතු තිරු පැහැදිලිව සඳහන් කරයි, දත්ත වගු ව්‍යුහය සමඟ නිවැරදිව පෙළගස්වන බව සහතික කරයි. `name` සහ `age` තිරු වගු නාමයට පසුව වරහන් තුළ ලැයිස්තුගත කර ඇති අතර, පසුව `VALUES` වගන්තියේ ඒවායේ අදාළ අගයන් ඇත. මෙම ප්‍රවේශය අපැහැදිලි බව වළක්වන අතර වගුවේ සියලුම තිරු පුරවා නොමැති විට වඩාත් සුදුසුය.

7. Consider the following Python code.
පහත පයිතන් කේතය සලකා බලන්න.

```
def func(text):
    result = ""
    for char in text:
        result = char + result
    return result
print(func("Hello Python"))
```

What will be the output of the given code?
පහත කේතයේ ප්‍රතිදානය කුමක් වේද?

1) `Hello Python` 2) `nohtyP olleH` 3) `Python Hello`
4) `nohtyPHello` 5) `olleH Python`

Answer: 2)

The function `func(text)` reverses the input string by iterating through each character of the string and prepending it to the variable result. Here's how it works:
`func(text)` යන ශ්‍රිතය string එකෙහි එක් එක් අක්ෂර හරහා පුනරාවර්තනය කිරීමෙන් සහ විචල්‍ය ප්‍රතිඵලයට පෙරාතුව ආදාන string එක ආපසු හරවයි. එය ක්‍රියා කරන ආකාරය මෙලෙස වේ.

Initially, result is an empty string.
මුලදී, ප්‍රතිඵලය හිස් string එකකි.

For each character in text, the new character is added to the beginning of result.
පාඨයෙහි ඇති සෑම අක්ෂරයක් සඳහාම, නව අනුලක්ෂණයක් ප්‍රතිඵලයේ ආරම්භයට එක් කෙරේ.

This effectively reverses the order of characters in the input string text.
මෙය ආදාන string එක පාඨයෙහි අනුලක්ෂණ අනුපිළිවෙල ඵලදායී ලෙස ආපසු හරවයි.

When `func("Hello Python")` is called, the function reverses the string and returns `"nohtyP olleH"`.
`func("Hello Python")` ලෙස හැඳින්වූ විට, ශ්‍රිතය string එක ආපසු හරවා `"nohtyP olleH"` ලබා දෙයි.

The correct answer is 2).
නිවැරදි පිළිතුර 2) වේ.

8. What will be the output of the following code?
පහත කේතයේ ප්‍රතිදානය කුමක් වේද ?

```
thought, body = 24, -7
if thought > 0:
    if body < 0:
        print("Scientific Thought Prevails")
    else:
        print("Body of Knowledge Dominates")
else:
    print("Undefined Thinking")
```

1) `Scientific Thought`
2) `Scientific Thought Prevails`
3) `Scientific Prevails`
4) `Error`
5) `None`

Answer : 2)

Explanation / පැහැදිලි කිරීම

The code evaluates the values of `thought` and `body` to determine the output. The first condition `thought > 0` is checked, which evaluates to `True` because `thought` is `24`, a positive number. Inside this block, the nested condition `body < 0` is checked, which is also `True` since `body` is `-7`, a negative number. As both conditions are satisfied, the statement `print("Scientific Thought Prevails")` is executed. Therefore, the output of the code is "`Scientific Thought Prevails`".
කේතයේ ප්‍රතිදානය තීරණය කිරීම සඳහා `thought` සහ `body` අගයන් ඇගයීමට ලක් කරයි. පළමු කොන්දේසිය `thought > 0` පරීක්ෂා කර ඇත, එය `thought 24`, ධන අංකයක් නිසා `True` ලෙස ඇගයීමට ලක් කෙරේ. මෙම කොටස තුළ, nested තත්ත්වය `body < 0` පරීක්ෂා කරනු ලැබේ, එය ද `True` වන බැවින් `body -7`, සෘණ අංකයකි. කොන්දේසි දෙකම තෘප්තිමත් වන බැවින්, `print("Scientific Thought Prevails")` ප්‍රකාශය ක්‍රියාත්මක වේ. එබැවින්, කේතයේ ප්‍රතිදානය වන්නේ "`Scientific Thought Prevails`" යන්නයි.

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

9. What happens if you try to open a file for writing (`'w'`) that already exists?
ඔබ දැනටමත් පවතින ගොනුවක් (`'w'`)ලිවීම සඳහා විවෘත කිරීමට උත්සාහ කළහොත් කුමක් සිදුවේද?

1) Appends data to the existing file / පවතින ගොනුවට දත්ත එකතු කරයි.
2) Throws an error / දෝෂයක් ඇති කරයි.
3) Deletes the file / ගොනුව මකා දමයි.
4) Overwrites the existing file / පවතින ගොනුව මත ලියයි (Overwrites).
5) Opens the file without any changes / කිසිදු වෙනසක් නොමැතිව ගොනුව විවෘත කරයි.

Answer : 4)

Explanation / පැහැදිලි කිරීම

When you open a file in Python using the 'w' mode (write mode), the existing content of the file is completely erased, and the file is overwritten with new data. This means that if the file already exists, all its current data will be removed, and a fresh, empty file with the same name is created, ready for writing. If the file does not exist, a new file will be created.
ඔබ 'w' මාදිලිය (ලිවීමේ මාදිලිය) භාවිතා කර Python හි ගොනුවක් විවෘත කරන විට, ගොනුවේ පවතින අන්තර්ගතය සම්පූර්ණයෙන්ම මකා දමනු ලබන අතර, ගොනුව නව දත්ත සමඟ උඩින් ලියවේ. මෙයින් අදහස් කරන්නේ ගොනුව දැනටමත් තිබේ නම්, එහි සියලුම වත්මන් දත්ත ඉවත් කරනු ලබන අතර, එම නමම ඇති නැවුම්, හිස් ගොනුවක් නිර්මාණය වන අතර, ලිවීමට සූදානම් වේ. ගොනුව නොමැති නම්, නව ගොනුවක් සාදනු ලැබේ.

Other options do not describe the behavior of the 'w' mode accurately.
වෙනත් විකල්ප 'w' මාදිලියේ හැසිරීම නිවැරදිව විස්තර නොකරයි.

- **Appends data to the existing file / පවතින ගොනුවට දත්ත එක් කරයි:**
This describes the 'a' mode, which appends new data to the end of the file while preserving the existing content.
පවතින අන්තර්ගතය සංරක්ෂණය කරමින් ගොනුවේ අවසානයට නව දත්ත එකතු කරන 'a' මාදිලිය මෙය විස්තර කරයි.

- **Throws an error / දෝෂයක් ඉස්මතු කරයි:**
The 'w' mode does not throw an error unless there is a problem like missing write permissions or the file is locked.
ලිවීමේ අවසර නැතිවීම හෝ ගොනුව අගුළු දමා තිබීම වැනි ගැටලුවක් තිබේ නම් මිස 'w' මාදිලිය දෝෂයක් ඇති නොකරයි.

- **Deletes the file / ගොනුව මකා දමයි:**
While the content of the file is erased, the file itself remains; it is not deleted but overwritten.
ගොනුවේ අන්තර්ගතය මකා දැමූ අතර, ගොනුව ම පවති❑ එය මකා නැත නමුත් උඩින් ලියා ඇත.

- **Opens the file without any changes / කිසිදු වෙනසක් නොමැතිව ගොනුව විවෘත කරයි:**
This is incorrect because the 'w' mode always modifies the file by clearing its content.
මෙය වැරදියි මන්ද 'w' මාදිලිය සෑම විටම එහි අන්තර්ගතය ඉවත් කිරීමෙන් ගොනුව වෙනස් කරයි.

Therefore, the correct answer is 4).
එබැවින්, නිවැරදි පිළිතුර 4) වේ.

10. You're designing a webpage for a music festival. Which HTML5 element would you use to embed an audio player for the festival's theme song?
ඔබ සංගීත උළෙලක් සඳහා වෙබ් පිටුවක් නිර්මාණය කරයි. උත්සවයේ තේමා ගීතය සඳහා ශ්‍රව්‍ය ධාවකයක් යෙදීමට ඔබ භාවිතා කරන්නේ කුමන HTML5 මූලාංගයද?

1) <audio> 2) <sound> 3) <music> 4) <player> 5) <embed>

Answer: 1)

The correct answer is / නිවැරදි පිළිතුර නම්: 1) <audio>

Explanation / පැහැදිලි කිරීම:

The <audio> element in HTML5 is specifically designed to embed audio content, such as music or sound effects, directly into a webpage. It provides built-in playback controls, including play, pause, and volume adjustment.
HTML5 හි <audio> element එක විශේෂයෙන් නිර්මාණය කර ඇත්තේ සංගීතය හෝ ශබ්ද ප්‍රයෝග වැනි ශ්‍රව්‍ය අන්තර්ගතයන් සෘජුවම වෙබ් පිටුවකට කාවැද්දීමට ය. එය play, pause සහ volume adjustment ඇතුළුව ගොඩනඟන ලද playback controls සපයයි.

Example / උදාහරණ:

```
<audio controls>
  <source src="festival-theme.mp3" type="audio/mpeg">
  Your browser does not support the audio element.
</audio>
```

- The `controls` attribute ensures the player is visible with playback options.
`Controls` ගුණාංගය මඟින් playback options සමඟින් player එක දෘශ්‍යමාන බව සහතික කරයි.

- The <source> tag allows multiple formats for better browser compatibility.
<source> ටැගය වඩාත් හොඳ බ්‍රවුසර ගැළපුම සඳහා බහු ආකෘති වලට ඉඩ දෙයි.

- If the browser does not support the `<audio>` tag, fallback text will be displayed.
  බ්‍රවුසරය `<audio>` ටැගයට සහය නොදක්වන්නේ නම්, fallback text පෙළ දර්ශනය වේ.

Why the other options are incorrect / වෙනත් විකල්ප වැරදි ඇයි:

2)**`<sound>`**
Not a valid HTML5 element.
වලංගු HTML5 අංගයක් නොවේ.

3)**`<music>`**
Does not exist in HTML5.
HTML5 හි නොපවතියි.

4)**`<player>`**
Not an official HTML5 tag for embedding audio or video.
Audio හෝ video කාවැද්දීම සඳහා නිල HTML5 ටැගයක් නොවේ.

5) **`<embed>`**
The `<embed>` tag is used for embedding external content (ex: Flash or PDFs) but lacks built-in media controls and is not recommended for modern audio playback.
`<embed>` ටැගය බාහිර අන්තර්ගත කාවැද්දීම සඳහා භාවිතා කරයි (උදා: Flash හෝ PDF) නමුත් ගොඩනඟන ලද built-in media controls නොමැති අතර නවීන audio playback සඳහා නිර්දේශ නොකරයි.