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 235 MARKING

2025/08/02
ictfromabc
Ravindu Bandaranayake

1. Which of the following best describes the difference between DRAM and SRAM in terms of internal architecture and performance?
අභ්‍යන්තරය සහ කාර්ය සාධනය අනුව DRAM සහ SRAM අතර වෙනස වඩාත් හොඳින් විස්තර කරන්නේ පහත සඳහන් ඒවායින් කුමක් ද?

    1) DRAM is faster and uses flip-flops / DRAM වේගවත් වන අතර පිළි-පොළ භාවිතා කරයි.
    2) SRAM is slower and uses capacitors / SRAM මන්දගාමී වන අතර ධාරිත්‍රක භාවිතා කරයි.
    3) <u>DRAM requires refresh cycles and uses capacitors</u>
    DRAM සඳහා නැවුම් කිරීමේ චක්‍ර අවශ්‍ය වන අතර ධාරිත්‍රක භාවිතා කරයි.
    4) SRAM stores data magnetically / SRAM දත්ත චුම්භකව ගබඩා කරයි
    5) DRAM is non-volatile and requires no refresh
    DRAM නෂ්‍ය නොවන අතර නැවුම් කිරීමක් අවශ්‍ය නොවේ

Answer: 3)

Explanation / පැහැදිලි කිරීම;

RAM (Random Access Memory) is the main memory of a computer where the CPU temporarily stores data and instructions while executing programs. It is volatile, meaning all data is lost when the power is off.
RAM (සසම්භාවී ප්‍රවේශ මතකය) යනු පරිගණකයක ප්‍රධාන මතකය වන අතර එහිදී CPU වැඩසටහන් ක්‍රියාත්මක කරන අතරතුර දත්ත සහ උපදෙස් තාවකාලිකව ගබඩා කරයි. එය නෂ්‍ය වේ, එනම් විදුලිය විසන්ධි වූ විට සියලු දත්ත නැති වී යයි.

Let's go through the given options. / ලබා දී ඇති විකල්ප හරහා යමු.

Option / විකල්පය 1)
DRAM is slower, and it does not use flip-flops. SRAM does
DRAM මන්දගාමී වන අතර, එය flip-flops භාවිතා නොකරයි. SRAM එසේ කරයි

Option / විකල්පය 2)
SRAM is faster than DRAM and uses flip-flops, not capacitors.
SRAM , DRAM වලට වඩා වේගවත් වන අතර ධාරිත්‍රක නොව flip-flops භාවිතා කරයි.

Option / විකල්පය 3)
DRAM stores each bit in a tiny capacitor, which leaks charge over time. That's why DRAM must be refreshed
DRAM එකෙහි සෑම බිට් එකක්ම කුඩා ධාරිත්‍රකයක ගබඩා කරන අතර එමඟින් කාලයත් සමඟ ආරෝපණය කාන්දු වේ. ඒ නිසා DRAM යාවත්කාලීන කළ යුතුය.

Option / විකල්පය 4)
SRAM stores data in transistor-based flip-flops, not using any magnetic technology. Magnetic storage refers to hard drives or MRAM, not SRAM.
SRAM දත්ත ගබඩා කරන්නේ ට්‍රාන්සිස්ටර මත පදනම් වූ flip-flops වල වන අතර, කිසිදු චුම්භක තාක්ෂණයක් භාවිතා නොකරයි. චුම්භක ගබඩාව යන SRAM නොව දෘඪ තැටි හෝ MRAMය.

Option / විකල්පය 5)
DRAM is volatile, meaning it loses all data when power is lost, and it does require refreshing.
DRAM නෂ්‍ය වේ, එනම් බලය නැති වූ විට එහි සියලු දත්ත නැති වන අතර එය නැවුම් කිරීම අවශ්‍ය වේ.

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

2. Which of the following memory types is volatile and directly accessible by the CPU during execution?
ක්‍රියාත්මක කිරීමේදී CPU මඟින් නෂ්‍ය සහ සෘජුවම ප්‍රවේශ විය හැකි පහත සඳහන් මතක වර්ග මොනවාද?

1) ROM
2) EEPROM
3) Cache Memory / හැඹිලි මතකය
4) Virtual Memory / අතථ්‍ය මතකය
5) Flash Memory / සැනෙලි මතකය

Answer: 3)

Explanation / පැහැදිලි කිරීම;

Cache memory loses its data when the power is turned off.
බලය අක්‍රිය කළ විට හැඹිලි මතකය එහි දත්ත නැති වේ.

The CPU uses cache memory to store frequently accessed data and instructions, as it's much faster than main memory (RAM). This direct and rapid access is crucial for improving system performance.
ප්‍රධාන මතකයට CPU වඩා බොහෝ වේගවත් බැවින්, නිතර ප්‍රවේශ වන දත්ත සහ උපදෙස් ගබඩා කිරීම සඳහා (RAM) හැඹිලි මතකය භාවිතා කරයි. පද්ධති ක්‍රියාකාරිත්වය වැඩි දියුණු කිරීම සඳහා මෙම සෘජු හා වේගවත් ප්‍රවේශය ඉතා වැදගත් වේ.

ROM (Read-Only Memory)
This is non-volatile and cannot be written to during normal operation. It stores essential boot-up instructions.
මෙය වෙනස් නොවන අතර සාමාන්‍ය ක්‍රියාකාරිත්වය අතරතුර ලිවිය නොහැක. එය අත්‍යවශ්‍ය ආරම්භක උපදෙස් ගබඩා කරයි.

EEPROM (Electrically Erasable Programmable Read-Only Memory)
This is a type of non-volatile memory that can be erased and reprogrammed. It is not used for direct CPU access during execution.
මෙය මකා දැමිය හැකි සහ නැවත වැඩසටහන්ගත කළ හැකි වාෂ්පශීලී නොවන මතක වර්ගයකි. ක්‍රියාත්මක කිරීමේදී සෘජු කජඹ ප්‍රවේශය සඳහා එය භාවිතා නොවේ.

Virtual Memory / අතථ්‍ය මතකය
This is a memory management technique that uses a portion of a hard drive as if it were RAM. It is not a physical memory component directly accessed by the CPU in the same way as cache. Accessing virtual memory is significantly slower than accessing physical RAM or cache.
මෙය RAM එකක් මෙන් දෘඩ තැටියක කොටසක් භාවිතා කරන මතක කළමනාකරණ තාඬණයකි. එය හැඹිලිය මෙන් CPU මගින් සෘජුවම ප්‍රවේශ වන භෞතික මතක සංරචකයක් නොවේ. අථතය මතකයට ප්‍රවේශ වීම භෞතික RAM හෝ හැඹිලියට ප්‍රවේශ වීමට වඩා සැලකිය යුතු ලෙස මන්දගාමී වේ.

Flash Memory
This is a type of non-volatile memory, commonly used in solid-state drives (SSDs) and USB drives. It retains data without power.
මෙය SSD සහ USB ධාවක වල බහුලව භාවිතා වන වෙනස් නොවන මතක වර්ගයකි. එය බලය නොමැතිව දත්ත රඳවා ගනී.

Therefore, the correct answer is, 3).
එබැවින්, නිවැරදි පිළිතුර වන්නේ, 3) වේ.

3. What is the decimal equivalent of the 2's complement of binary 11100110_2?
ද්විමය 11100110_2 හි 2 හි අනුපූරකයේ දශම සමානතාවය කුමක්ද?

1) -26 2) -25 3) -22 4) -27 5) -30

Answer: 1)

Explanation / පැහැදිලි කිරීම;

To find the magnitude of the number, take the 2's complement of the binary value.
සංඛ්‍යාවේ විශාලත්වය සොයා ගැනීමට, ද්විමය අගයේ 2 හි අනුපූරකය ගන්න.

1	1	1	0	0	1	1	0
0	0	0	1	1	0	0	1
							+1
0	0	0	1	1	0	1	0
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1
16 + 8 + 2 = 26							

Since the most significant bit (the leftmost bit) is a 1, the number is negative.
වඩිම වෙසෙසි බිටුව (වම් කෙළවරේ ඇති බිට් එක) 1 වන බැවින්, අංකය සෘණ වේ.

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

4. The second, third, and fourth rows of the following table contain three English words and their binary representations according to the ASCII code. The binary representation of **"Start!"** is kept blank.
පහත වගුවේ දෙවන, තෙවන සහ සිව්වන පේළි වල ඉංග්‍රීසි වචන තුනක් සහ ASCII කේතයට අනුව ඒවායේ ද්විමය නිරූපණයන් අඩංගු වේ. **"Start!"** හි ද්විමය නිරූපණය හිස්ව තබා ඇත.

Word	Binary representation
star	01110011 01110100 01100001 01110010
S!	01010011 00100001
Sta!	01010011 01110100 01100001 00100001
Start!	

1) 01010011 01110100 01100001 01110010 00100001 00100001
2) 01110011 01100001 01110100 01110010 00100001 00100001
3) 01010011 01110100 01100001 01110010 01110100 00100001
4) 01110100 01110010 01010011 01100001 00100001 00100001
5) 00100001 01010011 01110100 01100001 01110010

Answer : 3)

ASCII Binary Codes:

S → 01010011
t → 01110100
a → 01100001
r → 01110010
t → 01110100
! → 00100001

The correct binary representation for **"Start!"** is: **01010011 01110100 01100001 01110010 01110100 00100001**
"Start!" සඳහා නිවැරදි ද්විමය නිරූපණය වන්නේ: **01010011 01110100 01100001 01110010 01110100 00100001**

Matching with Given Answer Choices / ලබා දී ඇති පිළිතුරු සමඟ ගැලපීම:
1) 01010011 01110100 01100001 01110010 00100001 00100001 → Incorrect (extra "!")
2) 01110011 01100001 01110100 01110010 00100001 00100001 → Incorrect (wrong order)
4) 01110100 01110010 01010011 01100001 00100001 00100001 → Incorrect (wrong order)
5) 00100001 01010011 01110100 01100001 01110010 → Incorrect (wrong order)

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

5. How many universal gates are required to build a 2-input XOR gate using only NAND gates?
NAND ද්වාර පමණක් භාවිතා කර 2-ආදාන XOR ද්වාරයක් තැනීමට universal ද්වාර කියක් අවශ්‍ය වේද?

1) 3 2) 4 3) 5 4) 6 5) 7

Answer: 2)

Explanation / පැහැදිලි කිරීම;

1. **First NAND Gate:**

   Inputs: A and B / A සහ B
   Output: $\overline{(AB)}$

   This gate generates a key intermediate signal that is used by the next two gates.
   මෙම ද්වාරය ඊළඟ දොරටු දෙක විසින් භාවිතා කරන ප්‍රධාන අතරමැදි සංඥාවක් ජනනය කරයි.

2. **Second NAND Gate:**

   Inputs: A and the output of the first gate, $\overline{(AB)}$ / A සහ පළමු ද්වාරයේ ප්‍රතිදානය, $\overline{(AB)}$
   Output: $\overline{(A.\overline{(AB)})}$

3. **Third NAND Gate:**

   Inputs: B and the output of the first gate, $\overline{(AB)}$ / B සහ පළමු ද්වාරයේ ප්‍රතිදානය, $\overline{(AB)}$
   Output: $\overline{(B.\overline{(AB)})}$

4. **Fourth NAND Gate:**

   Inputs: The outputs from the second and third gates. / දෙවන සහ තෙවන ද්වාර වලින් පිටවන ද්වාරය.
   Output: $\overline{\overline{(A.\overline{(AB)})}.\ \overline{(B.\overline{(AB)})}}$

This final output is the exact Boolean expression for an XOR gate.
මෙම අවසාන ප්‍රතිදානය XOR ද්වාරයක් සඳහා නිශ්චිත බූලියන් ප්‍රකාශනය වේ.

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

6. Which answer contains SOP and POS expressions obtained by the following Karnaugh map respectively?
පිළිවෙළින් පහත කානෝ සිතියම මගින් ලබාගත් SOP සහ POS ප්‍රකාශන අඩංගු පිළිතුර කුමක්ද?

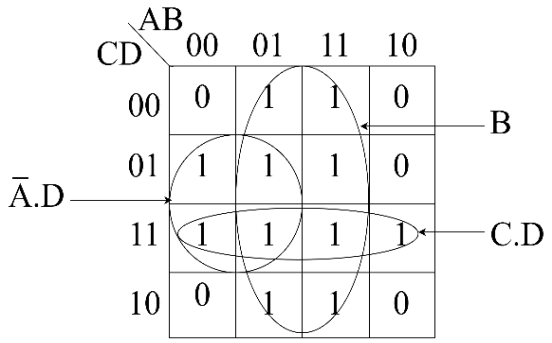
CD \ AB	00	01	11	10
00	0	1	1	0
01	1	1	1	0
11	1	1	1	1
10	0	1	1	0

1) SOP = $(\bar{A}.\bar{B}) + (A.B.C) + (B.\bar{D})$, POS = $(A + \bar{B}).(\bar{A} + C + \bar{D}).(\bar{A} + B + \bar{D})$
2) SOP = $B + (\bar{C}.D) + (A.C.\bar{D})$, POS = $(A + B + \bar{C}).(B + C + D).(A + B + D).(B + \bar{C} + \bar{D})$
3) SOP = $B + \bar{A}.D + C.D$, POS = $(B + C + D).(\bar{A} + B + C).(B + \bar{C} + D)$
4) SOP = $(A + \bar{B}).(\bar{A} + C + \bar{D}).(\bar{A} + B + \bar{D})$, POS = $(\bar{A}.\bar{B}) + (A.B.C) + (A.C.\bar{D}) + (A.\bar{C}.\bar{D})$
5) SOP = $(\bar{A}.\bar{B}) + (A.\bar{B}.C) + (\bar{A}.C.\bar{D}) + (A.\bar{C}.D)$, POS = $(\bar{A} + B).(A + C + \bar{D}).(\bar{A} + \bar{B} + D)$

Answer : 3)

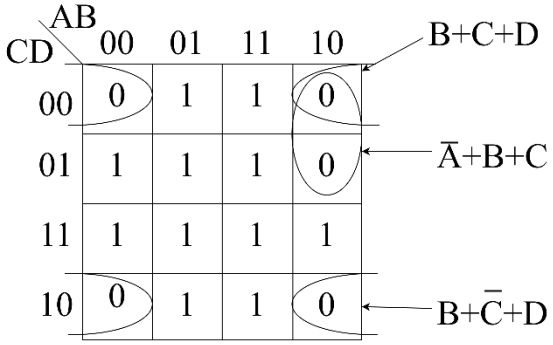
Explanation / පැහැදිලි කිරීම;

SOP =



SOP = B+$(\bar{A}.D)$+(C.D)

POS=



POS = (B+C+D). ($\bar{A}$+B+C). (B+$\bar{C}$+D)

So the answer will be 3). / **එබැවින් පිළිතුර** 3) **වේ.**

SOP = B+$(\bar{A}.D)$+(C.D), POS = (B+C+D). ($\bar{A}$+B+C). (B+$\bar{C}$+D)

7. Each block of a disk is 256 bytes. When a file of size 2025 bytes is stored on that disk, how many bytes allocated to the file would be wasted?
තැටියක සෑම block එකක්ම බයිට් 256 කි. බයිට් 2025 ප්‍රමාණයේ ගොනුවක් එම තැටියේ ගබඩා කර ඇති විට, ගොනුවට විභජනය කල බයිට් කොපමණ ප්‍රමණායක් අපතේ යයිද?

1) 89 2) 256 3) 279 4) 1792 5) 23

Answer : 5)

Explanation / පැහැදිලි කිරීම:

To determine the number of wasted bytes, follow these steps:
අපතේ යන බයිට් ගණන තීරණය කිරීම සඳහා, මෙම පියවර අනුගමනය කරන්න:

Block Size: Each block is 256 bytes.
බ්ලොක් ප්‍රමාණය: සෑම බ්ලොක් එකක්ම බයිට් 256 කි.

File Size: The file size is 2025 bytes.
ගොනු ප්‍රමාණය: ගොනු ප්‍රමාණය බයිට් 2025 කි.

Number of Blocks Needed: Since each block can store 256 bytes, we calculate:
අවශ්‍ය Blocks ගණන: සෑම block එකකටම බයිට් 256ක් ගබඩා කළ හැකි බැවින්, අපි ගණනය කරන්නේ:

Number of blocks =
Blocks ගණන =2025/256

=7.91=8 blocks (blocks)

Total Allocated Bytes: Since each block is 256 bytes, and 8 blocks are allocated,
මුළු වෙන් කළ බයිට්: සෑම බ්ලොක් එකක්ම බයිට් 256ක් වන අතර, blocks 8ක් වෙන් කර ඇති නිසා,

=8×256=2048 bytes (බයිට්)

Wasted Space: The unused space in the last allocated block is:
නාස්ති කරන ලද ඉඩ: අවසන් වරට වෙන් කළ කොටසෙහි භාවිතයට නොගත් ඉඩ:

=2048−2025 = 23 bytes

So the answer is 5).
එබැවින් පිළිතුර 5) වේ.

23 bytes are wasted due to internal fragmentation.
අභ්‍යන්තර බණ්ඩනය නිසා බයිට් 23ක් අපතේ යයි.

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

```
a = 10
b = 3
result = (a + b) // 2 + (a % b) * (a & b) - (a | b)
print(result)
```

1) -8 2) 9 3) 5 4) -3 5) -5

Answer: 4)

Explanation / පැහැදිලි කිරීම;

The given Python code calculates a result based on the variables a = 10 and b = 3, using a combination of arithmetic and bitwise operations.
දී ඇති පයිතන් කේතය, අංක ගණිත සහ බිට්වයිස් මෙහෙයුම්වල සංයෝජනයක් භාවිතා කරමින්, a = 10 and b = 3 **යන විචල්‍යයන් මත පදනම්ව ප්‍රතිඵලයක් ගණනය කරයි.**

Let's evaluate each part of the expression carefully.
ප්‍රකාශනයේ සෑම කොටසක්ම ප්‍රවේශමෙන් ඇගයීමට ලක් කරමු.

1.(a+b)
=10 + 3 = 13

2.(a% b)
Finds the remainder when one number (the dividend) is divided by another number.
එක් සංඛ්‍යාවක් (ලාභාංශය) තවත් සංඛ්‍යාවකින් බෙදූ විට ඉතිරිය සොයා ගනී.
=(10 % 3) = 1

3.(a & b) [Bitwise AND]
= (10 & 3)

	8	4	2	1
10	1	0	1	0
3	0	0	1	1
10 AND 3	0	0	1	0

= 2

4.(a | b) [Bitwise OR]
= (10 | 3)

	8	4	2	1
10	1	0	1	0
3	0	0	1	1
10 OR 3	1	0	1	1

=11

Result = 13 // 2 +1 *2 -11

Combining these parts with operator precedence (multiplication before addition/subtraction)
මෙම කොටස් ක්‍රියාකරු ප්‍රමුඛතාවය සමඟ ඒකාබද්ධ කිරීම (එකතු කිරීමට/අඩු කිරීමට පෙර ගුණ කිරීම)
result = 13 // 2 +1 *2 -11

10 and 3 to get 13, then performs integer division by 2, resulting in 6
13 **ලබා ගැනීමට** 10 **සහ** 3, **පසුව නිඛිල** 2 **න් බෙදීම සිදු කරයි, ප්‍රතිඵලයක් ලෙස** 6 **ලැබේ.**
= 6 +1 * 2 – 11
= 6 +2 -11
=8 – 11
= -3

Therefore, the answer is 4).
එබැවින් පිළිතුර 4) **වේ.**

9. Which of the following best describes the difference between mutable and immutable data types in Python?
පයිතන් හි mutable සහ immutable දත්ත වර්ග අතර වෙනස වඩාත් හොඳින් විස්තර කරන්නේ පහත සඳහන් ඒවායින් කුමක් ද?

1) Mutable objects can be copied; immutable objects cannot
Mutable වස්තූන් පිටපත් කළ හැකිය immutable වස්තූන් කළ නොහැක
2) Immutable objects can be modified in place; mutable objects cannot
Immutable වස්තූන් ස්ථානයේදීම වෙනස් කළ හැකිය mutable එසේ කළ නොහැක
3) Mutable objects can be changed after creation; immutable objects cannot
මැවීමෙන් පසු mutable වෙනස් කළ හැකිය immutable වස්තූන්ට එසේ කළ නොහැක
4) Only strings are immutable; all others are mutable
immutable වන්නේ strings පමණි අනෙක් සියල්ල mutable
5) Mutable and immutable objects behave the same in memory
මතකයේ දී mutable සහ immutable වස්තූන් එකම ආකාරයකින් ක්‍රියා කරයි.

Answer: 3)

Explanation / පැහැදිලි කිරීම;

In Python, mutable objects (like lists, dictionaries, and sets) can be changed after they are created. For example, you can add or remove elements in a list.
Python හි, mutable objects (like lists, dictionaries, and sets) නිර්මාණය කිරීමෙන් පසු වෙනස් කළ හැක. උදාහරණයක් ලෙස, ඔබට ලැයිස්තුවකට මූලද්‍රව්‍ය එකතු කිරීමට හෝ ඉවත් කිරීමට හැකිය.

Immutable objects (like strings, integers, tuples) cannot be changed once they are created any operation on them creates a new object in memory.
Immutable objects (strings, පූර්ණ සංඛ්‍යා, tuples වැනි) නිර්මාණය කළ පසු වෙනස් කළ නොහැක, ඒවා මත සිදුවන ඕනෑම මෙහෙයුමක් මතකයේ නව වස්තුවක් නිර්මාණය කරයි.

Let's go through the given options. / ලබා දී ඇති විකල්ප හරහා යමු.

Option / විකල්පය 1)
Both types can be copied in Python, just in different ways
වර්ග දෙකම පයිතන් හි පිටපත් කළ හැකිය, නමුත් විවිධ ආකාරවලින්

Option / විකල්පය 2)
This is the opposite of the truth. Mutable objects can be changed in place. Immutable ones cannot be changed after they are made.
මෙය සත්‍යයේ ප්‍රතිවිරුද්ධයයි. Mutable objects ස්ථානයේදීම වෙනස් කළ හැකිය. Immutable ඒවා සෑදූ පසු වෙනස් කළ නොහැක.

Option / විකල්පය 3)
This is the correct answer. Mutable means you can change the data (like a list). Immutable means you cannot change the data (like a string).
මෙය නිවැරදි පිළිතුරයි. Mutable යන්නෙන් අදහස් කරන්නේ ඔබට දත්ත වෙනස් කළ හැකි බවයි (ලැයිස්තුවක් වැනි). Immutable යන්නෙන් අදහස් කරන්නේ ඔබට දත්ත වෙනස් කළ නොහැකි බවයි (string වැනි).

Option / විකල්පය 4)
Strings are immutable, but so are numbers and tuples. Lists and dictionaries are examples of mutable objects.
Strings immutable ය, නමුත් සංඛ්‍යා සහ tuples ද එසේමය. ලැයිස්තු සහ dictionaries විකෘති කළ හැකි වස්තූන් සඳහා උදාහරණ වේ.

Option / විකල්පය 5)
Wrong. When you change a mutable object, it stays in the same memory place. But changing an immutable object creates a new one in memory.
වැරදියි. ඔබ විකෘති වස්තුවක් වෙනස් කරන විට, එය එකම මතක ස්ථානයේ පවතී. නමුත් immutable වස්තුවක් mutable මතකයේ නව එකක් නිර්මාණය වේ.

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

10. Consider the dictionary: `d = {'x':[1,2], 'y':[3,4]}`
මෙම dictionary එක සලකන්න: `d = {'x':[1,2], 'y':[3,4]}`

What does `d['x'][1]` return?
`d['x'][1]` ආපසු ලබා දෙන්නේ කුමක්ද?

1) `1` 2) `2` 3) `x` 4) `[1,2]` 5) `Error`

Answer: 3)

Explanation / පැහැදිලි කිරීම;

`d['x']`
This part of the expression accesses the value associated with the key `'x'` in the dictionary `d`. The value is the list `[1,2]`.
ප්‍රකාශනයේ මෙම කොටස `d` dictionary යේ `'x'` සමඟ සම්බන්ධිත අගයට ප්‍රවේශ වේ. අගය ලැයිස්තුව `[1,2]` වේ.

`[1]`
This part of the expression accesses the element at index `1` of the list `[1,2]`. In Python, list indexing starts at `0`.
ප්‍රකාශනයේ මෙම කොටස ලැයිස්තුවේ `[1,2]` දර්ශකයේ `1` හි මූලද්‍රව්‍යයට ප්‍රවේශ වේ. පයිතන් හි, ලැයිස්තු සුචිගත කිරීම `0` න් ආරම්භ වේ.

- Index `0` corresponds to the value `1`.
  දර්ශකය `0` අගය `1` ට අනුරූප වේ.

- Index `1` corresponds to the value `2`.
  දර්ශකය `1` අගය `2` ට අනුරූප වේ.

Therefore the correct answer is, 3).
එබැවින් නිවැරදි පිළිතුර වන්නේ, 3) වේ.