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 266 MARKING

2025/09/02
ictfromabc
Ravindu Bandaranayake

1. Which of the following is an advantage of using a Command-Line Interface (CLI)?
විධාන-පේළි අතුරුමුහුණත් (CLI) භාවිතා කිරීමේ වාසිය පහත ඒවායින් කුමක් ද?

   1) It is easier for beginners to learn and use.
   ආරම්භකයින්ට ඉගෙන ගැනීමට සහ භාවිතා කිරීමට පහසු වේ.
   2) It requires fewer system resources compared to graphical interfaces.
   චිත්‍රක අතුරුමුහුණත් වලට සාපේක්ෂව අඩු පද්ධති සම්පත් අවශ්‍ය වේ.
   3) It provides a more visually appealing interface.
   එය වඩාත් දෘශ්‍යමය වශයෙන් ආකර්ෂණීය අතුරු මුහුණතක් සපයයි.
   4) It supports only a limited number of commands and functions.
   එය සීමිත විධාන සහ කාර්යයන් සඳහා පමණක් සහය දක්වයි.
   5) It is not suitable for complex tasks or automation.
   එය සංකීර්ණ කාර්යයන් හෝ ස්වයංක්‍රියකරණය සඳහා සුදුසු නොවේ.

   Answer: 2)

   Explanation / පැහැදිලි කිරීම

   It requires fewer system resources compared to graphical interfaces.
   චිත්‍රක අතුරුමුහුණත් වලට සාපේක්ෂව අඩු පද්ධති සම්පත් අවශ්‍ය වේ.

   CLI uses fewer system resources, making it faster and more efficient, especially for experienced users. It does not rely on graphical elements, so it consumes less memory and processing power.
   CLI අඩු පද්ධති සම්පත් භාවිතා කරයි, එය වේගවත් හා වඩා කාර්යක්ෂම කරයි, විශේෂයෙන් පළපුරුදු පරිශීලකයින් සඳහා. එය චිත්‍රක මූලාංග මත රඳා නොපවතී, එබැවින් එය අඩු මතකයක් සහ සැකසුම් බලයක් පරිභෝජනය කරයි.

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

2. What is the primary function of RAM (Random Access Memory) in a computer?
පරිගණකයක සසම්භාවී ප්‍රවේශ මතකයේ ප්‍රාථමික කාර්යයක් වනුයේ කුමක්ද?

   1) To permanently store data even when the computer is turned off.
   පරිගණකය ක්‍රියා විරහිත වූ විට පවා දත්ත ස්ථිරව ගබඩා කිරීම.
   2) To provide temporary storage for data and programs currently in use.
   දැනට භාවිතා වන දත්ත සහ වැඩසටහන් සඳහා තාවකාලික ගබඩාවක් ලබා දීම.
   3) To manage the computer's hardware resources and operations.
   පරිගණකයේ දෘඪාංග සම්පත් සහ ක්‍රියාකාරකම් කළමනාකරණය කිරීම.
   4) To store the computer's BIOS and firmware settings.
   පරිගණකයේ BIOS සහ ස්ථිරාංග සැකසුම් ගබඩා කිරීම.
   5) To increase the speed of the CPU by caching instructions.
   CPU හි වේගය ඉහළ නැංවීම සඳහා උපදෙස් නශ්‍ය කිරීම.

   Answer: 2)

   Explanation of each option / එක් එක් විකල්පය පැහැදිලි කිරීම:

Option / විකල්ප 1)
This describes non-volatile memory like ROM or storage devices (ex: HDD, SSD).
මෙය ROM හෝ ගබඩා උපාංග (උදා: HDD, SSD) විස්තර කරයි.

Option / විකල්ප 2)
RAM is a type of volatile memory that stores data and programs temporarily while the computer is running, allowing quick access for the CPU.
RAM යනු නශ්‍ය මතකයක් වන අතර, පරිගණකය ක්‍රියාත්මක වන අතරතුර දත්ත සහ වැඩසටහන් තාවකාලිකව ගබඩා කරයි. මෙමගින් CPU වෙත දත්තවලට ඉක්මනින් ප්‍රවේශ වීමට හැකියාව ලබා දේ.

Option / විකල්ප 3)
This is the role of the operating system, not RAM.
මෙය RAM හි නොව මෙහෙයුම් පද්ධතියේ කාර්යයකි.

Option / විකල්ප 4)
This is the function of ROM (Read-Only Memory) or flash memory.
මෙය ROM වල හෝ සැනෙලි මතකයෙහි කාර්යයකි.

Option / විකල්ප 5)
This describes CPU cache memory, not RAM.
මෙය CPU නිහිත මතකය විස්තර කරයි, RAM නොවේ.

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

3. What is the purpose of ASCII code?
ASCII කේතයේ අරමුණ කුමක්ද?

1) To compress images / රූප සම්පිඩනය කිරීමට
2) To represent characters using numbers / සංඛ්‍යා භාවිතයෙන් අනුලක්ෂණ නිරූපණය කිරීමට
3) To encrypt data / දත්ත සංකේතනය කිරීමට
4) To format hard drives / දෘඩ තැටි ආකෘතිකරණයට
5) To block viruses / වයිරස ඇතුලුවීම වැළැක්වීමට

Answer: 2)

Explanation / පැහැදිලි කිරීම

The purpose of ASCII (American Standard Code for Information Interchange) is to represent characters, such as letters, digits, and symbols, using numerical values. Each character is assigned a unique numeric code within a range of 0 to 127 in the standard ASCII set. For example, the letter 'A' is represented by the number 65, and 'a' is represented by the number 97. This system allows computers, which operate on numbers, to efficiently process and communicate textual data. ASCII serves as a universal standard for character encoding, ensuring compatibility across different platforms and devices.
ASCII (තොරතුරු හුවමාරුව සඳහා ඇමරිකානු සම්මත කේතය) හි අරමුණ වන්නේ සංඛ්‍යාත්මක අගයන් භාවිතා කරමින් අකුරු, ඉලක්කම් සහ සංකේත වැනි අනුලක්ෂණ නිරූපණය කිරීමයි. සම්මත ASCII කට්ටලයේ 0 සිට 127 දක්වා පරාසයක් තුළ සෑම අනුලක්ෂණයකටම අනන්‍ය සංඛ්‍යාත්මක කේතයක් පවරනු ලැබේ. උදාහරණයක් ලෙස, 'A' අක්ෂරය අංක 65 මගින් ද 'a' අංකය 97 මගින් ද නිරූපණය කෙරේ. මෙම ක්‍රමය මඟින් සංඛ්‍යා මත ක්‍රියා කරන පරිගණකවලට පාඨ දත්ත කාර්යක්ෂමව සැකසීමට සහ සන්නිවේදනය කිරීමට ඉඩ සලසයි. ASCII විවිධ වේදිකා සහ උපාංග අතර ගැළපුම සහතික කරමින් අනුලක්ෂණ කේතනය සඳහා විශ්වීය සම්මතයක් ලෙස සේවය කරයි.

By encoding characters numerically, ASCII enables seamless data exchange between systems, regardless of the underlying hardware or software. This is essential for applications like text storage, file formats, and network communication, where accurate representation and interpretation of textual data are critical. Although modern character encoding systems like Unicode have expanded upon ASCII to support a wider range of characters, ASCII remains fundamental in computing as a foundation for character representation.
අක්ෂර සංඛ්‍යාත්මකව කේතනය කිරීම මගින්, ASCII මඟින් යටින් පවතින දෘඩාංග හෝ මෘදුකාංග නොසලකා පද්ධති අතර බාධාවකින් තොරව දත්ත හුවමාරුව සක්‍රිය කරයි. පාඨ ආචයනය, ගොනු ආකෘති සහ ජාල සන්නිවේදනය වැනි යෙදුම් සඳහා මෙය අත්‍යවශ්‍ය වේ, එහිදී පාඨමය දත්ත නිවැරදිව නිරූපණය කිරීම සහ අර්ථ නිරූපණය කිරීම ඉතා වැදගත් වේ. යුනිකෝඩ් වැනි නවීන අක්ෂර කේතිකරණ පද්ධති පුළුල් පරාසයක අක්ෂර සඳහා සහය දැක්වීම සඳහා ASCII මත ව්‍යාප්ත වී ඇතත්, අනුලක්ෂණ නිරූපණය සඳහා පදනමක් ලෙස ASCII පරිගණකකරණයේදී මූලිකව පවතී.

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

4. What is the 2's complement representation of -82?
-82 හි දෙකෙහි අනුපූරක නියෝජනය වනුයේ කුමක්ද?

1) 10100000 2) 10100001 3) 10101101 4) 01010010 5) <u>10101110</u>

Answer: 5)

Explanation / පැහැදිලි කිරීම:

Here, the question asks to get the 2's complement of -82. Firstly, we want to get the binary value of 82.
මෙහිදී -82 හි 2 හි අනුපූරකය ලබා ගැනීමට ප්‍රශ්නය අසයි. පළමුව, අප 82 හි ද්විමය අගය ලබා ගත යුතුයි.

2	82		
2	41	-	0
2	20	-	1
2	10	-	0
2	5	-	0
2	2	-	1
2	1	-	0
	0	-	1

$= 1010010_2$

As we represent the 2's complement value in 8 bits, we should add 0s before the whole value.
අපි බිටු 8 කින් 2 හි අනුපූරක අගය නියෝජනය කරන බැවින් සම්පූර්ණ අගයට පෙර 0 එකතු කළ යුතුය.

$= 01010010_2$

After this, the binary value should be inverted.
මෙයින් පසු, ද්විමය අගය ප්‍රතිලෝම කළ යුතුය.

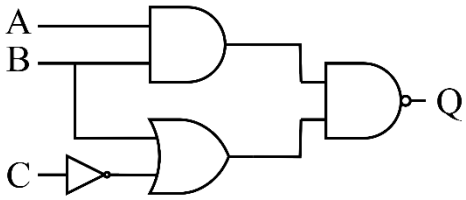
$01010010_2 \rightarrow 10101101_2$

Finally, add 1 to the inverted binary value.
අවසාන වශයෙන්, ප්‍රතිලෝම ද්විමය අගයට 1 ක් එකතු කරන්න.

1	0	1	0	1	1	0	1
						+	1
1	0	1	0	1	1	1	0

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

5. What is the most simplified Boolean expression of the below logic circuit?
පහත තාර්කික පරිපථයේ සරලම බූලියානු ප්‍රකාශනය කුමක්ද?



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

Answer: 1)

Explanation / පැහැදිලි කිරීම:

Step / පියවර 1:
Determine the Boolean expression from the circuit.
පරිපථයෙන් බූලියානු ප්‍රකාශනය තීරණය කරන්න.

First, derive the Boolean expression for the given logic circuit. The expression is:
පළමුව, දී ඇති තාර්කික පරිපථය සඳහා බූලියානු ප්‍රකාශනය ව්‍යුත්පන්න කරන්න. ප්‍රකාශනය වන්නේ:

$$Q = \overline{(A.B).(B + \overline{C})}$$

Step / පියවර 2:
Let's simplify the above Boolean expression / **ඉහත බූලියානු ප්‍රකාශනය සරල කරමු.**

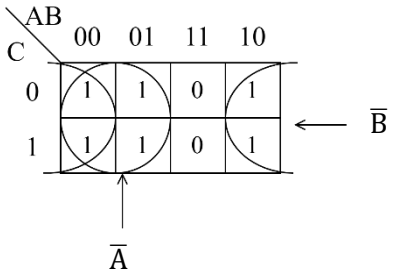
Method / ක්‍රමය 1:
Using Boolean laws / **බූලියානු නීති භාවිතා කිරීම.**

$= \overline{(A.B).(B + \overline{C})}$
$= \overline{(A.B)} + \overline{(B + \overline{C})}$ De Morgan's Law / **ද මුවාර් ප්‍රමේයය**
$= (\overline{A} + \overline{B}) + (\overline{B}.\overline{\overline{C}})$ De Morgan's Law / **ද මුවාර් ප්‍රමේයය**
$= \overline{A} + \overline{B} + \overline{B}.C$ Double complement Law / **ද්විත්ව ප්‍රතිලෝම න්‍යාය**
$= \overline{A} + \overline{B}$ Redundancy Law / **සමතිරික්ත න්‍යාය**

Method / ක්‍රමය 2:
Using Karnaugh map/ **කානෝ සිතියම භාවිතා කිරීම.**

A	B	C	A.B	$B + \overline{C}$	$\overline{(A.B).(B + \overline{C})}$
0	0	0	0	1	1
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	1
1	0	0	0	1	1
1	0	1	0	0	1
1	1	0	1	1	0
1	1	1	1	1	0

Using the values where the output Q = 1 in the truth table, we can now plot them on the Karnaugh map to simplify the Boolean expression.
සත්‍යතා වගුවේ Q = 1 ප්‍රතිදානය ඇති අගයන් භාවිතා කරමින්, බූලියානු ප්‍රකාශනය සරල කිරීම සඳහා අපට ඒවා කානෝ සිතියමේ සටහන් කළ හැකිය.



Both methods above confirm that the simplified Boolean expression is $\overline{A} + \overline{B}$, and we can use either method to get the correct answer.
ඉහත ක්‍රම දෙකම, සරල කළ බූලියානු ප්‍රකාශනය $\overline{A} + \overline{B}$ බව තහවුරු කරන අතර, නිවැරදි පිළිතුර ලබා ගැනීමට අපට ඕනෑම ක්‍රමයක් භාවිතා කළ හැකිය.

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

6. A student is practicing list slicing in Python. Given the list `colors = ["red", "green", "blue", "yellow", "purple"]`, what will be the output of `print(colors[::-2])`?
ශිෂ්‍යයෙක් පයිතන් හි list slicing පුහුණු වෙමින් සිටී. ලැයිස්තුවේ `colors = ["red", "green", "blue", "yellow", "purple"]` ලබා දී ඇති අතර, `print(colors[::-2])` හි ප්‍රතිදානය කුමක් වේවිද?

1) `['purple', 'blue', 'red']`
2) `['red', 'blue', 'purple']`
3) `['purple', 'green']`
4) `['purple', 'yellow', 'green']`
5) `Error`

Answer: 1)

Explanation / පැහැදිලි කිරීම:

To determine the output of `print(colors[::-2])`, we need to break down the slicing operation.
`print(colors[::-2])` ප්‍රතිදානය තීරණය කිරීමට, slicing ක්‍රියාවලිය බිඳ දැමිය යුතුයි.

The notation `[::-2]` means that the list will be reversed (`[::-1]`) and then every second element will be selected in that reversed order. Given the list `colors = ["red", "green", "blue", "yellow", "purple"]`, reversing it would produce `["purple", "yellow", "blue", "green", "red"]`. Taking every second element from this reversed list, starting from `"purple"`, gives us `["purple", "blue", "red"]`.
`[::-2]` අංකනය යන්නෙන් අදහස් වන්නේ ලැයිස්තුව ආපසු හරවනු ලබන බවයි (`[::-1]`) ඉන්පසු සෑම දෙවන අංගයක්ම එම ප්‍රතිලෝම අනුපිළිවෙලින් තෝරා ගනු ලැබේ. ලැයිස්තුවේ `colors = ["red", "green", "blue", "yellow", "purple"]` ලබා දී ඇත. එය ආපසු හැරවීමෙන් `["purple", "yellow", "blue", "green", "red"]` නිපදවනු ඇත. මෙම ප්‍රතිලෝම ලැයිස්තුවෙන් සෑම දෙවන අංගයක්ම ගත් විට, `"purple"` වලින් පටන් ගෙන, අපට `["purple", "blue", "red"]` ලැබේ.

Thus, the output of `print(colors[::-2])` is `['purple', 'blue', 'red']`.
මේ අනුව, `print(colors[::-2])` ප්‍රතිදානය `['purple', 'blue', 'red']` වේ.

So, the correct answer is 1) `['purple', 'blue', 'red']`.
එබැවින්, නිවැරදි පිළිතුර 1) `['purple', 'blue', 'red']` වේ.

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

```
i = -1
sum_val = 0
while i > -10:
    if i % 2 == 0:
        sum_val += i
    else:
        sum_val -= i
    i -= 1
print(sum_val)
```

1) -6 2) 5 3) 5.5 4) -5 5) -7.0

Answer: 2)

Explanation / පැහැදිලි කිරීම:

The given code initializes `i` to `-1` and `sum_val` to `0`, and then enters a while loop that continues as long as `i` is greater than `-10`. Within the loop, if `i` is even, its value is added to `sum_val`, and if `i` is odd, its value is subtracted from `sum_val`. After each iteration, `i` is decremented by `1`. As the loop progresses, the value of `sum_val` alternates between adding and subtracting the values of i based on whether `i` is even or odd. After all iterations, the final value of `sum_val` is `5`, which is printed as the output. Thus, the output of the code is `5`.
ලබා දී ඇති කේතය `i` සිට `-1` දක්වා සහ `sum_val`, `0` දක්වා ආරම්භ කරයි, පසුව `i -10` ට වඩා වැඩි වන තාක් කල් පවතින ලූපයකට ඇතුල් වේ. ලූපය තුළ, `i` ඉරට්ටේ නම්, එහි අගය `sum_val` වෙත එකතු වේ, `i` ඔත්තේ නම්, එහි අගය `sum_val` වෙතින් අඩු කරනු ලැබේ. එක් එක් පුනරාවර්තනයට පසුව, `i` `1` කින් අඩු කරනු ලැබේ. ලූපය ප්‍රගතියත් සමඟ, `sum_val` හි අගය `i` ඉරට්ටේ හෝ ඔත්තේ ද යන්න මත පදනම්ව `i` හි අගයන් එකතු කිරීම සහ අඩු කිරීම අතර ප්‍රත්‍යාවර්ත වේ. සියලුම පුනරාවර්තනයෙන් පසුව, `sum_val` හි අවසාන අගය `5` වේ, එය ප්‍රතිදානය ලෙස මුද්‍රණය වේ. මේ අනුව, කේතයේ ප්‍රතිදානය `5` වේ.

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

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

```
def func(a, b=5, c=10):
    return a + b + c
print(func(3, 4))
```

1) 12 2) 17 3) 22 4) 10 5) Error

Answer: 2)

Explanation / පැහැදිලි කිරීම

The output of the given Python code will be 17. In the function `func(a, b=5, c=10)`, the parameters `b` and `c` have default values of 5 and 10, respectively. When the function is called with `func(3, 4)`, `a` is assigned the value 3, and `b` is assigned the value 4, overriding its default value. Since `c` is not provided in the function call, it retains its default value of 10. Therefore, the function returns the result of 3 + 4 + 10, which equals 17.
ලබා දී ඇති පයිතන් කේතයේ ප්‍රතිදානය 17 වනු ඇත. `func(a, b=5, c=10)` ශ්‍රිතයේ `b` සහ `c` පරාමිති වලට පිළිවෙලින් 5 සහ 10 පෙරනිමි අගයන් ඇත. ශ්‍රිතය `func(3, 4)` සමඟින් හැඳින්වූ විට, `a` ට 3 අගය ද, `b` හට එහි පෙරනිමි අගය අභිබවා 4 අගය ද පවරනු ලැබේ. ශ්‍රිත ඇමතුමෙහි `c` සපයා නොමැති බැවින්, එය එහි පෙරනිමි අගය 10 රඳවා ගනී. එබැවින්, ශ්‍රිතය 17 ට සමාන වන 3 + 4 + 10 හි ප්‍රතිඵලය ලබා දෙයි.

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

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

```
i = 1
total = 0
while i <= 5:
    if i % 2 == 0:
        total += i * 3
    else:
        total += i * 2
    i += 1

print(total)
```

1) 25 2) 22 3) 36 4) 20 5) 18

Answer: 3)

Explanation / පැහැදිලි කිරීම

The output of this Python code is 36. The code initializes i to 1 and total to 0.It then enters a while loop that iterates while i is less than or equal to 5.
මෙම පයිතන් කේතයේ ප්‍රතිදානය 36 වේ. කේතය i 1 සිට 0 දක්වා ආරම්භ කරයි. එය i 5 ට අඩු හෝ සමාන වන විට පුනරාවර්තනය වන while loop එකකට ඇතුල් වේ.

Inside the loop, it checks if i is even (`i % 2 == 0`). If so, it adds `i * 3` to total, otherwise, it adds `i * 2` to total. After the loop completes, total is 36, which is then printed.
ලූපය තුළ, එය i ඉරට්ටේ දැයි පරීක්ෂා කරයි (`i % 2 == 0`). එසේ නම්, එය මුළු එකතුවට `i * 3` එකතු කරයි, එසේ නොමැති නම්, එය මුළු එකතුවට `i * 2` එකතු කරයි. ලූපය සම්පූර්ණ වූ පසු, එකතුව 36 වේ, එය මුද්‍රණය කෙරේ.

For each iteration / එක් එක් පුනරාවර්තනය සඳහා:

- When i = 1 (odd), total += 1 * 2 → total = 2
  i = 1 (ඔත්තේ) විට total += 1 * 2 → total = 2
- When i = 2 (even), total += 2 * 3 → total = 8
  i = 2 (ඉරට්ටේ) විට total += 2 * 3 → total = 8
- When i = 3 (odd), total += 3 * 2 → total = 14
  i = 3 (ඔත්තේ) විට total += 3 * 2 → total = 14
- When i = 4 (even), total += 4 * 3 → total = 26
  i = 4 (ඉරට්ටේ) විට total += 4 * 3 → total = 26
- When i = 5 (odd), total += 5 * 2 → total = 36
  i = 5 (ඔත්තේ) විට total += 5 * 2 → total = 36

Therefore, the correct answer is 3) 36.
එබැවින්, නිවැරදි පිළිතුර 3) 36 වේ.

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

```
a, b = 0, 1
count = 0
while count = 7:
    print(a)
    a, b = b, a + b
    count += 1
```

1)
```
0
1
1
2
3
5
8
```

2)
```
0
1
2
3
4
5
6
```

3)
```
0
1
1
2
2
3
3
```

4)
```
0
1
1
2
3
5
7
```

5) Error

Answer: 5)

Explanation / පැහැදිලි කිරීම

The code will produce a syntax error due to an incorrect assignment in the `while loop` condition. Specifically, `while count = 7:` uses a single equals sign (`=`), which is an assignment operator rather than a comparison operator. In Python, `=` is used to assign values, whereas `==` is used to compare values. Since `while count = 7:` attempts to assign `7` to `count` rather than check if `count` is equal to `7`, Python will raise a syntax error. To correct this, the line should be changed to `while count == 7:` if the goal is to create a loop that only executes when `count` is exactly `7`, or modify it as needed to achieve the intended behavior.
`while loop` තත්වයේ වැරදි පැවරුමක් හේතුවෙන් කේතය කාරක රීති දෝෂයක් ඇති කරයි. නිශ්චිතවම, `while count = 7:` තනි සමාන ලකුණක් (`=`) භාවිතා කරයි, එය සංසන්දන මෙහෙයවනයට වඩා පැවරුම් මෙහෙයවනයක් වේ. Python හි, අගයන් පැවරීමට `=` භාවිතා කරන අතර, අගයන් සංසන්දනය කිරීමට `==` භාවිතා කරයි. `while count = 7:` `count` `7` ට සමාන දැයි පරීක්ෂා කරනවාට වඩා `7` `count` වෙත පැවරීමට උත්සාහ කරන බැවින්, පයිතන් කාරක රීති දෝෂයක් මතු කරයි. මෙය නිවැරදි කිරීම සඳහා, ඉලක්කය වන්නේ `count` හරියටම `7` වන විට පමණක් ක්‍රියාත්මක වන ලූපයක් සෑදීම හෝ අපේක්ෂිත හැසිරීම සාක්ෂාත් කර ගැනීමට අවශ්‍ය පරිදි එය වෙනස් කිරීම නම්, කේත පේළිය `while count == 7:` ලෙස වෙනස් කළ යුතු වේ.

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