සියලු ම හිමිකම් ඇවිරිණි / All Rights Reserved]

Bandaranayake - ictfromabc Ravindu Bandaranayake - ictfromabc Ravindu

Information and Communication Technology
තොරතුරු හා සන්නිවේදන තාක්ෂණය

2026 QUIZ NO 353 MARKING

2025/11/28
ictfromabc
Ravindu Bandaranayake

1. Which of the following data storage devices has the largest capacity?
   පහත දත්ත ගබඩා කරන උපාංග අතරින් වැඩිම ධාරිතාවක් සහිත උපාංගය කුමක්ද?

   1) Floppy Disk
   2) Solid State Drive
   3) Pen Drive
   4) Magnetic Tape
   5) DVD

   Answer: 4)
   Magnetic tapes are renowned for their high storage capacity and cost-effectiveness, making them a favored choice for long-term data storage and archival needs. Their durability ensures data integrity over time, while sequential access facilitates efficient storage and retrieval processes. However, they suffer from slower access speeds compared to modern storage solutions like SSDs or HDDs, limiting their suitability for applications requiring rapid data access. Additionally, magnetic tapes lack the convenience of random access, are physically delicate, and demand specialized management systems for optimal operation. Despite these drawbacks, their significant storage capabilities and economical nature continue to make them a viable option for organizations seeking large-scale data storage solutions.
   චුම්බක පටි දිගු කාලීන දත්ත ගබඩා කිරීම සහ ලේඛනාගාර අවශ්‍යතා සඳහා හොඳ තේරීමක් වන අතර ඉහළ ගබඩා ධාරිතාව සහ පිරිවැය-ඵලදායීතාවය සඳහා ප්‍රසිද්ධය. ඒවායේ කල්පැවැත්ම කාලයත් සමඟ දත්ත අඛණ්ඩතාව සහතික කරයි, අනුක්‍රමික ප්‍රවේශය කාර්යක්ෂම ගබඩා කිරීම සහ නැවත ලබා ගැනීමේ ක්‍රියාවලීන් සඳහා පහසුකම් සපයයි. කෙසේ වෙතත්, ඔවුන් SSD හෝ HDD වැනි නවීන ගබඩා විසඳුම් සමඟ සසඳන විට මන්දගාමී ප්‍රවේශ වේගයකින් සහිතය, වේගවත් දත්ත ප්‍රවේශයක් අවශ්‍ය යෙදුම් සඳහා මෙය යෝග්‍යතාවය අඩුය. මීට අමතරව, චුම්බක පටිවලට අහඹු ලෙස ප්‍රවේශ වීමේ පහසුව නොමැත, භෞතිකව සියුම් වන අතර ප්‍රශස්ත ක්‍රියාකාරිත්වය සඳහා විශේෂිත කළමනාකරණ පද්ධති ඉල්ලා සිටී. මෙම අඩුපාඩු තිබියදීත්, ඒවායේ සැලකිය යුතු ගබඩා හැකියාවන් සහ ආර්ථිකමය ස්වභාවය මහා පරිමාණ දත්ත ගබඩා විසඳුම් සොයන සංවිධාන සඳහා ඒවා ශක්‍ය විකල්පයක් බවට පත් කරයි.

2. Who Invented the first Automatic sequence Calculator? (Mark I)
   පළමු ස්වයංක්‍රිය අනුක්‍රමණ ගණක යන්ත්‍රය සොයාගත්තේ කවුද? (මාර්ක් I)

   1) Blaise Pascal බ්ලේස් පැස්කල්
   2) Charles Babbage චාල්ස් බැබේජ්
   3) Howard Aiken හොවර්ඩ් අයිකන්
   4) Alan Turing ඇලන් ටියුරින්
   5) John von Neumann ජෝන් වොන් නියුමාන්

   Answer:3)

   The first automatic sequence calculator, known as the Mark I, was invented by Howard Aiken. He built the Mark I between 1937 and 1945 at Harvard University. The Mark I was a large electromechanical computer that could perform complex calculations automatically.While other individuals like Blaise Pascal, Charles Babbage, Alan Turing, and John von Neumann made significant contributions to the field of computing, they were not directly responsible for inventing the first automatic sequence calculator, the Mark I.
   මාර්ක් (1) ලෙස හැඳින්වෙන පළමු ස්වයංක්‍රිය අනුක්‍රමික කැල්කියුලේටරය හොවාර්ඩ් අයිකන් විසින් සොයා ගන්නා ලදී. ඔහු හාවඩ් විශ්ව විද්‍යාලයේ (1937) සහ (1945) අතර මාර්ක් 1 නිර්මාණය කර ඇත. මාර්ක් 1 යනු සංකීර්ණ ගණනය කිරීම් ස්වයංක්‍රියව සිදු කළ හැකි විශාල විද්‍යුත් යාන්ත්‍රික පරිගණකයකි.බ්ලේස් පැස්කල්, චාල්ස් බැබේජ්ල ඇලන් ටියුරින් සහ ජෝන් වොන් නියුමාන් වැනි අනෙකුත් පුද්ගලයින් පරිගණක ක්ෂේත්‍රයට සැලකිය යුතු දායකත්වයක් ලබා දුන් අතර, පළමු ස්වයංක්‍රිය අනුක්‍රමික කැල්කියුලේටරය වන මාර්ක් 1 සොයා ගැනීම සම්බන්ධයෙන් ඔවුන් සෘජුව වගකිව යුතු නොවේ.

Blaise Pascal is known for inventing the mechanical calculator called the Pascaline in the 17th century. Charles Babbage is considered the "father of the computer" for his work on the Analytical Engine and the Difference Engine, but he did not build the first computer.
බ්ලේස් පැස්කල් 17 වන සියවසේ පැස්කල් නම් යාන්ත්‍රික ගණක යන්ත්‍රය සොයා ගැනීම සඳහා ප්‍රසිද්ධය. චාල්ස් බැබේජ් පරිගණකයේ පියා ලෙස සැලකෙන්නේ විශ්ලේෂණාත්මක එන්ජිම (Analytical Engine) සහ Difference Engine පිළිබඳ ඔහුගේ වැඩ සඳහා, නමුත් ඔහු පළමු පරිගණකය ගොඩනඟා නැත.

Alan Turing is known for his contributions to computer science and the development of the concept of a universal machine, but he did not invent the first automatic sequence calculator.John von Neumann, while influential in the development of computer architecture and programming, did not invent the first automatic sequence calculator.Therefore, the correct answer is (3) Howard Aiken.
ඇලන් ටියුරින් පරිගණක විද්‍යාව සඳහා ඔහුගේ දායකත්වය සහ විශ්වීය යන්ත්‍රයක්(universal machine) පිළිබඳ සංකල්පය වර්ධනය කිරීම සඳහා ප්‍රසිද්ධය, නමුත් ඔහු පළමු ස්වයංක්‍රිය අනුක්‍රමික කැල්කියුලේටරය (sequence calculator) නිර්මාණය කළේ නැත.ජෝන් වොන් නියුමන්, පරිගණක ගෘහ නිර්මාණ ශිල්පය සහ ක්‍රමලේඛනය සංවර්ධනයට බලපෑ අතර, පළමු ස්වයංක්‍රිය අනුක්‍රමික කැල්කියුලේටරය (automatic sequence calculator) සොයා ගත්තේ නැත.එබැවින්, නිවැරදි පිළිතුර (3) හොවර්ඩ් අයිකන් වේ.

3. How much memory capacity does the EBCDIC coding system use to represent one character.?
EBCDIC කේත ක්‍රමය මගින් එක් අනුලක්ෂණයක් නිරූපණය කිරීම සඳහා යොදා ගනු ලබන මතක ධාරිතාව වන්නේ?

1) 4 bit 2) <u>8 bit</u> 3) 16 bit 4) 48 bit 5) 7 bit

Answer:2)

The EBCDIC coding system uses 8 bits to represent one character.
EBCDIC කේත ක්‍රමය මගින් එක් අනුලක්ෂණයක් නිරූපණය කිරීම සඳහා බිටු 8 යොදා ගනී.

The ASCII coding system uses 7 bits to represent one character.
ASCII කේත ක්‍රමය මගින් එක් අනුලක්ෂණයක් නිරූපණය කිරීම සඳහා බිටු 7 යොදා ගනී.

The BCD coding system uses 4 bits to represent one character.
BCD කේත ක්‍රමය මගින් එක් අනුලක්ෂණයක් නිරූපණය කිරීම සඳහා බිටු 4 යොදා ගනී.

Unicode uses 16 bits to represent one character.
යුනිකේත ක්‍රමය මගින් එක් අනුලක්ෂණයක් නිරූපණය කිරීම සඳහා බිටු 16 යොදා ගනී.

4. $\overline{1.C} + \overline{D}(D.A + \overline{D.A})$

Solve this boolean expression.
මෙම බූලියානු ප්‍රකාශනය සුළු කරන්න.

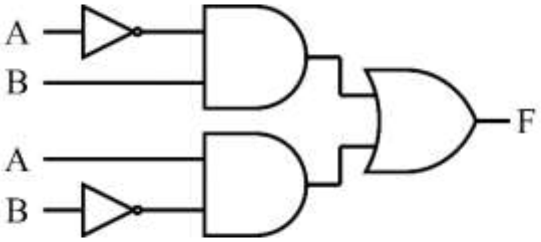
1) $\overline{A} + \overline{D}$ 2) $\overline{C}.\overline{D}$ 3) $A.C.\overline{D}$ 4) <u>$\overline{C} + \overline{D}$</u> 5) $A + \overline{C} + \overline{D}$

Answer : 4)

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

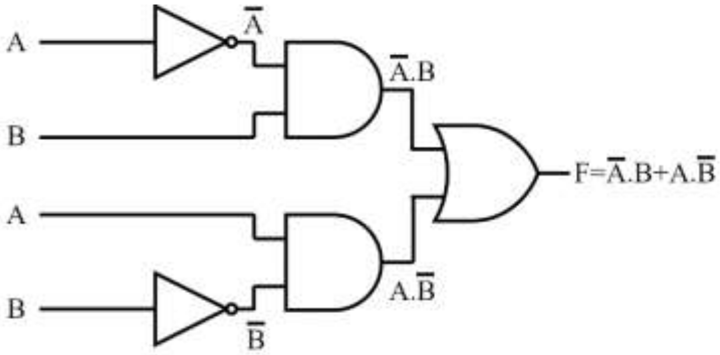
$\overline{1.C} + \overline{D}(D.A + \overline{D.A})$	
$\overline{1} + \overline{C} + \overline{D}(D.A + \overline{D.A})$	De Morgan's law ද මුවාර් ප්‍රමේයය
$\overline{1} + \overline{C} + \overline{D}(D.A + \overline{D} + \overline{A})$	De Morgan's law ද මුවාර් ප්‍රමේයය
$0 + \overline{C} + \overline{D}(D.A + \overline{D} + \overline{A})$	$\overline{1} = 0$
$\overline{C} + \overline{D}(D.A + \overline{D} + \overline{A})$	Identity law සර්වසාම්‍ය න්‍යාය
$\overline{C} + \overline{D}(\overline{D} + A + \overline{A})$	Redundancy law සමතිරික්ත න්‍යාය
$\overline{C} + \overline{D}(\overline{D} + 1)$	Inverse law ප්‍රතිලෝම න්‍යාය
$\overline{C} + \overline{D}(1)$	Identity law සර්වසාම්‍ය න්‍යාය
$\overline{C} + \overline{D}$	Identity law සර්වසාම්‍ය න්‍යාය

5. What is the logic gate represented by the following logic circuit?
පහත තාර්කික පරිපථයෙන් නිරූපණය කරන තාර්කික ද්වාරය කුමක් ද?



1) NOR 2) AND 3) <u>XOR</u> 4) XNOR 5) OR

Answer : 3)



Here is the truth table related to the above logic circuit.
මෙහි දැක්වෙන්නේ ඉහත සඳහන් තාර්කික පරිපථයට අදාළ සත්‍යතා වගුවයි.

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

Here is the truth table for XOR gate.
මෙහි දැක්වෙන්නේ XOR ද්වාරයට අදාළ සත්‍යතා වගුව යි.

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

Since the output from the above logic circuit is the same as the output related to the XOR gate, it appears that the logic circuit related to the XOR gate is shown here.
ඉහත තාර්කික පරිපථයෙන් ලැබෙන ප්‍රතිදානය හා XOR ද්වාරයට අදාළ ප්‍රතිදානය සමාන බැවින් මෙහි දැක්වෙන්නේ XOR ද්වාරයට අදාළ තාර්කික පරිපථය බව පෙනී යයි.

6. When converting an ER diagram to a relational database:
ER රූප සටහනක් සම්බන්ධතා දත්ත ගබඩාවක් බවට පරිවර්තනය කරන විට:

How is a one-to-many relationship typically represented in tables?
ඒක-බහු සම්බන්ධතාවක් සාමාන්‍යයෙන් වගු තුළ නිරූපණය වන්නේ කෙසේද?

1) By adding a column with the entity name.
හුතාර්ථ නාමය සමඟ තීරුවක් එක් කිරීමෙනි.

2) By creating separate tables with a foreign key in one of them.
ඒවායින් එකක ආගන්තුක යතුරක් සහිත වෙනම වගු නිර්මාණය කිරීමෙනි.

3) By combining all entities into a single table.
සියලුම හුතාර්ථ තනි වගුවකට ඒකාබද්ධ කිරීමෙනි.

4) By duplicating entity attributes across multiple tables.
වගු කිහිපයක් හරහා හුතාර්ථ උපලක්ෂණ අනුපිටපත් කිරීමෙනි.

5) By removing relationships entirely.
සබඳතා සම්පූර්ණයෙන්ම ඉවත් කිරීමෙනි.

Answer : 2)

Explanation / පැහැදිලි කිරීම

In a relational database, a one-to-many relationship is typically represented by creating separate tables where one of the tables contains a foreign key that links to the primary key of the other table. This approach is best described by Option 2. For example, in a relationship between Customers and Orders, each customer can have multiple orders. In this case, the Orders table would include a foreign key referencing the Customer ID in the Customers table. Option 1 is incorrect because a simple column with an entity name does not establish a relationship. Option 3 is wrong as combining entities into a single table would violate normalization rules. Option 4 is also incorrect because duplicating attributes across tables leads to redundancy and inefficiency. Lastly, Option 5 is incorrect since relationships are essential in a relational database to maintain proper connections between tables and ensure data integrity.
සම්බන්ධතා දත්ත සමුදායක් තුළ, ඒක-බහු සම්බන්ධතාවක් සාමාන්‍යයෙන් නිරූපණය කරනු ලබන්නේ වෙනම වගු සෑදීමෙනි, එහිදී එක් වගුවක අනෙක් වගුවේ ප්‍රාථමික යතුරට සම්බන්ධ වන ආගන්තුක යතුරක් අඩංගු වේ. මෙම ප්‍රවේශය විකල්ප 2 මගින් වඩාත් හොඳින් විස්තර කෙරේ. උදාහරණයක් ලෙස, පාරිභෝගිකයින් සහ ඇණවුම් අතර සම්බන්ධතාවයකදී, සෑම පාරිභෝගිකයෙකුටම බහුවිධ ඇණවුම් තිබිය හැක. මෙම අවස්ථාවෙහිදී, ඇණවුම් වගුවේ පාරිභෝගිකයින්ගේ වගුවේ පාරිභෝගික හැඳුනුම්පත යොමු කරන ආගන්තුක යතුරක් ඇතුළත් වේ. හුතාර්ථ නාමයක් සහිත සරල තීරුවක් සම්බන්ධතාවයක් ස්ථාපිත නොකරන නිසා විකල්ප 1 වැරදියි. තනි වගුවකට හුතාර්ථ ඒකාබද්ධ කිරීම ප්‍රමටකරණ රීති උල්ලංඝනය කරන බැවින් විකල්ප 3 වැරදිය. වගු හරහා උපලක්ෂණ අනුපිටපත් කිරීම අතිරික්තය සහ අකාර්යක්ෂමතාවයට හේතු වන බැවින් විකල්ප 4 ද වැරදිය. අවසාන වශයෙන්, වගු අතර නිසි සම්බන්ධතා පවත්වා ගැනීමට සහ දත්ත අඛණ්ඩතාව සහතික කිරීමට සම්බන්ධතා දත්ත සමුදායක් තුළ සම්බන්ධතා අත්‍යවශ්‍ය බැවින් විකල්ප 5 වැරදිය.

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

7. Which of the following scenarios represents redundancy in a database?
දත්ත සමුදායක අනුපිටපත් වීම නියෝජනය කරන්නේ පහත දැක්වෙන කුමන අවස්ථාද?

1) An "Order" table contains multiple records of orders placed by the same customer.
"Order" වගුවක එකම පාරිභෝගිකයා විසින් කරන ලද ඇණවුම් පිළිබඳ වාර්තා කිහිපයක් අඩංගු වීම.

2) A customer name appears as a foreign key in an "Invoice" table.
"Invoice" වගුවක ආගන්තුක යතුරක් ලෙස පාරිභෝගික නාමයක් දිස්වීම.

3) A table has missing rows for some customers who didn’t make a purchase.
මිලදී ගැනීමක් සිදු නොකළ සමහර පාරිභෝගිකයින් සඳහා වගුවක පේළි හිස්ව ඇත.

4) A database stores customer contact information in both the "Customer" and "Order" tables.
දත්ත සමුදායක් පාරිභෝගික සම්බන්ධතා තොරතුරු "Customer" සහ "Order" වගු දෙකෙහිම ගබඩා කරයි.

5) None of the above.
ඉහත කිසිවක් නොවේ.

Answer : 4)

Explanation / පැහැදිලි කිරීම

Redundancy in a database occurs when the same data is unnecessarily duplicated across multiple tables. In Option 4, customer contact information is stored in both the "Customer" and "Order" tables, which is an example of redundancy. This duplication not only wastes storage but also increases the likelihood of inconsistencies when data is updated, as changes made in one table may not be reflected in the other.
එකම දත්ත වගු කිහිපයක් හරහා අනවශ්‍ය ලෙස අනුපිටපත් කරන විට දත්ත සමුදායක අතිරික්තය සිදුවේ. විකල්ප 4 හි, පාරිභෝගික සම්බන්ධතා තොරතුරු "Customer" සහ "Order" වගු දෙකෙහිම ගබඩා කර ඇත, එය අතිරික්තයට උදාහරණයකි. මෙම අනුපිටපත් කිරීම ගබඩාව නාස්ති කරනවා පමණක් නොව දත්ත යාවත්කාලීන කරන විට නොගැලපීම් ඇතිවීමේ සම්භාවිතාව වැඩි කරයි, මන්ද එක් වගුවක සිදු කරන ලද වෙනස්කම් අනෙක් වගුවේ පිළිබිඹු නොවිය හැකිය.

In contrast, the other options do not represent redundancy. Option 1 involves multiple records of orders for the same customer, which is normal for tracking individual transactions. Option 2 establishes a relationship between tables using a foreign key, which is standard practice and avoids duplication. Option 3 indicates incomplete data rather than duplicated data. Therefore, Option 4 is the correct answer as it highlights unnecessary duplication.
ඊට වෙනස්ව, අනෙකුත් විකල්ප අතිරික්තය නියෝජනය නොකරයි. විකල්ප 1 එකම පාරිභෝගිකයා සඳහා ඇණවුම් පිළිබඳ බහුවිධ වාර්තා ඇතුළත් වේ, එය තනි ගනුදෙනු නිරීක්ෂණය සඳහා සාමාන්‍ය වේ. විකල්ප 2 විදේශ යතුරක් භාවිතා කරමින් වගු අතර සම්බන්ධතාවයක් ඇති කරයි, එය සම්මත භාවිතයක් වන අතර අනුපිටපත් කිරීම වළක්වයි. විකල්ප 3 අනුපිටපත් දත්ත වලට වඩා අසම්පූර්ණ දත්ත දක්වයි. එබැවින් අනවශ්‍ය අනුපිටපත් ඉස්මතු කරන බැවින් විකල්ප 4 නිවැරදි පිළිතුර වේ.

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

```
def ABC(x, y=[]):
    y.append(x)
    return y

print(ABC(1))
print(ABC(3, []))
```

1) [1]
[3]

2) [1, 3]

3) [3]
[1]

4) [3, 1]
[3]

5) Syntax error

Answer : 1)

Let's consider the process of this python code step by step.
මෙම පයිතන් කේතයේ ක්‍රියාවලිය පියවරෙන් පියවර සලකා බලමු.

```
def ABC(x, y=[]):
    y.append(x)
    return y
```

The function ABC is defined with two parameters: x and y. The parameter y has a default value of an empty list ([ ]).
ABC ශ්‍රිතය x සහ y ලෙස පරාමිති දෙකකින් අර්ථ දක්වා ඇත. y පරාමිතියට හිස් ලැයිස්තුවක පෙරනිමි අගයක් ඇත ([ ]).

First Call (ABC(1)):
ABC(1) is called with x = 1 and y as the default empty list [ ].
The value 1 is appended to y, so y becomes [1].
The function returns [1] and prints [1].
ABC(1) මගින් x = 1 සහ y පෙරනිමි හිස් ලැයිස්තුව [ ] ලෙස අර්ථ දැක්වේ.
1 අගය y ට එකතු කරයි, එබැවින් y, [1] බවට පත්වේ.
ශ්‍රිතය මගින් [1] පැමිණ [1] මුද්‍රණය කරයි.

Second Call (ABC(3, [])):
ABC(3, []) is called with x = 3 and y explicitly provided as a new empty list [].
The value 3 is appended to y, so y becomes [3].
The function returns [3] and prints [3].
ABC(3, []) මගින් x = 3 සමඟින් අර්ථ දක්වන අතර y පැහැදිලිවම නව හිස් ලැයිස්තුවක් [] ලෙස සපයා ඇත.
3 අගය y ට එකතු කරයි, එබැවින් y, [3] බවට පත්වේ.
ශ්‍රිතය මගින් [3] පැමිණ [3] මුද්‍රණය කරයි.

Accordingly the following output is obtained.
ඒ අනුව පහත ප්‍රතිදානය ලැබේ.

```
[1]
[3]
```

So, the correct answer number is 1.
එමනිසාල නිවැරදි පිළිතුරු අංකය 1 වේ.

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

```
data = {"a": 1, "b": 2, "c": 3}
result = {k: v**2 for k, v in data.items() if v % 2 == 1}
print(result)
```

1) ['a': 1, 'c': 9]
2) {'a', 'c'}
3) {1, 9}
4) {'a': 1, 'c': 9}
5) {1: 'a', 9: 'c'}

Answer : 4)

Let's consider the process of this python code step by step.
මෙම පයිතන් කේතයේ ක්‍රියාවලිය පියවරෙන් පියවර සලකා බලමු.

```
data = {"a": 1, "b": 2, "c": 3}
```
A dictionary named data is created with three key-value pairs.
data නම් dictionary එකක් යතුරු-අගය යුගල තුනක් සමඟ නිර්මාණය කර ඇත.

Key 'a' has the value 1 / 'a' යතුරට 1 අගය ඇත
Key 'b' has the value 2 / 'b' යතුරට 2 අගය ඇත
Key 'c' has the value 3 / 'c' යතුර 3 අගය ඇත

```
result = {k: v**2 for k, v in data.items() if v % 2 == 1}
```
This line uses a dictionary comprehension to create a new dictionary called `result` by iterating over each key-value pair in `data`.
`data` හි එක් එක් යතුරු අගය යුගලය මත පුනරාවර්තනය කිරීමෙන් `result` නමින් නව dictionary එකක් නිර්මාණය කිරීමට මෙම පේළිය භාවිත කරයි.

`data.items()` returns an iterable view of key-value pairs, allowing the `for k, v in data.items()` part to loop through each (`k, v`) pair in the dictionary.
`data.items()` යතුරු-අගය යුගලවල පුනරාවර්තනයක් ලබා දෙයි, `for k, v in data.items()` කොටස dictionary එකෙහි එක් එක් (`k, v`) යුගල හරහා ලූප් කිරීමට ඉඩ සලසයි.

`if v % 2 == 1` is a conditional statement that checks if the value `v` is odd (i.e., the remainder when divided by 2 is 1).
`if v % 2 == 1` යනු `v` අගය ඔත්තේ ද යන්න පරීක්ෂා කරන කොන්දේසි සහිත ප්‍රකාශයකි. (එනම්, 2 න් බෙදූ විට ඉතිරිය 1 වේ).

For each key-value pair where v is odd, the comprehension creates an entry in result where the key is k and the value is v**2 (the square of v).
v ඔත්තේ වන සෑම යතුරු අගය යුගලයක් සඳහාම, යතුර k සහ අගය v**2 (v හි වර්ගය) වන result සඳහා ප්‍රවේශයක් නිර්මාණය කරයි.

```
print(result)
```
This prints the result dictionary to the console.
මෙය result dictionary එක කොන්සෝලයට මුද්‍රණය කරයි.

The dictionary comprehension filters out only the entries from data where the value is odd and then squares those values.
dictionary මගින් අගය ඔත්තේ සහ පසුව එම අගයන් වර්ග කරන දත්ත පමණක් ඇතුල් කරගනී.

In data, the odd values are 1 (key 'a') and 3 (key 'c'). The value 2 (key 'b') is even, so it is excluded.
දත්තවල, ඔත්තේ අගයන් 1 (key 'a') සහ 3 (key 'c') වේ. අගය 2 (key 'b') ඉරට්ටේ වේ, එබැවින් එය බැහැර කර ඇත.

The result will be {'a': 1, 'c': 9}
ප්‍රතිඵලය වනුයේ {'a': 1, 'c': 9}

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

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

```
n = 1
while n <= 5:
    if n < 3:
        print(f"{n} is less than 3")
    elif n == 3:
        print(f"{n} is equal to 3")
    else:
        if n > 3:
            print(f"{n} is greater than 3")
    n += 1
```

1)
```
1 is less than 3
2 is less than 3
3 is equal to 3
4 is greater than 3
5 is greater than 3
```

2)
```
0 is less than 3
1 is less than 3
2 is equal to 3
3 is greater than 3
4 is greater than 3
```

3)
```
0 is less than 5
1 is less than 5
2 is equal to 5
3 is greater than 5
4 is greater than 5
```

4)
```
1 is less than 5
2 is less than 5
3 is equal to 5
4 is greater than 5
5 is greater than 5
```

5)
```
1 is less than -3
2 is less than -3
3 is equal to -3
4 is greater than -3
5 is greater than -3
```

Answer : 1)

Explanation / පැහැදිලි කිරීම

The given code initializes n to 1 and uses a while loop that runs as long as n is less than or equal to 5. Within the loop, a series of conditions is checked. If n is less than 3, it outputs "{n} is less than 3"; if n equals 3, it outputs "{n} is equal to 3". If neither of these conditions is true, an `else` block executes. Inside this `else` block, an `if` condition checks if n is greater than 3, and, if so, it outputs "{n} is greater than 3". After each conditional check, n is incremented by 1 to move toward the loop's end.

ලබා දී ඇති කේතය n සිට 1 දක්වා ආරම්භ කරන අතර n 5 ට අඩු හෝ සමාන වන තාක් ධාවනය වන while ලූපයක් භාවිතා කරයි. ලූපය තුළ, කොන්දේසි මාලාවක් පරීක්ෂා කරනු ලැබේ. n 3 ට වඩා අඩු නම්, එය ප්‍රතිදානය කරන්නේ "{n} is less than 3" යනුවෙනි, n 3 ට සමාන නම්, එය ප්‍රතිදානය කරන්නේ "{n} is equal to 3" යනුවෙනි. මෙම කොන්දේසි කිසිවක් සත්‍ය නොවේ නම්, else කොටස ක්‍රියාත්මක වේ. මෙම else කොටස තුළ, if කොන්දේසියක් මඟින් n 3 ට වඩා වැඩි දැයි පරීක්ෂා කරයි, සහ, එසේ නම්, එය "{n} is greater than 3" යනුවෙන් ප්‍රතිදානය කරයි. එක් එක් කොන්දේසි පරීක්ෂාවකට පසුව, ලූපයේ අවසානය දෙසට ගමන් කිරීමට n 1 කින් වැඩි කෙරේ.

With n starting at 1, the code evaluates each condition for values of n from 1 to 5. For n = 1 and n = 2, the output will be "1 is less than 3" and "2 is less than 3", respectively. When n is 3, it matches elif n == 3, so the output is "3 is equal to 3." For n = 4 and n = 5, the condition if n > 3 inside the else block is satisfied, resulting in the output "4 is greater than 3" and "5 is greater than 3." So, the complete output will be,
1 න් ආරම්භ වන n සමඟින්, කේතය 1 සිට 5 දක්වා n අගයන් සඳහා එක් එක් කොන්දේසිය ඇගයීමට ලක් කරයි. n = 1 සහ n = 2 සඳහා, ප්‍රතිදානය "1 is less than 3" සහ "2 is less than 3" වේ. n 3 වන විට, එය elif n == 3 ට ගැළපේ, එබැවින් ප්‍රතිදානය "3 is equal to 3." වේ. n = 4 සහ n = 5 සඳහා, else කොටස තුළ ඇති n > 3 කොන්දේසිය සත්‍ය වන අතර, එහි ප්‍රතිඵලයක් ලෙස "4 is greater than 3" සහ "5 is greater than 3." වේ. එබැවින්, සම්පූර්ණ ප්‍රතිදානය වන්නේ,

```
1 is less than 3
2 is less than 3
3 is equal to 3
4 is greater than 3
5 is greater than 3
```

So, the correct answer number is 1).
එබැවින්, නිවැරදි පිළිතුරු අංකය 1) වේ.