සියලු ම හිමිකම් ඇවිරිණි / *All Rights Reserved*]

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

Information and Communication Technology

තොරතුරු හා සන්නිවේදන තාක්ෂණය

2026 QUIZ NO 269 MARKING

2025/09/05
ictfromabc
Ravindu Bandaranayake

1. Which of the following technologies was predominantly utilized in first-generation computers, which were operational from 1946 to 1959?
1946 සිට 1959 දක්වා ක්‍රියාත්මක වූ පළමු පරම්පරාවේ පරිගණකවල ප්‍රධාන වශයෙන් භාවිත කළේ පහත සඳහන් කුමන තාක්ෂණයද?

   1) Transistors / ට්‍රාන්සිස්ටර
   2) Microprocessors / ක්ෂුද්‍ර සකසනය
   3) Integrated Circuits / අනුකලිත පරිපථ
   4) Vacuum Tubes / රික්තක නළ
   5) Cloud Computing / වලාකුලු පරිගණනය

   Answer: 4)

   Explanation of each option / එක් එක් විකල්පය පිළිබඳ පැහැදිලි කිරීම:

   Option / විකල්ප 1)
   Transistors were introduced in the second generation of computers (1959-1965), replacing vacuum tubes. Transistors were much smaller, more efficient, and more reliable, which significantly reduced the size of computers.
   රික්තක නල ප්‍රතිස්ථාපනය කරමින් ට්‍රාන්සිස්ටර පරිගණකවල දෙවන පරම්පරාවේ (1959-1965) හඳුන්වා දෙන ලදී. ට්‍රාන්සිස්ටර ඉතා කුඩා, වඩා කාර්යක්ෂම සහ විශ්වාසදායක වූ අතර එමඟින් පරිගණකවල ප්‍රමාණය සැලකිය යුතු ලෙස අඩු විය.

   Option / විකල්ප 2)
   Microprocessors were introduced in the fourth generation (from 1971 onwards) and are integrated circuits that contain all the functions of a CPU on a single chip.
   ක්ෂුද්‍ර සකසන හතරවන පරම්පරාවේ (1971 සිට) හඳුන්වා දෙන ලද අතර ඒවා තනි චිපයක් මත CPU හි සියලුම ක්‍රියාකාරකම් අඩංගු අනුකලිත පරිපථ වේ.

   Option / විකල්ප 3)
   Integrated Circuits were a key technology in the third generation (1965-1971), allowing for even smaller and more efficient computers compared to those using transistors.
   ට්‍රාන්සිස්ටර භාවිතා කරන පරිගණක හා සසඳන විට ඊටත් වඩා කුඩා හා කාර්යක්ෂම පරිගණක සඳහා ඉඩ සලසන තුන්වන පරම්පරාවේ (1965-1971) ප්‍රධාන තාක්ෂණයක් වූයේ අනුකලිත පරිපථය.

   Option / විකල්ප 4)
   First-generation computers (1946-1959) were characterized by the use of vacuum tubes as the main technology for their circuitry. These tubes controlled electrical signals and allowed for the processing of data. Although effective at the time, vacuum tubes were large, consumed a lot of power, and generated a significant amount of heat, making these computers bulky and prone to overheating.
   පළමු පරම්පරාවේ පරිගණක (1946-1959) ඒවායේ පරිපථ සඳහා ප්‍රධාන තාක්ෂණය ලෙස රික්තක නල භාවිතය මගින් සංලක්ෂිත විය. මෙම නල මගින් විද්‍යුත් සංඥා පාලනය කරන අතර දත්ත සැකසීමට ඉඩ ලබා දෙන ලදී. එකල ඵලදායී වුවද, රික්ත නල විශාල වූ අතර, විශාල බලයක් පරිභෝජනය කරන අතර, සැලකිය යුතු තාප ප්‍රමාණයක් ජනනය කරන ලද අතර, මෙම පරිගණක විශාල හා අධික ලෙස රත් වීමට ඉඩ ඇත.

Option / විකල්ප 5)
Cloud Computing is a modern technology that enables the storage, management, and processing of data over the internet, and is not associated with the hardware-based generations of early computers. It emerged much later in the 21st century.
වලාකුලු පරිගණක යනු අන්තර්ජාලය හරහා දත්ත ගබඩා කිරීම, කළමනාකරණය කිරීම සහ සැකසීම සක්‍රිය කරන නවීන තාක්ෂණයක් වන අතර මුල් පරිගණකවල දෘඩාංග පදනම් වූ පරම්පරාවන් සමඟ සම්බන්ධ නොවේ. එය බොහෝ කලකට පසුව 21 වන සියවසේදී මතු විය.

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

2. Which of the following best describes "refresh rate" of a monitor?
මොනිටරයක "නැවුම් කිරීමේ අනුපාතය" වඩාත් හොඳින් විස්තර කරන්නේ පහත සඳහන් ඒවායින් කුමක් ද?

1) Number of pixels on the screen / තිරයේ ඇති පික්සල ගණන
2) Number of times the screen updates per second / තිරය තත්පරයකට යාවත්කාලීන වන වාර ගණන
3) Time it takes for a pixel to change color / පික්සලයක වර්ණය වෙනස් වීමට ගතවන කාලය
4) Maximum brightness of the display / සංදර්ශකයේ උපරිම දීප්තිය
5) Power consumption of the monitor / මොනිටරයේ බල පරිභෝජනය

Answer: 2)

Explanation / පැහැදිලි කිරීම:

The refresh rate of a monitor refers to how many times per second the display redraws the image on the screen. It is measured in Hertz (Hz)
මොනිටරයක නැවුම් කිරීමේ අනුපාතය යනු තිරය මත රූපය තත්පරයට කී වතාවක් නැවත අඳිනවාද යන්නයි. එය හර්ට්ස් (Hz) වලින් මනිනු ලැබේ.

- Number of pixels on the screen – This describes the resolution of the monitor, such as 1920×1080, and determines the clarity and detail of the image, not the refresh rate.
  තිරයේ ඇති පික්සල ගණන - මෙය (1920-1080) වැනි මොනිටරයේ විභේදනය විස්තර කරන අතර, නැවුම් කිරීමේ අනුපාතය නොව, රූපයේ පැහැදිලි බව සහ විස්තර තීරණය කරයි.

- Time it takes for a pixel to change color – This is called response time, which affects motion blur but is different from refresh rate.
  පික්සලයක වර්ණය වෙනස් වීමට ගතවන කාලය - මෙය ප්‍රතිචාර කාලය ලෙස හැඳින්වේ, එය චලන බොඳවීමට බලපාන නමුත් නැවුම් කිරීමේ අනුපාතයට වඩා වෙනස් වේ.

- Maximum brightness of the display – This measures how bright the monitor can get, affecting visibility in bright environments, but unrelated to refresh rate.
  සංදර්ශකයේ උපරිම දීප්තිය - මෙය මොනිටරයට කෙතරම් දීප්තිමත් විය හැකිද යන්න මනිනු ලබන අතර, දීප්තිමත් පරිසරවල දෘශ්‍යතාවයට බලපායි, නමුත් නැවුම් කිරීමේ අනුපාතයට සම්බන්ධ නොවේ.

- Power consumption of the monitor – Refers to the energy used by the monitor during operation; it does not affect how often the screen updates.
  මොනිටරයේ බල පරිභෝජනය - ක්‍රියාත්මක වන විට මොනිටරය භාවිතා කරන ශක්තියට යොමු වේ එය තිරය කොපමණ වාරයක් යාවත්කාලීන වේද යන්නට බලපාන්නේ නැත.

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

3. What is the result of the bitwise AND operation of the binary numbers 11110000 and 10101010?
ද්විමය අංක 11110000 සහ 10101010 හි bitwise AND යටතේ ක්‍රියාත්මක කලවිට ප්‍රතිඵලය කුමක්ද?

1) 10101010_2 2) 11100000_2 3) $\underline{10100000_2}$ 4) 11111111_2 5) 10000000_2

Answer: 3)

Explanation / පැහැදිලි කිරීම: -Bitwise AND / බිටු අනුසාරිත AND

- The bitwise AND operator is a fundamental operation in computer science used to compare the individual bits of two binary numbers.
බිටු අනුසාරිත AND මෙහෙයවනය යනු ද්විමය සංඛ්‍යා දෙකක තනි බිටු සංසන්දනය කිරීමට භාවිතා කරන පරිගණක විද්‍යාවේ මූලික මෙහෙයුමකි.

- It performs a logical AND operation between each pair of corresponding bits in the two numbers. The result is 1 if both bits are 1, otherwise the result is 0.
එය සංඛ්‍යා දෙකෙහි එක් එක් අනුරූප බිටු යුගල අතර තාර්කික AND මෙහෙයුමක් සිදු කරයි. බිටු දෙකම 1 නම් ප්‍රතිඵලය 1 වේ, එසේ නොමැති නම් ප්‍රතිඵලය 0 වේ.

Let's perform bitwise AND on 11110000_2 and 10101010_2.
බිටු අනුසාරිත AND මෙහෙයුම 11110000_2 සහ 10101010_2 මත කරමු.

1	1	1	1	0	0	0	0
1	0	1	0	1	0	1	0
1	0	1	0	0	0	0	0

Therefore, the correct answer is option 3) 10100000_2.
එබැවින්, නිවැරදි පිළිතුර විකල්පය 3) 10100000_2 වේ.

4. Which of the following statements about ASCII is correct?
ASCII පිළිබඳ පහත සඳහන් ප්‍රකාශවලින් නිවැරදි වන්නේ කුමක්ද?

1) ASCII uses 16 bits to represent each character.
ASCII හි සෑම අක්ෂරයක්ම නිරූපණය කිරීමට බිටු 16 ක් භාවිතා කරයි.
2) Standard ASCII can represent a maximum of 128 characters.
සම්මත ASCII මඟින් උපරිම වශයෙන් අක්ෂර 128ක් නිරූපණය කළ හැක.
3) Extended ASCII is identical to Unicode. / Extended ASCII යුනිකෝඩ් වලට සමාන වේ.
4) ASCII includes built-in support for all world languages.
ASCII හි සියලුම ලෝක භාෂා සඳහා built-in සහාය ඇතුළත් වේ.
5) ASCII was developed in the 21st century. / ASCII 21 වන සියවසේදී සංවර්ධනය කරන ලදී.

Answer: 2)

Explanation of each option / එක් එක් විකල්පය පැහැදිලි කිරීම:

Option / විකල්ප 1)
Incorrect: ASCII is based on 7 bits (not 16 bits), allowing 128 unique codes (0–127).
වැරදියි: ASCII බිටු 7ක් (බිටු 16ක් නොවේ) මත පදනම් වී ඇති අතර, අනන්‍ය කේත 128ක් (0 -127) ලබා දෙයි.

Option / විකල්ප 2)
Correct: Standard ASCII indeed represents 128 characters, including letters, digits, punctuation, and control characters.
නිවැරදි: සම්මත ASCII ඇත්ත වශයෙන්ම අකුරු, ඉලක්කම්, විරාම ලකුණු සහ පාලන අක්ෂර ඇතුළුව අක්ෂර 128 ක් නියෝජනය කරයි.

Option / විකල්ප 3)
Incorrect: Extended ASCII uses 8 bits (256 codes), but it is not the same as Unicode. Unicode is far larger and supports global languages.
වැරදියි: Extended ASCII බිටු 8ක් (කේත 256ක්) භාවිතා කරයි, නමුත් එය යුනිකෝඩ් වලට සමාන නොවේ. යුනිකෝඩ් ඉතා විශාල වන අතර ගෝලීය භාෂාවන්ට සහය දක්වයි.

Option / විකල්ප 4)
Incorrect: ASCII does not support all world languages; it mainly covers English.
වැරදියි: ASCII සියලුම ලෝක භාෂා සඳහා සහය නොදක්වයි එය ප්‍රධාන වශයෙන් ඉංග්‍රීසි ආවරණය කරයි.

Option / විකල්ප 5)
Incorrect: ASCII was developed in the 1960s, not in the 21st century.
වැරදියි: ASCII සංවර්ධනය කරන ලද්දේ 21 වන සියවසේ නොව 1960 ගණන්වල ය.

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

5. Which of the following logic gate combinations can be used to construct a 2-input XOR gate?
2-ආදාන XOR ද්වාරයක් තැනීම සඳහා පහත සඳහන් තාර්කික ද්වාර සංයෝජනවලින් කුමන එකක් භාවිතා කළ හැකිද?

1) Two AND gates, one OR gate, and one NOT gate
2) One AND gate, two OR gates, and two NOT gates
3) Two NAND gates, one AND gate, and one OR gate
4) Two AND gates, two NOT gates, and one OR gate
5) Three NOR gates only

Answer: 4)

Explanation / පැහැදිලි කිරීම:

The Exclusive-OR (XOR) gate produces a high output (1) only when the two inputs are different. Its Boolean expression is:
The Exclusive-OR (XOR) ද්වාරය ඉහළ ප්‍රතිදානයක් 1 නිපදවන්නේ ආදාන දෙක වෙනස් වූ විට පමණි" එහි බූලියන් ප්‍රකාශනය වන්නේ:
$A \oplus B = \overline{A}.B + A.\overline{B}$

This means the XOR function can be implemented by combining two AND gates, two NOT gates, and one OR gate. Specifically, one AND gate processes $A.\overline{B}$, the other processes$\overline{A}.B$, and the OR gate combines these two results.
මෙහි තේරුම XOR ශ්‍රිතය AND ද්වාර දෙකක්, NOT ද්වාර දෙකක් සහ OR ද්වාරයක් ඒකාබද්ධ කිරීමෙන් ක්‍රියාත්මක කළ හැකි බවයි. විශේෂයෙන්, එක් AND ද්වාර ක්‍රියාවලි $\overline{A}.B$ අනෙක් ක්‍රියාවලි $\overline{A}.B$ සහ OR ද්වාරය මෙම ප්‍රතිඵල දෙක ඒකාබද්ධ කරයි.

Therefore, option 4 correctly represents the construction of an XOR gate.
එබැවින්, විකල්ප 4 XOR ගේට්ටුවක් ඉදිකිරීම නිවැරදිව නිරූපණය කරයි.

6. Which of the following statements about Python data structures is not true?
පයිතන් දත්ත ව්‍යුහයන් පිළිබඳ පහත සඳහන් කුමන ප්‍රකාශය සත්‍ය නොවේද?

   1) A list in Python can contain elements of different data types, including other lists.
පයිතන් හි ලැයිස්තුවක වෙනත් ලැයිස්තු ඇතුළුව විවිධ දත්ත වර්ගවල මූලාංග අඩංගු විය හැක.

   2) A tuple is immutable, meaning its contents cannot be changed after creation.
ටපල් එකක් වෙනස් කළ නොහැකිය, එනම් නිර්මාණය කිරීමෙන් පසු එහි අන්තර්ගතය වෙනස් කළ නොහැක.

   3) A set in Python automatically removes duplicate elements when created.
පයිතන් හි set එකක් සාදන විට අනුපිටපත් මූලාංග ස්වයංක්‍රීයව ඉවත් කරයි.

   4) A dictionary in Python is an ordered collection of key-value pairs as of Python 3.7.
පයිතන් හි dictionary එකක් යනු පයිතන් 3.7ට අනුව පිළිවෙලට සාදන ලද යතුරු-අගය යුගල එකතුවකි.

   5) None of the Above
ඉහත කිසිවක් නොවේ

Answer: 5)

Explanation of each statement / **එක් එක් ප්‍රකාශය පැහැදිලි කිරීම:**

Statement / ප්‍රකාශය 1)
This statement is true. Python lists are heterogeneous, meaning they can contain elements of different data types, including other lists.
මෙම ප්‍රකාශය නිවැරදියි. Python **ලැයිස්තු විෂමජාතීය, එනම් ඒවාට වෙනත් ලැයිස්තු ඇතුළුව විවිධ දත්ත වර්ගවල මූලාංග අඩංගු විය හැක.**

Statement / ප්‍රකාශය 2)
This statement is true. Tuples are immutable, and once created, their elements cannot be modified.
මෙම ප්‍රකාශය නිවැරදියි. ටපල් වෙනස් කළ නොහැකි අතර, නිර්මාණය කළ පසු, ඒවායේ මූලාංග වෙනස් කළ නොහැක.

Statement / ප්‍රකාශය 3)
This statement is true. Sets in Python only store unique elements, and duplicates are automatically removed.
මෙම ප්‍රකාශය නිවැරදියි. Python **හි ඇති** sets **අද්විතීය මූලාංග පමණක් ගබඩා කරන අතර අනුපිටපත් ස්වයංක්‍රීයව ඉවත් කෙරේ.**

Statement / ප්‍රකාශය 4)
This statement is true. Starting from Python 3.7, dictionaries maintain insertion order.
මෙම ප්‍රකාශය නිවැරදියි. Python 3.7 **සිට,** dictionary **ඇතුළත් කිරීමේ අනුපිළිවෙල පවත්වාගෙන යයි**

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

7. Which of the following statements best describes the difference between low-level and high-level programming languages?
පහළ මට්ටමේ සහ ඉහළ මට්ටමේ ක්‍රමලේඛන භාෂා අතර වෙනස වඩාත් හොඳින් විස්තර කරන්නේ පහත සඳහන් කුමන ප්‍රකාශයෙන්ද?

1) Low-level languages are platform-independent, while high-level languages are platform-dependent.
පහළ මට්ටමේ භාෂා වේදිකා-ස්වාධීන වන අතර, ඉහළ මට්ටමේ භාෂා වේදිකා-රඳා පවතී.

2) Low-level languages are closer to machine hardware, while high-level languages are closer to human language.
පහළ මට්ටමේ භාෂා යන්ත්‍ර දෘඩාංග වලට සමීප වන අතර ඉහළ මට්ටමේ භාෂා මිනිස්භාෂාවට සමීප වේ.

3) Low-level languages provide automatic memory management, while high-level languages require manual memory handling.
පහළ මට්ටමේ භාෂා ස්වයංක්‍රිය මතක කළමනාකරණය සපයන අතර ඉහළ මට්ටමේ භාෂාවලට අතින් මතක හැසිරවීම අවශ්‍ය වේ.

4) Low-level languages are always easier to debug, while high-level languages are always harder to maintain.
පහළ මට්ටමේ භාෂා සෑම විටම නිදොස් කිරීමට පහසු වන අතර, ඉහළ මට්ටමේ භාෂා නඩත්තු කිරීමට සැමවිටම අපහසු වේ.

5) Low-level languages are designed only for web applications, while high-level languages are designed for embedded systems.
පහළ මට්ටමේ භාෂා වෙබ් යෙදුම් සඳහා පමණක් නිර්මාණය කර ඇති අතර, ඉහළ මට්ටමේ භාෂා එම්බෙඩඩ් පද්ධති සඳහා නිර්මාණය කර ඇත.

Answer: 2)

Explanation of each option / **එක් එක් විකල්පය පැහැදිලි කිරීම:**

Option / විකල්ප 1)
Incorrect. Low-level languages (like Assembly) are highly dependent on the hardware architecture, while many high-level languages (like Python or Java) are designed to be platform-independent.
වැරදියි. පහළ මට්ටමේ භාෂා (ඇසෙම්බ්ලි වැනි) දෘඩාංග නිර්මාණ ශිල්පය මත බෙහෙවින් රඳා පවතින අතර, බොහෝ ඉහළ මට්ටමේ භාෂා (පයිතන් හෝ ජාවා වැනි) වේදිකා-ස්වාධීන වීමට නිර්මාණය කර ඇත.

Option / විකල්ප 2)
Correct. Low-level languages provide direct access to hardware and memory, making them efficient but harder to read. High-level languages use syntax and structures that are easier for humans to understand and write.
නිවැරදියි. පහළ මට්ටමේ භාෂා දෘඩාංග සහ මතකයට සෘජු ප්‍රවේශයක් සපයන අතර, ඒවා කාර්යක්ෂම කරයි නමුත් කියවීමට අපහසු වේ. ඉහළ මට්ටමේ භාෂා මිනිසුන්ට තේරුම් ගැනීමට සහ ලිවීමට පහසු වාක්‍ය ඛණ්ඩ සහ ව්‍යුහයන් භාවිතා කරයි.

Option / විකල්ප 3)
Incorrect. Actually, low-level languages require manual memory management, whereas many high-level languages include automatic memory management (like garbage collection).
වැරදියි. ඇත්ත වශයෙන්ම, පහළ මට්ටමේ භාෂාවලට අතින් මතක කළමනාකරණය අවශ්‍ය වන අතර, බොහෝ ඉහළ මට්ටමේ භාෂාවන්ට ස්වයංක්‍රිය මතක කළමනාකරණය (කුණු එකතු කිරීම වැනි) ඇතුළත් වේ.

Option / විකල්ප 4)
Incorrect. Low-level code is generally harder to debug and maintain because it is less readable. High-level code is designed to be more readable and maintainable.
වැරදියි. පහළ මට්ටමේ කේතයක් සාමාන්‍යයෙන් දෝෂහරණය කිරීමට සහ නඩත්තු කිරීමට අපහසු වන්නේ එය අඩුවෙන් කියවිය හැකි බැවිනි. ඉහළ මට්ටමේ කේතයක් නිර්මාණය කර ඇත්තේ වඩාත් කියවිය හැකි සහ නඩත්තු කළ හැකි ආකාරයටය.

Option / විකල්ප 5)
Incorrect. Low-level languages are often used in embedded systems or performance-critical applications, whereas high-level languages are commonly used for applications like web development.
වැරදියි. පහළ මට්ටමේ භාෂා බොහෝ විට embedded පද්ධති හෝ කාර්ය සාධන-තීරණාත්මක යෙදුම්වල භාවිතා වන අතර, ඉහළ මට්ටමේ භාෂා බහුලව භාවිතා වන්නේ වෙබ් සංවර්ධනය වැනි යෙදුම් සඳහා ය.

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

8. What will be the output of the following Python code?
පහත පයිතන් කේතයේ ප්‍රතිදානය කුමක්ද?

```
a={1:"A",2:"B",3:"C"}
b=a.copy()
b[2]="D"
print(b)
```

1) {1: 'A', 2: 'B', 3: 'C'}
2) {1: 'A', 2: 'D', 3: 'C'}
3) {1: 'A', 2: 'B', 3: 'D'}
4) "None" is printed
5) Error, copy() method doesn't exist for dictionaries

Answer: 2)

Let's review the python code.
පහත පයිතන් කේතය විශ්ලේෂණය කරමු

Initial Dictionary / ආරම්භක Dictionary එක
`a: {1: "A", 2: "B", 3: "C"}`

Using `a.copy()`:
This creates a copy of dictionary `a`, resulting in a new dictionary `b` with the same key-value pairs.
මෙය a dictionary එකෙහි පිටපතක් නිර්මාණය කරයි, එහි ප්‍රතිඵලයක් ලෙස එම key-value යුගල සහිත නව ශබ්ද කෝෂයක් b වේ.

Modifying `b[2]`:
Changes the value associated with key 2 in `b` to "D". This change affects b only, not the original dictionary a.
b හි දෙවන key එක හා සම්බන්ධ අගය "D" ලෙස වෙනස් කරයි. මෙම වෙනස බලපාන්නේ b ට මිස a මුල් dictionary එකකට නොවේ.

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

9. What will be the output of the following Python code?
පහත පයිතන් කේතයේ ප්‍රතිදානය කුමක්ද?

```
x = 10
y = 20
def calculate(a, b):
    global x
    x = a + b
    y = x - b
    return y
z = calculate(x, y)
print(x, y, z)
```

1) `30 10 10` 2) `10 20 10` 3) `30 20 10` 4) `30 10 20`

5) `Error: local variable 'y' referenced before assignment`

Answer: 3)

Explanation / පැහැදිලි කිරීම:

Initially, two global variables x and y are defined with values 10 and 20, respectively. The function `calculate(a, b)` takes two parameters and modifies the global variable x by assigning it the sum of a and b. Inside the function, a local variable y is defined as x - b, which depicts the global y within the function's scope. The function then returns this local y. When the function is called with `calculate(x, y)` (i.e., a = 10 and b = 20), the global x is updated to 30 (10 + 20), while the local y is calculated as 10 (30 - 20). The returned value 10 is assigned to z. Finally, the print statement outputs the updated global x (30), the unchanged global y (20), and the returned value z (10), resulting in the output `30 20 10`.
මුලදී, ගෝලීය විචල්‍යයන් දෙකක් x සහ y පිළිවෙළින් 10 සහ 20 අගයන් සමඟ අර්ථ දක්වා ඇත. ශ්‍රිතය `calculate(a, b)` පරාමිති දෙකක් ගන්නා අතර ගෝලීය විචල්‍ය x ට a සහ b හි එකතුව ලබා දීමෙන් එය වෙනස් කරයි. ශ්‍රිතය තුළ, y දේශීය විචල්‍යයක් x - b ලෙස අර්ථ දක්වා ඇත, එය ශ්‍රිතයේ විෂය පථය තුළ ගෝලීය y නිරූපණය කරයි. ශ්‍රිතය පසුව මෙම දේශීය y නැවත ලබා දෙයි. `calculate(x, y)` ශ්‍රිතය (i.e., a = 10 and b = 20) සමඟින් හැඳින්වූ විට, ගෝලීය y, 30 (10 + 20) ලෙස යාවත්කාලීන වන අතර, දේශීය y 10 (30 - 20) ලෙස ගණනය කෙරේ. ආපසු ලබා දුන් අගය 10, z වෙත පවරා ඇත. අවසාන වශයෙන්, print ප්‍රකාශය යාවත්කාලීන කරන ලද ගෝලීය x (30), නොවෙනස්ව ඇති ගෝලීය y (20) සහ ආපසු ලැබෙන අගය z (10) ප්‍රතිදානය කරයි, ප්‍රතිඵලයක් ලෙස `30 20 10` ප්‍රතිදානය ලැබේ.

Therefore, the answer is 3).
එබැවින්, පිළිතුර 3) වේ.

10. Which of the following is NOT a valid way to define a function in Python?
පයිතන් හි ශ්‍රිතයක් නිර්වචනය කිරීමට වලංගු නොවන ක්‍රමය පහත සඳහන් මොනවාද?

1) `def my_function():`
2) `def my_function(x, y):`
3) `my_function = lambda x, y: x + y`
4) `def my_function(x = 10):`
5) `function my_function():`

Answer: 5)

Explanation of each option / එක් එක් විකල්පය පැහැදිලි කිරීම:

Option / විකල්ප 1)
This is valid. This defines a simple function with no parameters in Python.
මෙය වලංගු වේ. මෙය පයිතන් හි පරාමිති නොමැති සරල ශ්‍රිතයක් නිර්වචනය කරයි.

Option / විකල්ප 2)
This is valid. This defines a function with two parameters.
මෙය වලංගු වේ. මෙය පරාමිති දෙකක් සහිත ශ්‍රිතයක් නිර්වචනය කරයි.

Option / විකල්ප 3)
This is valid. This creates a `lambda` function (anonymous function) and assigns it to the name `my_function`.
මෙය වලංගු වේ. මෙය `lambda` ශ්‍රිතයක් (නිර්නාමික ශ්‍රිතයක්) සාදා එය `my_function` යන නාමයට පවරයි.

Option / විකල්ප 4)
This is valid. This defines a function with a default parameter value for `x`.
මෙය වලංගු වේ. මෙය `x` සඳහා පෙරනිමි පරාමිති අගයක් සහිත ශ්‍රිතයක් නිර්වචනය කරයි.

Option / විකල්ප 5)
This is invalid. Python does not use the keyword `function` to define functions. Instead, it uses `def`.
මෙය වලංගු නොවේ. පයිතන් විසින් ශ්‍රිත නිර්වචනය කිරීමට `function` ශ්‍රිතය භාවිතා නොකරයි. ඒ වෙනුවට, එය `def` භාවිතා කරයි.

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