සියලු ම හිමිකම් ඇවිරිණි / All Rights Reserved]

Bandaranayake - ictfromabc Ravindu Bandaranayake - ictfromabc Ravindu Bandaranayake - ictfromabc Ravindu

Information and Communication Technology
තොරතුරු හා සන්නිවේදන තාක්ෂණය

2026 QUIZ NO 305 MARKING

2025/10/11
ictfromabc
Ravindu Bandaranayake

1. Who invented the Mark 1, the first automatic sequence controlled calculator?
   ප්‍රථම ස්වයංක්‍රිය අනුක්‍රමන පාලක ගණක යන්ත්‍රය වන මාර්ක් 1 සොයාගනු ලැබුවේ කවුරුන් විසින්ද?

   1) Blaise Pascal / බ්ලේස් පැස්කල්.
   2) Charles Babbage / චාල්ස් බැබේජ්.
   3) John V. Atanasoff / ජෝන් වී. ඇටනසොෆ්.
   4) Ada Augusta Lovelace / ඒඩා ඔගෙස්ටා ලව්ලේස්.
   5) <u>Howard Aiken / හෝවර්ඩ් අයිකන්.</u>

Answer: 5)

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

1) Blaise Pascal (බ්ලේස් පැස්කල්):
Blaise Pascal invented the Pascaline in the 1640s, which was an early mechanical calculator capable of performing basic arithmetic operations. It is not related to the development of the Mark 1.
Blaise Pascal විසින් 1640 දී Pascaline සොයා ගන්නා ලද අතර එය මූලික ගණිතමය මෙහෙයුම් සිදු කළ හැකි මුල් යාන්ත්‍රික ගණක යන්ත්‍රයක් විය. එය Mark 1 හි වර්ධනයට සම්බන්ධ නොවේ.

2) Charles Babbage (චාල්ස් බැබේජ්):
Charles Babbage is known as the "father of the computer" for his conceptual designs of the Analytical Engine and the Difference Engine in the 19th century. These were mechanical precursors to modern computers, but Babbage did not invent the Mark 1.
චාල්ස් බැබේජ් "පරිගණකයේ පියා" ලෙස හඳුන්වනු ලබන්නේ 19 වන සියවසේ Analytical Engine සහ Difference Engine පිළිබඳ ඔහුගේ සංකල්පීය සැලසුම් සඳහා ය. මේවා නවීන පරිගණක සඳහා යාන්ත්‍රික පුරෝගාමීන් වූ නමුත් බැබේජ් විසින් මාර්ක් 1 නිර්මාණය කළේ නැත.

3) John V. Atanasoff (ජෝන් වී. ඇටනසොෆ්):
John V. Atanasoff, along with his student Clifford Berry, developed the Atanasoff-Berry Computer (ABC) in the late 1930s and early 1940s. The ABC was the first electronic digital computer, but it was not the Mark 1.
John V. Atanasoff, ඔහුගේ ශිෂ්‍ය Clifford Berry සමඟ එක්ව 1930 අගභාගයේ සහ 1940 මුල් භාගයේදී Atanasoff Berry පරිගණකය ABC සංවර්ධනය කරන ලදී. ABC යනු පළමු ඉලෙක්ට්‍රොනික ඩිජිටල් පරිගණකය වූ නමුත් එය මාර්ක් 1 නොවේ.

4) Ada Augusta Lovelace (ඒඩා ඔගෙස්ටා ලව්ලේස්):
Ada Lovelace wrote her first program for the Analytical Engine, Charles Babbage's original mechanical general-purpose computer. She is therefore often considered the first computer programmer, although she did not invent the Mark 1.
ඇඩා ලව්ලේස්, චාල්ස් බැබේජ්ගේ මුල් යාන්ත්‍රික සාමාන්‍ය කාර්ය පරිගණකය වන විශ්ලේෂණ එන්ජිම සඳහා ඇයගේ පළමු ක්‍රමලේඛය ලියන ලදී. එම නිසා ඇය බොහෝ විට පළමු පරිගණක ක්‍රමලේඛකයා ලෙස සැලකේ, නමුත් ඇය මාර්ක් 1 නිර්මාණය කළේ නැත.

5) Howard Aiken (හෝවර්ඩ් අයිකන්)
The Harvard Mark I, also known as the IBM Automatic Sequence Controlled Calculator (ASCC), was conceived by Howard Aiken and built by IBM. It was one of the earliest general-purpose computers and was used during World War II.
Harvard Mark I , IBM Automatic Sequence Controlled Calculator (ASCC) ලෙසද හඳුන්වනු ලබන අතර, Howard Aiken විසින් සංකල්පනය කරන ලද අතර IBM විසින් ගොඩනගා ඇත. එය පැරණිතම පොදු කාර්ය පරිගණක වලින් එකක් වූ අතර දෙවන ලෝක යුද්ධ සමයේදී භාවිතා කරන ලදී.

So, the answer is 5. / ඒ අනුව, පිළිතුර වන්නේ 5 වේ.

2. What is the main advantage of magnetic Hard Disk Drives (HDDs) over Solid State Drives (SSDs)?
ඝන තත්වයේ ධාවක (SSDs) වලට වඩා චුම්බක දෘඪ තැටි ධාවක (HDDs) වල ඇති ප්‍රධාන වාසිය කුමක්ද?

1) Higher data transfer rates / ඉහළ දත්ත හුවමාරු අනුපාතය.
2) Lower power consumption / අඩු බලශක්ති පරිභෝජනය.
3) Lower heat generation / අඩු තාප උත්පාදනය.
4) Lower cost per gigabyte / ගිගාබයිට් එකකට අඩු පිරිවැය.
5) Compatibility with a wider range of devices / පුළුල් පරාසයක උපාංග සමඟ අනුකූල වීම.

Answer: 4)

Explanation / පැහැදිලි කිරීම:

The main advantage of magnetic Hard Disk Drives (HDDs) over Solid State Drives (SSDs) is their lower cost per gigabyte. HDDs use older, more established technology and can store large amounts of data at a relatively lower price compared to SSDs. Although SSDs are faster and more energy-efficient, HDDs are still widely used in situations where cost-effectiveness is a priority, especially for bulk data storage, making them a more affordable option per unit of storage. Therefore, the answer is option 4.
Solid State Drives (SSDs) වලට වඩා චුම්භක දෘඪ තැටි ධාවකයන්ගේ (HDDs) ප්‍රධාන වාසිය වන්නේ ගිගාබයිට් එකකට ඒවායේ අඩු පිරිවැයයි. HDDs පැරණි, වඩා ස්ථාපිත තාක්ෂණය භාවිතා කරන අතර SSD වලට සාපේක්ෂව අඩු මිලකට විශාල දත්ත ප්‍රමාණයක් ගබඩා කළ හැක. SSDs වේගවත් හා වඩා බලශක්ති-කාර්යක්ෂම වුවද, HDDs තවමත් පිරිවැය-ඵලදායිතාවය ප්‍රමුඛතාවයක් වන අවස්ථාවන්හිදී බහුලව භාවිතා වේ, විශේෂයෙන් තොග දත්ත ගබඩා කිරීම සඳහා, ඒවා ගබඩා ඒකකයකට වඩා දැරිය හැකි විකල්පයක් බවට පත් කරයි. එබැවින්, පිළිතුර විකල්ප 4) වේ.

3. Which of the following is the smallest in value?
පහත ඒවායින් කුඩාම අගය කුමක්ද?

1) 201_8 2) 128_{10} 3) 82_{16} 4) 10000011_2 5) 177_8

Answer: 5)

Explanation / පැහැදිලි කිරීම:

Let's first convert all the numbers given in the above question into decimals.
ඉහත ප්‍රශ්නයෙහි දැක්වෙන සියළු සංඛ්‍යා දශමය බවට පරිවර්ථනය කරමු.

201_8		
2	0	1
8^2	8^1	8^0
128	-	1
128 + 1		
129_{10}		

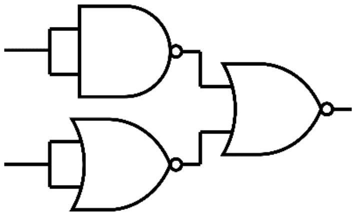
82_{16}	
8	2
16^1	16^0
128	2
128 + 2	
130_{10}	

10000011_2							
1	0	0	0	0	0	1	1
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	-	-	-	-	-	2	1
128 + 2 + 1							
131_{10}							

177_8		
1	7	7
8^2	8^1	8^0
64	56	7
64 + 56 + 7		
127_{10}		

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

4. Which gate is equivalent to the circuit shown below?
පහත දැක්වෙන පරිපථය තුල්‍ය වන්නේ කුමන ද්වාරයටද?



1) AND 2) NAND 3) OR 4) NOR 5) NOT

Answer: 1)

Let A be one input. Consider the other input as B and draw the truth table corresponding to circuit above.
එක් ආදානයක් A ලෙස සලකමු. අනෙක් අදානය B ලෙස සලකා ඉහත පරිපථයට ගැලපෙන සත්‍යතා වගුව අඳිමු.

A	B	$\overline{A.A}$	$\overline{B+B}$	$\overline{A.A}+\overline{B+B}$	$\overline{\overline{A.A}+\overline{B+B}}$
0	0	1	1	1	0
0	1	1	0	1	0
1	0	0	1	1	0
1	1	0	0	0	1

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

The truth table above shows that this is the output of the AND gate. So the answer is 1.
ඉහත සත්‍යතා වගුවට අනුව පැහැදිලි වන්නේ මෙය ද්වාරයේ ප්‍රතිදානය බවයි.එම නිසා පිළිතුර වන්නේ 1 වේ.

5. What is the result of trivializing the Boolean expression $\overline{AB} + C(AD + DC)$?
$\overline{AB} + C(AD + DC)$ යන බූලිය ප්‍රකාශනය සුළු කළ විට ලැබෙන ප්‍රතිඵලය කුමක්ද?

1) 1 2) CD 3) 0 4) $\overline{A} + \overline{B} + CD$ 5) $\overline{A} + C\overline{D}$

Answer: 4)

Let's get the answer as below.
පහත පරිදි පිළිතුර ලබාගනිමු.

$= \overline{AB} + C(AD + DC)$
$= \overline{A} + \overline{B} + C(AD + DC)$ De Morgan's Law / ද මුවර් ප්‍රමේය
$= \overline{A} + \overline{B} + CAD + CDC$ Distributive Law / විඝටන න්‍යාය
$= \overline{A} + \overline{B} + CAD + CD$ Idempotent Law / තදේවභාවි න්‍යාය
$= \overline{A} + \overline{B} + CD$ Redunclancy Law / සමතිරික්ත න්‍යාය

So, the answer here is 4).
එබැවින්, මෙහි පිළිතුර 4) වේ.

6. What is an Entity-Relationship (ER) diagram used for in database design?
දත්ත සමුදාය සැලසුම් කිරීමේදී භාවිතා කරන භූතාර්ථ-සම්බන්ධතා (ER) රූප සටහනක අරමුණ කුමක්ද?

1) To visualize the structure of a database දත්ත සමුදායක ව්‍යුහය දෘශ්‍යමාන කිරීමට
2) To generate code automatically කේතය ස්වයංක්‍රියව උත්පාදනය කිරීමට
3) To analyze network traffic ජාල ගමනාගමනය විශ්ලේෂණය කිරීමට
4) To send emails ඊමේල් යැවීමට
5) To design user interfaces පරිශීලක අතුරුමුහුණත් නිර්මාණය කිරීමට

Answer: 1)

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

Statement / ප්‍රකාශය 1)
An ER diagram is a visual representation of the entities (tables) in a database and their relationships. It helps in designing the database schema by depicting how data entities interact with each other, which is crucial for understanding the database structure and planning its design.
ER රූප සටහනක් යනු දත්ත සමුදායක ඇති භූතාර්ථ (වගු) සහ ඒවායේ සම්බන්ධතා වල දෘශ්‍ය නිරූපණයකි. දත්ත සමුදා ව්‍යුහය අවබෝධ කර ගැනීමට සහ එහි සැලසුම සැලසුම් කිරීමට ඉතා වැදගත් වන දත්ත ආයතන එකිනෙකා සමඟ අන්තර් ක්‍රියා කරන ආකාරය නිරූපණය කිරීමෙන් දත්ත සමුදා ක්‍රමය සැලසුම් කිරීමට එය උපකාරී වේ.

Statement / ප්‍රකාශය 2)
ER diagrams are not used to generate code automatically. They are primarily used for database design and visualization.
ER රූප සටහන් ස්වයංක්‍රියව කේතය ජනනය කිරීමට භාවිතා නොවේ. ඒවා මූලික වශයෙන් දත්ත සමුදා නිර්මාණය සහ දෘශ්‍යකරණය සඳහා භාවිතා වේ.

Statement / ප්‍රකාශය 3)
Analyzing network traffic is not related to ER diagrams. Network analysis involves tools and techniques specifically designed for monitoring and managing network performance.
ජාල ගමනාගමනය විශ්ලේෂණය කිරීම ER රූප සටහන් වලට සම්බන්ධ නොවේ. ජාල විශ්ලේෂණයට ජාල ක්‍රියාකාරිත්වය අධීක්ෂණය සහ කළමනාකරණය සඳහා විශේෂයෙන් නිර්මාණය කර ඇති මෙවලම් සහ ශිල්පීය ක්‍රම ඇතුළත් වේ.

Statement / ප්‍රකාශය 4)
Sending emails is not related to ER diagrams. Email operations are handled by email clients and servers, not by database design tools.
විද්‍යුත් තැපැල් යැවීම ER රූප සටහන් වලට සම්බන්ධ නොවේ. විද්‍යුත් තැපැල් මෙහෙයුම් හසුරුවන්නේ විද්‍යුත් තැපැල් සේවාදායකයින් සහ සේවාදායකයන් විසින් මිස දත්ත සමුදා නිර්මාණ මෙවලම් මගින් නොවේ.

Statement / ප්‍රකාශය 5)
Designing user interfaces is not the purpose of ER diagrams. User interface design involves different tools and practices focused on the visual and functional aspects of applications.
පරිශීලක අතුරුමුහුණත් සැලසුම් කිරීම ER රූප සටහන් වල අරමුණ නොවේ. පරිශීලක අතුරුමුහුණත් නිර්මාණය යෙදුම්වල දෘශ්‍ය සහ ක්‍රියාකාරී අංශ කෙරෙහි අවධානය යොමු කරන විවිධ මෙවලම් සහ භාවිතයන් ඇතුළත් වේ.

So, the correct answer is 1). / එමනිසා, නිවැරදි පිළිතුර 1) වේ.

7. What is an entity in the context of a database?
   දත්ත සමුදායක සන්දර්භය තුළ ඇති භූතාර්ථයක් යනු කුමක්ද?

   1) A single column in a table / වගුවක තනි තීරුවක්
   2) A specific SQL query / නිශ්චිත SQL විමසුමක්
   3) A distinct object or thing in the database / දත්ත සමුදායේ ඇති වෙනස් වස්තුවක් හෝ දෙයකි
   4) A type of data encryption / දත්ත සංකේතාංකන වර්ගයකි
   5) A database index / දත්ත සමුදා දර්ශකය

Answer: 3)

The correct answer is 3) A distinct object or thing in the database.
නිවැරදි පිළිතුර 3) දත්ත සමුදායේ ඇති වෙනස් වස්තුවක් හෝ දෙයක් ය.

In database terms, an entity refers to any real-world object or concept that has a distinct existence and is represented in the database. Examples include a person, product, or order, each of which can be uniquely identified and described.
දත්ත සමුදාය අනුව, භූතාර්ථයක් යනු දත්ත සමුදාය තුළ නිරූපණය වන පැහැදිලි පැවැත්මක් ඇති ඕනෑම තත්‍ය-ලෝක වස්තුවක් හෝ සංකල්පයකට යොමු කරයි. උදාහරණ ලෙස පුද්ගලයෙකු, නිෂ්පාදනයක් හෝ අනුපිළිවෙලක් ඇතුළත් වේ, ඒ සෑම එකක්ම අනන්‍ය ලෙස හඳුනාගෙන විස්තර කළ හැකිය.

Given options / ලබා දී ඇති විකල්ප:

1) **A single column in a table / 1 වගුවක තනි තීරුවකි**
   This is called an attribute, not an entity.
   මෙය භූතාර්ථයක් නොව උපලක්ෂණයක් ලෙස හැඳින්වේ.

2) **A specific SQL query / නිශ්චිත SQL විමසුමකි**
   SQL queries are used to retrieve or manipulate data, not entities.
   SQL විමසුම් භාවිතා කරනුයේ දත්ත ලබාගැනීමට හෝ හැසිරවීමට මිස භූතාර්ථ නොවේ.

3) **A distinct object or thing in the database / දත්ත සමුදායේ ඇති වෙනස් වස්තුවක් හෝ දෙයකි**
   This is the correct definition of an entity.
   මෙය භූතාර්ථයක නිවැරදි නිර්වචනයයි.

4) **A type of data encryption / දත්ත ගුප්තකේතාංකන වර්ගයකි**
   Data encryption is unrelated to entities.
   දත්ත ගුප්තකේතනය භූතාර්ථවලට සම්බන්ධ නැත.

5) **A database index / දත්ත සමුදා දර්ශකය**
   An index is used to improve the speed of data retrieval, but it is not an entity.
   දත්ත ලබා ගැනීමේ වේගය වැඩි දියුණු කිරීම සඳහා දර්ශකයක් භාවිතා කරයි, නමුත් එය භූතාර්ථයක් නොවේ.

8. Which keyword is used to create a function in Python?
   පයිතන් හි ශ්‍රිතයක් සෑදීමට භාවිතා කරන මූල පදය කුමක්ද?

   1) `func` 2) `def` 3) `function` 4) `create` 5) `define`

   Answer: 2)

   The correct answer for the keyword used to create a function in Python is 2) `def`.
   Python හි ශ්‍රිතයක් සෑදීමට භාවිතා කරන මූල පදය සඳහා නිවැරදි පිළිතුර වන්නේ 2) `def`.

   Explanation of the options / පිළිතුරු පැහැදිලි කිරීම:

1) **`func`**
   This is not a valid keyword in Python for creating a function. It might be seen in other programming languages or contexts, but not in Python.
   මෙය ශ්‍රිතයක් සෑදීම සඳහා Python හි වලංගු මූල පදයක් නොවේ. එය වෙනත් ක්‍රමලේඛන භාෂා හෝ සන්දර්භ (contexts) වල දැකිය හැක, නමුත් Python වල නොවේ.

2) **`def`**
   This is the correct answer. In Python, the def keyword is used to define a new function. It is followed by the function name, parentheses (which may include parameters), and a colon. The function body follows, indented under the def line. Here is an example:
   මෙය නිවැරදි පිළිතුරයි. Python හි, නව ශ්‍රිතයක් අර්ථ දැක්වීමට (def) මූල පදය භාවිතා කරයි. එය පසුව ශ්‍රිත නාමය, වරහන් (පරාමිතීන් ඇතුළත් විය හැක) සහ මහා බඩවැලේ. ක්‍රියාකාරී ශරීරය අනුගමනය කරයි, ඩෙෆ් රේඛාව යටතේ ඉන්ඩෙන්ට් කර ඇත. මෙන්න උදාහරණයක්:

```
def my_function():
    print("Hello, World!")
```

3) **`function`**
   This is not a keyword in Python for creating a function. It is a common term used to describe functions, but in Python, you use `def` to define them.
   මෙය ශ්‍රිතයක් සෑදීම සඳහා Python හි ඇති මූල පදයක් නොවේ. එය ශ්‍රිත විස්තර කිරීමට භාවිතා කරන පොදු යෙදුමකි, නමුත් Python හි, ඔබ ඒවා නිර්වචනය කිරීමට `def` භාවිතා කරයි.

4) **`create`**
This is not a keyword in Python for creating a function. Python does not use the create keyword for defining functions or any other constructs.
මෙය ශ්‍රිතයක් සෑදීම සඳහා Python හි ඇති මුල පදයක් නොවේ. Python විසින් ශ්‍රිතයන් නිර්වචනය කිරීම සඳහා හෝ වෙනත් නිර්මිතයන් සඳහා create keyword භාවිතා නොකරයි.

5) **`define`**
This is not a keyword in Python. While "define" is conceptually related to creating functions, in Python, the actual keyword used is def.
මෙය Python හි මුල පදයක් නොවේ. "define" යන්න සංකල්පමය වශයෙන් ශ්‍රිත සෑදීමට සම්බන්ධ වන අතර, Python හි, භාවිතා කරන සැබෑ මුල පදය def වේ.

9. Which of the following statement best describes an interpreted language?
අර්ථකථනය කරන ලද භාෂාවක් වඩාත් හොඳින් විස්තර කරන්නේ පහත සඳහන් කුමන ප්‍රකාශයද?

1) An interpreted language is a programming language that is directly executed by the operating system without the need for compilation.
අර්ථකථනය කරන ලද භාෂාවක් යනු සම්පාදනය කිරීමේ අවශ්‍යතාවයකින් තොරව මෙහෙයුම් පද්ධතිය විසින් සෘජුවම ක්‍රියාත්මක කරන ක්‍රමලේඛන භාෂාවකි.

2) An interpreted language is a programming language in which the code is translated into machine code before execution.
අර්ථකථනය කරන ලද භාෂාවක් යනු ක්‍රියාත්මක කිරීමට පෙර කේතය යන්ත්‍ර කේතයට පරිවර්තනය කරන ක්‍රමලේඛන භාෂාවකි.

3) <u>An interpreted language is a programming language that is executed line-by-line by an interpreter at runtime.</u>
අර්ථකථනය කරන ලද භාෂාවක් යනු ක්‍රමලේඛන භාෂාවක් වන අතර එය ධාවන වේලාවේදී පරිවර්තකයක් විසින් පේළියෙන් පේළිය ක්‍රියාත්මක කරනු ලැබේ.

4) An interpreted language is a programming language that must be compiled into bytecode before it can be executed.
අර්ථකථනය කරන ලද භාෂාවක් යනු ක්‍රමලේඛන භාෂාවක් වන අතර එය ක්‍රියාත්මක කිරීමට පෙර එය බයිට්කේතයට සම්පාදනය කළ යුතුය.

5) An interpreted language is a programming language used exclusively for scripting web pages.
අර්ථකථනය කරන ලද භාෂාවක් යනු වෙබ් පිටු පිටපත් කිරීම සඳහා පමණක් භාවිතා කරන ක්‍රමලේඛන භාෂාවකි.

Answer: 3)

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

Statement / ප්‍රකාශය 1)
This statement is incorrect because an interpreted language relies on an interpreter, not the operating system, for execution. Interpreted languages are processed by an interpreter at runtime, not directly by the OS.
පරිවර්ථනය කරන ලද භාෂාවක් ක්‍රියාත්මක කිරීම සඳහා මෙහෙයුම් පද්ධතිය මත නොව පරිවර්තකයෙකු මත රඳා පවතින බැවින් මෙම ප්‍රකාශය වැරදිය. පරිවර්ථනය කරන ලද භාෂා සකසනු ලබන්නේ ධාවන වේලාවේදී පරිවර්තකයෙකු විසිනි, සෘජුවම OS මගින් නොවේ.

Statement / ප්‍රකාශය 2)
This describes a compiled language, not an interpreted one. In compiled languages, the code is converted to machine code before execution. Interpreted languages do not compile the code into machine code beforehand; instead, they are executed by an interpreter.
මෙය විස්තර කරන්නේ සම්පාදනය කරන ලද භාෂාවක් මිස අර්ථකථනය කළ භාෂාවක් නොවේ. සම්පාදනය කරන ලද භාෂාවලින්, කේතය ක්‍රියාත්මක කිරීමට පෙර යන්ත්‍ර කේතයට පරිවර්තනය වේ. පරිවර්ථනය කරන ලද භාෂාවන් පෙර කේතය යාන්ත්‍රික කේතයට සම්පාදනය නොකරයි. ඒ වෙනුවට, ඒවා පරිවර්තකයෙකු විසින් ක්‍රියාත්මක කරනු ලැබේ.

Statement / ප්‍රකාශය 3)
This is the correct definition of an interpreted language. In such languages, the code is executed line-by-line by an interpreter during runtime, rather than being compiled into machine code beforehand.
අර්ථකථනය කරන ලද භාෂාවක නිවැරදි අර්ථ දැක්වීම මෙයයි. එවැනි භාෂාවලදී, කේතය පෙර යන්ත්‍ර කේතයට සම්පාදනය කරනවාට වඩා ධාවන කාලය තුළ පරිවර්තකයෙකු විසින් පේළියෙන් පේළිය ක්‍රියාත්මක කරයි.

Statement / ප්‍රකාශය 4)
This statement refers more to languages like Java, where the source code is first compiled into an intermediate form (bytecode) and then interpreted by a virtual machine. This isn't the typical process for purely interpreted languages like Python or Ruby.
මෙම ප්‍රකාශය ජාවා වැනි භාෂාවලට වැඩි වශයෙන් යොමු දක්වයි, එහිදී ප්‍රභව කේතය මුලින්ම අතරමැදි ආකෘතියකට (බයිටෙකෝඩ්) සම්පාදනය කර පසුව අථත්‍ය යන්ත්‍රයක් මගින් අර්ථකථනය කරයි. මෙය Python හෝ Ruby වැනි සම්පූර්ණයෙන්ම අර්ථකථනය කරන ලද භාෂා සඳහා සාමාන්‍ය ක්‍රියාවලිය නොවේ.

Statement / ප්‍රකාශය 5)
This is incorrect because interpreted languages are used in many areas, not just for scripting web pages. While some interpreted languages, like JavaScript, are commonly used in web development, the term "interpreted language" is much broader.
මෙය වැරදියි මන්ද පරිවර්ථනය කරන ලද භාෂා බොහෝ ප්‍රදේශවල භාවිතා වන අතර, වෙබ් පිටු පිටපත් කිරීම සඳහා පමණක් නොවේ. JavaScript වැනි සමහර පරිවර්ථනය කරන ලද භාෂා වෙබ් සංවර්ධනයේදී බහුලව භාවිතා වන අතර, "interpreted language" යන යෙදුම වඩා පුළුල් වේ.

10. What is the output of the below python code?
පහත පයිතන් කේතයේ ප්‍රතිදානය කුමක්ද?

```
a=[10,20]
b=[30,40]
c=(a+b)
print(c)
```

1) `[40, 60]`
2) `([10, 20]+[30, 40])`
3) `[10, 20, 30, 40]`
4) `[[10, 20], [30, 40]]`
5) `(10, 20, 30, 40)`

Answer: 3)

Let's consider the explanation of each code.
එක් එක් කේතයේ පැහැදිලි කිරීම සලකා බලමු.

`a = [10, 20]`
A list a is created with two elements 10 and 20.
10 සහ 20 යන මූලාංග දෙකකින් a ලැයිස්තුවක් සාදනු ලැබේ.

`b = [30, 40]`
Another list b is created with two elements 30 and 40.
30 සහ 40 යන තවත් මූලාංග දෙකකින් b ලැයිස්තුවක් සාදනු ලැබේ

`c = (a + b)`
Here, the two lists a and b are concatenated using the + operator.
මෙන්න, a සහ b ලැයිස්තු දෙක + මෙහෙයවනය භාවිතයෙන් සංයුක්ත වේ.

In Python, using + with lists joins the two lists together. The elements of both lists are combined into a single list.
පයිතන් වලදී, + සමග list භාවිතා කිරීමෙන් ලැයිස්තු දෙක එකට එකතු කරයි. ලැයිස්තු දෙකෙහිම මූලාංග තනි ලැයිස්තුවකට ඒකාබද්ධ වේ.

So, `a + b` results in `[10, 20, 30, 40]`.
එබැවින්, `a + b` ප්‍රතිඵලය `[10, 20, 30, 40]` වේ.

The parentheses `()` around `(a + b)` do not create a tuple, they are just grouping the expression. So, the result is still a list.
`(a + b)` වටා ඇති වරහන් `()` tuple එකක් නිර්මාණය නොකරයි, එය ප්‍රකාශනය සමූහගත කරයි. එබැවින්, ප්‍රතිඵලය තවමත් ලැයිස්තුවකි.

Hence, c is assigned the value `[10, 20, 30, 40]`.
එබැවින්, `[10, 20, 30, 40]` අගය පවරනු ලැබේ.

`print(c)`
The `print` function outputs the value of `c`, which is `[10, 20, 30, 40]`.
`print` ශ්‍රිතය `c` හි අගය ප්‍රතිදානය කරයි, එය `[10, 20, 30, 40]`වේ.

The correct output of the code is `[10, 20, 30, 40]`.
කේතයේ නිවැරදි ප්‍රතිදානය `[10, 20, 30, 40]` වේ.