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 228 MARKING

2025/07/26
ictfromabc
Ravindu Bandaranayake

1. In a course registration form, the manager needs to check whether all the data fields are filled. Which of the following data validation methods is suitable to fulfill that need?
**පාඨමාලා ලියාපදිංචි කිරීමේ පෝරමයක, කළමනාකරුට සියලු දත්ත ක්ෂේත්‍ර පුරවා ඇත්දැයි පරීක්ෂා කිරීමට අවශ්‍ය වේ. එම අවශ්‍යතාවය සපුරාලීම සඳහා පහත සඳහන් කුමන දත්ත වලංගු කිරීමේ ක්‍රම සුදුසුද?**

   1) Type check  2) Length check  3) Range check  4) Presence check  5) Format check

   **Answer: 4)**

   **Explanation / පැහැදිලි කිරීම:**

   Presence check is specifically designed to verify if data is present in a field. This is exactly what the manager needs to ensure all data fields in the course registration form are filled.
   Presence **පරීක්ෂාව විශේෂයෙන් නිර්මාණය කර ඇත්තේ ක්ෂේත්‍රයක දත්ත තිබේද යන්න තහවුරු කිරීමටය. පාඨමාලා ලියාපදිංචි කිරීමේ පෝරමයේ සියලුම දත්ත ක්ෂේත්‍ර පුරවා ඇති බව සහතික කිරීමට කළමනාකරුට අවශ්‍ය වන්නේ මෙයයි.**

   1) **Type check / වර්ග පරීක්ෂාව**
   Verifies if the data is of the correct data type (e.g., number, text).
   **දත්ත නිවැරදි දත්ත වර්ගය (උදා. අංකය, පාඨ) දැයි තහවුරු කරයි.**

   **2) Length check**
   Ensures data is within a specified length.
   **දත්ත නිශ්චිත දිගක් (length) තුළ ඇති බව සහතික කරයි.**

   **3) Range check / පරාසය පරීක්ෂා කිරීම**
   Confirms data falls within a predefined range.
   **දත්ත පූර්ව නිශ්චිත පරාසයක් තුළට වැටෙන බව තහවුරු කරයි.**

   **4) Format check / ආකෘති පරීක්ෂාව**
   Verifies data adheres to a specific format (e.g., date, time).
   **දත්ත නිශ්චිත ආකෘතියකට අනුගත බව තහවුරු කරයි (උදා: දිනය, වේලාව).**

   So, the correct answer is 4).
   **ඒ අනුව, නිවැරදි පිළිතුර වන්නේ** 4).

2. In 1890 ..... used punch card concepts to invent ..... Most suitable words that can be used to fill in the blanks are,
1890 දී ..... විසින් සිදුරුපත් ක්‍රමය භාවිත කරමින්..... නිපදවිය. හිස්තැන් පිරවීම සඳහා සුදුසු වචන වන්නේ,

1) Joseph jacquard, Electrical weaving loom
2) Charles Babbage , Analytical Engine
3) Howard Aiken, Mark 1
4) Blesie Pascal, Adding Machine
5) Herman Hollerith, Electric Mechanical Calculator

**Answer: 5)**

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

**Option / විකල්පය 1)**
Joseph Jacquard used punch card technology to control a mechanical loom, which was a precursor to computing devices. However, this was not in 1890 and the term "Electrical Weaving Loom" is not accurate. Jacquard's loom was a mechanical device, not an electrical one.
Joseph Jacquard පරිගණක උපාංග සඳහා පූර්වගාමියා වූ යාන්ත්‍රික රෙදි වියනක් පාලනය කිරීමට සිදුරු පත් තාක්ෂණය භාවිතා කළේය. කෙසේ වෙතත්, මෙය 1890 දී සිදු නොවූ අතර "Electrical Weaving Loom" යන යෙදුම නිවැරදි නොවේ. Jacquard ගේ රෙදි වියන යන්ත්‍ර උපකරණයක් මිස විද්‍යුත් එකක් නොවේ.

**Option / විකල්පය 2)**
Charles Babbage is known for designing the Analytical Engine, a mechanical general-purpose computer. However, Babbage's work was primarily in the 1830s and 1840s, not 1890.
Charles Babbage යනු යාන්ත්‍රික පොදු කාර්ය පරිගණකයක් වන විශ්ලේෂණ එන්ජිම සැලසුම් කිරීම සඳහා ප්‍රසිද්ධය. කෙසේ වෙතත්, Charles Babbage ගේ කාර්යය මූලික වශයෙන් 1830 සහ 1840 ගණන්වල මිස 1890 නොවේ.

**Option / විකල්පය 3)**
Howard Aiken was involved with the development of the Mark I (also known as the IBM Automatic Sequence Controlled Calculator), which was completed in 1944. This does not fit the 1890 timeframe or the use of punch cards.
Howard Aiken 1944 දී නිම කරන ලද Mark I (IBM Automatic Sequence Controlled Calculator ලෙසද හැඳින්වේ) සංවර්ධනයට සම්බන්ධ විය. මෙය 1890 කාල රාමුවට හෝ සිදුරු පත් භාවිතයට නොගැලපේ.

**Option / විකල්පය 4)**
Blaise Pascal invented an early mechanical calculator known as the Pascaline in the 17th century. This device did not use punch cards and is not related to the 1890s.
Blaise Pascal විසින් 17 වන සියවසේදී Pascaline ලෙස හැඳින්වෙන මුල් යාන්ත්‍රික ගණක යන්ත්‍රයක් සොයා ගන්නා ලදී. මෙම උපාංගය සිදුරු පත් භාවිතා නොකළ අතර 1890 ගණන්වලට සම්බන්ධ නොවේ.

**Option / විකල්පය 5)**
Herman Hollerith used punch card technology to invent a mechanical calculator designed for processing census data, which was significant in 1890. This machine, known as the Hollerith Electric Mechanical Calculator, was instrumental in the development of early data processing systems.
Herman Hollerith විසින් සංගණන දත්ත සැකසීම සඳහා නිර්මාණය කරන ලද යාන්ත්‍රික කැල්කියුලේටරයක් නිර්මාණය කිරීම සඳහා සිදුරු පත් තාක්ෂණය භාවිතා කරන ලද අතර එය 1890 දී සැලකිය යුතු විය. හොලරිත් විද්‍යුත් යාන්ත්‍රික කැල්කියුලේටරය ලෙස හැඳින්වූ මෙම යන්ත්‍රය මුල් දත්ත සැකසුම් පද්ධති සංවර්ධනය සඳහා උපකාරී විය.

The correct pair is Herman Hollerith and Electric Mechanical Calculator because Hollerith used punch card technology to develop his calculator in the 1890s, making option 5 the most accurate choice.
නිවැරදි යුගල Herman Hollerith සහ Electric Mechanical Calculator වේ, මන්ද Hollerith 1890 ගණන්වල ඔහුගේ කැල්කියුලේටරය සංවර්ධනය කිරීමට සිදුරු පත් තාක්ෂණය භාවිතා කළ අතර, විකල්පය 5 වඩාත් නිවැරදි තේරීම වේ.

3. When the two binary numbers $10110100_2 + 01000111_2$ are added, consider that the resulting number is a complement of the two and find its decimal value.
$10110100_2 + 01000111_2$ යන ද්විමය සංඛ්‍යා දෙක එකතු කල විට ලැබෙන සංඛ්‍යාව දෙකෙහි අනුපූරකයක් යැයි සලකා එහි දශමය අගය සොයන්න.

1) 5 2) 251 3) -251 4) -5 5) 4

**Answer: 4)**

**Explanation / පැහැදිලි කිරීම:**

First add 2 binary numbers together.
පළමුව ද්විමය සංඛ්‍යා 2ක එකට එකතු කල යුතුය.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |

Then the two's complement with the binary sum should be converted to a decimal number as follows.
ඉන්පසු එම ද්විමය එකතුව සහිත දෙකෙහි අනුපූරකය පහත පරිදි දශමය සංඛ්‍යාවකට පරිවර්ථනය කර ගත යුතුය.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| Invert(1's Compliment) | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| Add 1 | + | | | | | | | 1 |
| 2's Compliment | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

| $00000101_2$ | | |
|---|---|---|
| 1 | 0 | 1 |
| $2^2$ | $2^1$ | $2^0$ |
| 4 | 2 | 1 |
| 4+1 = 5 | | |
| $5_{10}$ | | |

Since the first digit is 1, this is a negative number.
මෙහි පළමු සංඛ්‍යාංකය 1 බැවින්, මෙය සෘණ සංඛ්‍යාවක් වේ.

So, the answer is 4) -5.
එබැවින්, පිළිතුර 4) -5 වේ.

4. What is the corresponding BCD code for the following numbers?
පහත සංඛ්‍යා සඳහා ගැලපෙන BCD කේතයන් පිළිවෙලින් දැක්වෙන පිළිතුර වන්නේ කුමක්ද?

A. 1950 B. 1248 C. 1111

1) 0001100101010000 , 1111111111111111 , 0000000011111000
2) 1100101010000 , 1001001001000 , 1000100010001000
3) 0001100101010000 , 0001001001001000 , 0001000100010001
4) 0001001001001000 , 0001100101010000 , 0001000100010001
5) 1000100010001 , 1100101010000 , 1001001001000

**Answer: 3)**

**Explanation / පැහැදිලි කිරීම:**

Binary-Coded Decimal (BCD) is a form of binary encoding where each digit of a number is represented by its own binary sequence. Each decimal digit (0-9) is represented by a 4-bit binary number.
Binary-Coded Decimal (BCD) යනු ද්විමය කේතිකරණ ආකාරයකි, එහිදී සංඛ්‍යාවක එක් එක් ඉලක්කම් එහි ද්විමය අනුක්‍රමය මගින් නිරූපණය කෙරේ. සෑම දශම ඉලක්කමක්ම (0-9) 4-bit ද්විමය අංකයකින් නිරූපණය කෙරේ.

Let's convert the given numbers to their corresponding BCD codes.
ලබා දී ඇති සංඛ්‍යා ඒවායේ අනුරූප BCD කේත බවට පරිවර්තනය කරමු.

| **1950** | | | |
|---|---|---|---|
| 1 | 9 | 5 | 0 |
| 0001 | 1001 | 0101 | 0000 |
| 0001 | 1001 | 0101 | 0000 |

| **1248** | | | |
|---|---|---|---|
| 1 | 2 | 4 | 8 |
| 0001 | 0010 | 0100 | 1000 |
| 0001 | 0010 | 0100 | 1000 |

| **1111** | | | |
|---|---|---|---|
| 1 | 1 | 1 | 1 |
| 0001 | 0001 | 0001 | 0001 |
| 0001 | 0001 | 0001 | 0001 |

Accordingly, the BCD codes for each number are,
ඒ අනුව එක් එක් සංඛ්‍යා සඳහා BCD කේතයන් වන්නේ,

- 1950 – 0001 1001 0101 0000
- 1248 – 0001 0010 0100 1000
- 1111 – 0001 0001 0001 0001

So, the correct answer is 3).
ඒ අනුව, නිවැරදි පිළිතුර 3) වේ.

5. $A.\overline{B} \oplus A.B.\overline{C}$

Solve this boolean expression.
මෙම බූලියානු ප්‍රකාශනය සුළු කරන්න.

1) $A.B + \overline{C}$ 2) $A.B.\overline{C} + A.C$ 3) $B.C + \overline{C}.A$ 4) $\underline{A.\overline{B} + A.\overline{C}}$ 5) $A.\overline{B}.C$

**Answer: 4)**

This Boolean expression can be simplified as follows.
මෙම බූලියානු ප්‍රකාශනය පහත ආකාරයට සුළු කළ හැකිය.

$A\overline{B} \oplus AB\overline{C}$

| | |
|---|---|
| $\overline{A\overline{B}}.AB\overline{C} + A\overline{B}.\overline{AB\overline{C}}$ | $A \oplus B = \overline{A}B + A\overline{B}$ |
| $(\overline{A} + \overline{\overline{B}}).AB\overline{C} + A\overline{B}.\overline{AB\overline{C}}$ | De Morgan's law / ද මුවාර් ප්‍රමේයය |
| $(\overline{A} + B).AB\overline{C} + A\overline{B}.\overline{AB\overline{C}}$ | Double inverse law / ද්විත්ව ප්‍රතිලෝම න්‍යාය |
| $(\overline{A} + B).AB\overline{C} + A\overline{B}.(\overline{A} + \overline{B} + \overline{\overline{C}})$ | De Morgan's law / ද මුවාර් ප්‍රමේයය |
| $(\overline{A} + B).AB\overline{C} + A\overline{B}.(\overline{A} + \overline{B} + C)$ | Double inverse law / ද්විත්ව ප්‍රතිලෝම න්‍යාය |
| $\overline{A}AB\overline{C} + BAB\overline{C} + A\overline{B}\overline{A} + A\overline{B}\overline{B} + A\overline{B}C$ | Distributive law / විෂටන න්‍යාය |
| $0B\overline{C} + BAB\overline{C} + A\overline{B}\overline{A} + A\overline{B}\overline{B} + A\overline{B}C$ | Inverse law / ප්‍රතිලෝම න්‍යාය |
| $0 + BAB\overline{C} + A\overline{B}\overline{A} + A\overline{B}\overline{B} + A\overline{B}C$ | Identity law / සර්වසාම්‍ය න්‍යාය |
| $BAB\overline{C} + A\overline{B}\overline{A} + A\overline{B}\overline{B} + A\overline{B}C$ | Identity law / සර්වසාම්‍ය න්‍යාය |

| | |
|---|---|
| $A\bar{B}\bar{C} + A\bar{B}\bar{A} + A\bar{B}\bar{B} + A\bar{B}C$ | Idempotent law / තදේවභාවී න්‍යාය |
| $A\bar{B}\bar{C} + \bar{B}0 + A\bar{B}\bar{B} + A\bar{B}C$ | Inverse law / ප්‍රතිලෝම න්‍යාය |
| $A\bar{B}\bar{C} + 0 + A\bar{B}\bar{B} + A\bar{B}C$ | Identity law / සර්වසාම්‍ය න්‍යාය |
| $A\bar{B}\bar{C} + A\bar{B}\bar{B} + A\bar{B}C$ | Identity law / සර්වසාම්‍ය න්‍යාය |
| $A\bar{B}\bar{C} + A\bar{B} + A\bar{B}C$ | Idempotent law / තදේවභාවී න්‍යාය |
| $A\bar{B}\bar{C} + A\bar{B}$ | Redundancy law / සමතිරික්ත න්‍යාය |
| $A(B\bar{C} + \bar{B})$ | Distributive law / විඝටන න්‍යාය |
| $A(\bar{C} + \bar{B})$ | Redundancy law / සමතිරික්ත න්‍යාය |
| $A\bar{C} + A\bar{B}$ | Distributive law / විඝටන න්‍යාය |

So, the correct answer number here is 4).
එමනිසා, මෙහි නිවැරදි පිළිතුරු අංකය 4) වේ

6. Which of the following statements is equivalent to the Boolean statement?
පහත සඳහන් කුමන ප්‍රකාශය බූලීය ප්‍රකාශයට සමාන ද?

$(\overline{BD} + DD)1 + BC$

1) D 2) BD 3) BC 4) 1 5) B

**Answer: 4)**

This Boolean expression can be simplified as follows.
මෙම බූලියානු ප්‍රකාශනය පහත ආකාරයට සුළු කළ හැකිය.

| | |
|---|---|
| $(\overline{BD} + DD)1 + BC$ | |
| $(\bar{B} + \bar{D} + DD)1 + BC$ | De Morgan's Law / ද මුවාර් ප්‍රමේයය |
| $\bar{B} + \bar{D} + DD + BC$ | Identity Law / සර්වසාම්‍ය න්‍යාය |
| $\bar{B} + \bar{D} + D + BC$ | Idempotent Law / තදේවභාවී න්‍යාය |
| $\bar{B} + BC + 1$ | Complement Law / ප්‍රතිලෝම න්‍යාය |
| 1 | Identity Law / සර්වසාම්‍ය න්‍යාය |

So, the correct answer number here is 4).
එමනිසා, මෙහි නිවැරදි පිළිතුරු අංකය 4) වේ.

7. How many inputs and outputs does a full adder have?
පූර්ණ ආකලකයට ආදාන සහ ප්‍රතිදාන කියක් තිබේ ද?

1) 2 inputs and 1 output / ආදාන 2 ක් සහ ප්‍රතිදානය 1 ක්
2) 3 inputs and 2 outputs / ආදාන 3 ක් සහ ප්‍රතිදාන 2 ක්
3) 2 inputs and 2 outputs / ආදාන 2 ක් සහ ප්‍රතිදාන 2 ක්
4) 4 inputs and 2 outputs / ආදාන 4 ක් සහ ප්‍රතිදාන 2 ක්
5) 3 inputs and 1 output / ආදාන 3 ක් සහ ප්‍රතිදාන 1 ක්

**Answer: 2)**

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

**1) 2 inputs and 1 output**
According to this option, it's false. A full adder has more than two inputs and more than one output.
මෙම විකල්පයට අනුව, එය නිවැරදි නොවේ. පූර්ණ ආකලකයකට ආදාන දෙකකට වඩා සහ ප්‍රතිදානයකට වඩා ඇත.

**2) 3 inputs and 2 outputs**
According to this option, it's true. A full adder is designed to add three binary digits (two significant bits and one carry-in) and produces two outputs: the sum and the carry-out. This is the correct configuration for a full adder.
මෙම විකල්පය අනුව, එය නිවැරදි වේ. පූර්ණ ආකලකයකට ද්විමය ඉලක්කම් තුනක් (සැලකිය යුතු බිටු දෙකක් සහ එක් රැගෙන යාමක්) එකතු කිරීමට සැලසුම් කර ඇති අතර ප්‍රතිදානයන් දෙකක් නිපදවයි: එකතුව සහ ගෙනයාම. පූර්ණ ආකලකය සඳහා නිවැරදි වින්‍යාසය මෙයයි.

**3) 2 inputs and 2 outputs**
According to this option, it's false. While a full adder does have two outputs, it requires three inputs, not two.
මෙම විකල්පයට අනුව, එය නිවැරදි නොවේ. පූර්ණ ආකලකයකට ප්‍රතිදාන දෙකක් ඇති අතර, එයට ආදාන තුනක් අවශ්‍ය වේ, දෙකක් නොවේ.

**4) 4 inputs and 2 outputs**
According to this option, it's false. A full adder only needs three inputs, not four.
මෙම විකල්පයට අනුව, එය නිවැරදි නොවේ. පූර්ණ ආකලකයකට අවශ්‍ය වන්නේ ආදාන තුනක් මිස හතරක් නොවේ.

**5) 3 inputs and 1 output**
According to this option, it's false. A full adder has two outputs, not just one.
මෙම විකල්පයට අනුව, එය නිවැරදි නොවේ. පූර්ණ ආකලකයකට ප්‍රතිදානයන් දෙකක් ඇත, එකක් පමණක් නොවේ.

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

8. Which Python data structure is used to store key-value pairs?
යතුරු අගය යුගල ගබඩා කිරීමට භාවිතා කරන පයිතන් දත්ත ව්‍යුහය කුමක් ද?

1) List 2) Tuple 3) Set 4) Dictionary 5) Array

**Answer: 4)**

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

**1) List**
According to this option, it's false. A list in Python stores ordered collections of items but does not use key-value pairs. Lists are indexed by position.
මෙම විකල්පයට අනුව, එය අසත්‍ය වේ. Python ගබඩාවල ඇති ලැයිස්තුවක් අයිතම එකතු කිරීම් ඇණවුම් කළ නමුත් යතුරු අගය යුගල භාවිතා නොකරයි. ස්ථාන අනුව ලැයිස්තුගත කර ඇත.

**2) Tuple**
According to this option, it's false. A tuple in Python is an ordered collection of items, similar to a list, but it is immutable. Tuples also do not use key-value pairs.
මෙම විකල්පයට අනුව, එය අසත්‍ය වේ. Python හි tuple යනු ලැයිස්තුවකට සමාන, ඇණවුම් කළ අයිතම් එකතුවකි, නමුත් එය වෙනස් කළ නොහැක. Tuple ද යතුරු අගය යුගල භාවිතා නොකරයි.

**3) Set**
According to this option, it's false. A set in Python is a collection of unique items without duplicates. Sets do not store key-value pairs; they only store values.
මෙම විකල්පයට අනුව, එය අසත්‍ය වේ. Python හි Set යනු අනුපිටපත් නොමැතිව අනන්‍ය අයිතම එකතුවකි. Set යතුරු අගය යුගල ගබඩා නොකරයි; ඔවුන් ගබඩා කරන්නේ අගයන් පමණි.

**4) Dictionary**
According to this option, it's true. A dictionary in Python is specifically designed to store key-value pairs. Each key is associated with a value, and you can access, add, or modify these pairs.
මෙම විකල්පය අනුව, එය සත්‍ය වේ. පයිතන් හි dictionary විශේෂයෙන් නිර්මාණය කර ඇත්තේ key-value යුගල ගබඩා කිරීම සඳහා ය. සෑම යතුරක්ම අගයක් සමඟ සම්බන්ධ වී ඇති අතර, ඔබට මෙම යුගල වෙත ප්‍රවේශ වීමට, එකතු කිරීමට හෝ වෙනස් කිරීමට හැකිය.

**5) Array**
According to this option, it's false. An array in Python (using libraries like `array` or `numpy`) typically stores elements in a sequence but does not use key-value pairs.
**මෙම විකල්පයට අනුව, එය අසත්‍ය වේ.** Python **හි අරාවක්** (using libraries like `array` **හෝ** `numpy`**වැනි** libraries **භාවිතා කරමින් )** **සාමාන්‍යයෙන් මූලද්‍රව්‍ය අනුපිළිවෙලක ගබඩා කරන නමුත් යතුරු අගය යුගල භාවිතා නොකරයි.**

So, the correct answer is 4).
**ඒ අනුව, නිවැරදි පිළිතුර වන්නේ** 4).

9. What is the output of the following code?
**පහත කේතයේ ප්‍රතිදානය කුමක්ද?**

```
nums = [10, 20]
temp = nums
nums.append([30, 40])
temp.remove(10)
nums.extend([50])
temp.clear()
nums.append(60)
print(nums)
```

1) `[20, [30, 40], 50, 60]`
2) `[60]`
3) `[[30, 40], 50, 60]`
4) `[10, 20, [30, 40], 50, 60]`
5) `[50,60]`

**Answer: 2)**

**Explanation / පැහැදිලි කිරීම:**

The list nums is first defined as [10, 20], and temp is set to reference the same list (not a copy). So changes to either nums or temp affect the same underlying list.
**ලැයිස්තු අංක මුලින්ම** 10, 20], **ලෙස අර්ථ දක්වා ඇති අතර, තාවකාලිකය එකම ලැයිස්තුවට යොමු වීමට සකසා ඇත (පිටපතක් නොවේ). එබැවින් සංඛ්‍යා හෝ තාවකාලික වෙනස්කම් එකම යටින් පවතින ලැයිස්තුවට බලපායි.**

NOTE:
In Python, the = operator assigns a reference of a list (or any object) to another variable, meaning both variables point to the same object in memory.
Python **හි,** = **ක්‍රියාකරු ලැයිස්තුවක (හෝ ඕනෑම වස්තුවක) යොමුවක් වෙනත් විචල්‍යයකට පවරයි, එනම් විචල්‍ය දෙකම මතකයේ ඇති එකම වස්තුවකට යොමු කරයි.**

So, changing one affects the other. On the other hand, the **.copy()** method creates a shallow copy of the original list.
**ඉතින්, එකක් වෙනස් කිරීම අනෙකට බලපායි. අනෙක් අතට,** **.copy()** **ක්‍රමය මුල් ලැයිස්තුවේ නොගැඹුරු පිටපතක් නිර්මාණය කරයි.**

This means the new list has the same elements, but it is stored at a different memory location, so changes to one do not affect the other
**මෙයින් අදහස් කරන්නේ නව ලැයිස්තුවේ එකම අංග ඇති නමුත් එය වෙනත් මතක ස්ථානයක ගබඩා කර ඇති බැවින් එකකට කරන වෙනස්කම් අනෙකට බලපාන්නේ නැති බවයි**

When we do append([30, 40]), it adds the whole list [30, 40] as one item, not separate numbers.
append([30, 40]), **කරන විට, එය සම්පූර්ණ ලැයිස්තුව** [30, 40] **එක් අයිතමයක් ලෙස එකතු කරයි, වෙනම සංඛ්‍යා ලෙස නොවේ.**
nums=[ 10, 20, ,[30,40]]

Then remove(10) takes out the number 10. Next, extend([50]) adds 50 to the end of the list.
ඉන්පසු remove(10මඟින් 10 අංකය ඉවත් කරයි. ඊළඟට, extend([50]) මඟින් ලැයිස්තුවේ අවසානයට 50 ක් එකතු කරයි.
nums=[20 ,[30,40],50 ]

When we call clear(), it empties the whole list because temp and nums are the same.
clear(), ලෙස ඇමතූ විට, එය මුළු ලැයිස්තුවම හිස් කරයි, මන්ද temp සහ nums සමාන වේ.
nums=[]
temp=[]

Finally, append(60) adds the number 60 to the empty list. So, the list becomes [60] in the end.
අවසානයේ, append(60) හිස් ලැයිස්තුවට 60 අංකය එකතු කරයි. එබැවින්, ලැයිස්තුව අවසානයේ [60] බවට පත්වේ.
nums=[60]

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

10. What is the output of the following Python code?
පහත පයිතන් කේතයේ ප්‍රතිදානය කුමක් ද?

```
for A, *B in (('ictfromabc','new','website !'),('make','your','account','now')):
    print('A is %s and B is %s' % (A, B))
```

1) `A is ictfromabc and B is ['new', 'website !']`
   `A is make and B is ['your', 'account', 'now']`
2) `A is ictfromabc and B is ['new', 'website !']`
   `A is make and B is ['your', 'account']`
3) `A is ictfrom and B is ['new', 'website !']`
   `A is make and B is ['your', 'account', 'now']`
4) `A is ictfromabc and B is ['new', 'website']`
   `A is make and B is ['your', 'account', 'now']`
5) `A is ictfromabc and ['new', 'website !']`
   `A is make and B is ['your', 'account', 'now']`

**Answer: 1)**

**Explanation / පැහැදිලි කිරීම:**

The Python code processes two tuples by assigning the first element of each tuple to `A` and the remaining elements to `B` as a list. In the first tuple, `A` becomes `'ictfromabc'` and `B` is `['new', 'website !']`. In the second tuple, `A` becomes `'make'` and `B` is `['your', 'account', 'now']`. The `print` statement then outputs these values, which aligns with option 1), making it the correct answer.
පයිතන් කේතය එක් එක් tuple එකෙහි පළමු මූලද්‍රව්‍යය `A` ට සහ ඉතිරි මූලද්‍රව්‍ය `B` වෙත ලැයිස්තුවක් ලෙස ලබා දීමෙන් tuple දෙකක් සකසයි. පළමු tuple එකෙහි `A`, `'ictfromabc'` බවට පත් වන අතර `B`, `['new', 'website !']` වේ. දෙවන tuple එකෙහි `A` `'make'` බවට පත් වන අතර `B` යනු `['your', 'account', 'now']` වේ. `print` ප්‍රකාශය මෙම අගයන් ප්‍රතිදානය කරයි, එය 1) විකල්පය සමඟ සමපාත වේ, එය නිවැරදි පිළිතුර බවට පත් කරයි.