සියලු ම හිමිකම් ඇවිරිණි / All Rights Reserved]

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

Information and Communication Technology

තොරතුරු හා සන්නිවේදන තාක්ෂණය

2026 QUIZ NO 298 MARKING

2025/10/04
ictfromabc
Ravindu Bandaranayake

1. Which of the following was a major development in computing during the 1990s?
1990 කාලයේදී පරිගණනයේ ප්‍රධාන වර්ධනයක් වූයේ පහත ඒවායින් කුමක්ද?

1) The invention of the microprocessor / ක්ෂුද්‍ර සකසනය සොයාගැනීම.
2) The creation of the internet / අන්තර්ජාලය නිර්මාණය කිරීම.
3) The development of the personal computer / පුද්ගලික පරිගණකයේ සංවර්ධනය.
4) The widespread adoption of cloud computing / වලාකුළු පරිගණකකරණයේ පුළුල්ව පැතිරීම.
5) The introduction of artificial intelligence / කෘතිම බුද්ධිය හඳුන්වාදීම.

Answer: 2)

Explanation / පැහැදිලි කිරීම:

1) The invention of the microprocessor / ක්ෂුද්‍ර සකසනයේ සොයාගැනීම
This was a significant development in the 1970s, not the 1990s.
මෙය 1990 වේදී නොව 1970 වේදී සැලකිය යුතු වර්ධනයක් විය.

2) The creation of the internet / අන්තර්ජාලය නිර්මාණය කිරීම
The internet, as we know it today, began to take shape in the 1990s with the widespread adoption of the World Wide Web.
අද අප දන්නා පරිදි අන්තර්ජාලය 1990 වේදී ලෝක විසිරි වියමන පුළුල් ලෙස භාවිතා කිරීමත් සමඟ හැඩගැසීමට පටන් ගත්තේය.

3) The development of the personal computer / පුද්ගලික පරිගණකයේ දියුණුව
This was a major development in the 1970s and 1980s.
මෙය 1970 සහ 1980 වේදී විශාල වර්ධනයක් විය.

4) The widespread adoption of cloud computing / වලාකුළු පරිගණකකරණයේ පුලුල්ව පැතිරීම
Cloud computing became more prominent in the 2000s.
2000 දී වලාකුළු පරිගණකකරණය වඩාත් ප්‍රමුඛ විය.

5) The introduction of artificial intelligence / කෘතිම බුද්ධිය හඳුන්වාදීම
While AI research has been ongoing for decades, its widespread practical applications and advancements became more significant in recent years.
AI පර්යේෂණ දශක ගණනාවක් තිස්සේ සිදුවෙමින් පවතින අතර, එහි පුළුල් ප්‍රායෝගික යෙදුම් සහ දියුණුව මෑත වසරවලදී වඩාත් වැදගත් විය.

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

2. Which of the following was the first commercially available personal computer?
පහත සඳහන් ඒවායින් පළමු වාණිජමය වශයෙන් ලබා ගත හැකි පුද්ගලික පරිගණකය කුමක්ද?

1) Apple II 2) IBM PC 3) Commodore 64 4) TRS-80 5) Altair 8800

Answer: 5)

Explanation of each option / එක් එක් විකල්පය පැහැදිලි කිරීම:

1) **Apple II**
   A popular personal computer released in 1977, but not the first.
   1977 දී නිකුත් කරන ලද ජනප්‍රිය පුද්ගලික පරිගණකයක්, නමුත් පළමු එක නොවේ.

2) **IBM PC**
   Released in 1981, it became a standard for personal computers, but it was not the first.
   1981 දී නිකුත් කරන ලද එය පුද්ගලික පරිගණක සඳහා සම්මතයක් බවට පත් වූ නමුත් එය පළමුවැන්න නොවේ.

3) **Commodore 64**
   A popular home computer released in 1982, but not the first.
   1982 දී නිකුත් කරන ලද ජනප්‍රිය ගෘහ පරිගණකයක්, නමුත් පළමු එක නොවේ.

4) **TRS-80**
   Released in 1977, it was a popular early personal computer, but not the first.
   1977 දී නිකුත් කරන ලද, එය ජනප්‍රිය මුල් පුද්ගලික පරිගණකයක් වූ නමුත් පළමු එක නොවේ.

5) **Altair 8800**
   Released in 1975, it is widely considered the first commercially available personal computer. It was a kit that required assembly, but it sparked the personal computer revolution.
   1975 දී නිකුත් කරන ලද එය වාණිජමය වශයෙන් ලබා ගත හැකි පළමු පුද්ගලික පරිගණකය ලෙස පුළුල් ලෙස සැලකේ. එය එකලස් කිරීමට අවශ්‍ය කට්ටලයක් වූ නමුත් එය පුද්ගලික පරිගණක විප්ලවයක් ඇති කළේය.

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

3. Which of the following answer represents bitwise XNOR operation of the two binary numbers 11001011_2 and 11101010_2?
11001011_2 සහ 11101010_2 යන ද්විමය සංඛ්‍යා දෙකෙහි බිටු අනුසාරිත XNOR ක්‍රියාකාරීත්වය නියෝජනය කරන්නේ පහත ඒවායින් කුමන පිළිතුරද?

1) 11011110_2 2) 10011110_2 3) 11100010_2 4) 10111011_2 5) 11011111_2

Answer: 4)

Truth table regarding the XNOR operation. (Consider A and B are two inputs while F is the output)
XNOR මෙහෙයුම සම්බන්ධයෙන් සත්‍යතා වගුව. (A සහ B යනු ආදාන දෙකක් වන අතර F යනු ප්‍රතිදානය බව සලකන්න)

ICN	A	B	F
1	0	0	1
2	0	1	0
3	1	0	0
4	1	1	1

Applying XNOR operation for given two binary numbers.
ලබා දී ඇති ද්විමය සංඛ්‍යා දෙක සඳහා XNOR මෙහෙයුම යෙදීම.

```
        1  1  0  0  1  0  1  1
XNOR
        1  1  1  0  1  0  1  0
      ________________________
        1  1  0  1  1  1  1  0
```

Therefore, the correct answer is 11011110_2.
එමනිසා, නිවැරදි පිළිතුර වන්නේ 11011110_2 යන්නයි.

4. What is the result of the following boolean expression when trivialized?
පහත දී ඇති බූලියානු ප්‍රකාශනය සුළු කල විට ලැබෙන ප්‍රතිඵලය කුමක්ද?

$F = \overline{XZ}(\overline{X} + Z)(Y + \overline{Y})$

1) $\underline{\overline{X}}$ 2) $\overline{Y}$ 3) X 4) Y 5) XY

Answer: 1)

Explanation / පැහැදිලි කිරීම

$\overline{XZ}(\overline{X} + Z)(Y + \overline{Y})$	
$(\overline{X} + \overline{Z})(\overline{X} + Z)(Y + \overline{Y})$	De Morgan's theory / ද මුවාර් ප්‍රමේයය
$(\overline{X} + \overline{Z})(\overline{X} + Z).1$	Complement law / ප්‍රතිලෝම න්‍යාය
$(\overline{X} + \overline{Z})(\overline{X} + Z)$	Identity law / සර්වසාම්‍ය න්‍යාය
$(\overline{X} + Z)\overline{X} + (\overline{X} + Z)\overline{Z}$	Distribution law / විඝටන න්‍යාය
$\overline{X}\overline{X} + \overline{X}Z + (\overline{X} + Z)\overline{Z}$	Distribution law / විඝටන න්‍යාය
$\overline{X} + \overline{X}Z + (\overline{X} + Z)\overline{Z}$	Idempotent law / තදේවභාවී න්‍යාය
$\overline{X} + (\overline{X} + Z)\overline{Z}$	Redundancy law / සමතිරික්ත න්‍යාය
$\overline{X} + \overline{Z}\overline{X} + \overline{Z}Z$	Distribution law / විඝටන න්‍යාය
$\overline{X} + \overline{Z}\overline{X} + 0$	Complement law / ප්‍රතිලෝම න්‍යාය
$\overline{X} + \overline{Z}\overline{X}$	Identity law / සර්වසාම්‍ය න්‍යාය
$\overline{X}$	Redundancy law / සමතිරික්ත න්‍යාය

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

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

If the value of C in the above Boolean expression is 1, what is the simplified answer?
ඉහත බූලිය ප්‍රකාශනයේ C හි අගය 1 නම් සුළු කළ පසු පිළිතුර කුමක්ද?

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

Answer: 2)

Explanation / පැහැදිලි කිරීම

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

First, let's get the relevant expression by applying 1 to the places where C is present.
පළමුව මෙහි C පවතින ස්ථාන වලට 1 යොදා අදාළ ප්‍රකාශනය ලබාගනිමු.

$1.B + A((\overline{D.1} + 1.1)D + B.1)$

Now this expression needs to be simplified using Boolean laws.
දැන් මෙම ප්‍රකාශනය බූලියානු නිති යොදාගනිමින් සුළුකර ගත යුතුයි.

$1.B + A((\overline{D.1} + 1.1)D + B.1)$	
$B + A((\overline{D.1} + 1)D + B)$	Identity law / සර්වසාම්‍ය න්‍යාය
$B + A((1)D + B)$	Identity law / සර්වසාම්‍ය න්‍යාය
$B + A(D + B)$	Identity law / සර්වසාම්‍ය න්‍යාය
$B + A.D + A.B$	Distributive law / විඝටන න්‍යාය
$A.B + B + A.D$	Commutative law / න්‍යාදේශ න්‍යාය
$B + A.D$	Redundancy law / සමතිරික්ත න්‍යාය

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

6. Consider the following python code.
පහත පයිතන් කේතය සලකා බලන්න.

```
def func(x=1,y=2):
    x=x+y
    y+=1
    print(x,y)

func(y=2,x=1)
```

What is the output of the above code?
ඉහත කේතයේ ප්‍රතිදානය කුමක්ද?

1) `1 3` 2) `3 2` 3) `2 3` 4) `Error` 5) `3 3`

Answer: 5)

Explanation / පැහැදිලි කිරීම

The function `func` takes two optional parameters x and y , with default values of 1 and 2, respectively. Inside the function, it adds y to x and then increments y by 1. It then prints the updated values of x and y. When calling `func(y=2,x=1)` , the parameters are explicitly set to `x=1` and `y=2`. Thus, within the function, x becomes 1 + 2 which equals 3, and y is incremented from 2 to 3. Therefore, the output of the function will be `3 3`.
ශ්‍රිතය 1 සහ 2 පෙරනිමි අගයන් සමඟින්, x සහ y විකල්ප පරාමිති දෙකක් ගනී. ශ්‍රිතය තුළ, එය x වෙත y එකතු කරන අතර පසුව y 1 කින් වැඩි කරයි. එය x සහ y හි යාවත්කාලීන අගයන් මුද්‍රණය කරයි. `func(y=2,x=1)` ඇමතීමේ දී පරාමිති පැහැදිලිව `x=1` සහ `y=2` ලෙස සකසා ඇත. මේ අනුව, ශ්‍රිතය තුළ, x 1 + 2 වන අතර එය 3 ට සමාන වන අතර, y, 2හි සිට 3 දක්වා වැඩි වේ. එබැවින්, ශ්‍රිතයේ ප්‍රතිදානය `3 3` වනු ඇත.

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

7. How do you open a file "data.txt" in read mode?
"data.txt" ගොනුවක් කියවීම් මාදිලියෙන් විවෘත කරන්නේ කෙසේද?

1) `file = open("data.txt", "r")`
2) `file = open("data.txt", "w")`
3) `file = read("data.txt")`
4) `file = open("data.txt")`
5) `file = load("data.txt")`

Answer: 1)

Explanation / පැහැදිලි කිරීම

The correct answer is 1, as it shows the proper syntax to open a file named "data.txt" in read mode using Python. The open() function is a built-in function in Python that is used to open files, and it takes two arguments: the file name (as a string) and the mode in which the file should be opened. In this case, the file name is "data.txt", and the mode is "r", which stands for read mode. This mode allows the file to be opened for reading, and it does not modify the file's contents.
Python භාවිතයෙන් "data.txt" නම් ගොනුවක් කියවීමේ ප්‍රකාරයේදී විවෘත කිරීමට සුදුසු වාක්‍ය ඛණ්ඩය පෙන්වන බැවින් නිවැරදි පිළිතුර 1) වේ. open() ශ්‍රිතය යනු ගොනු විවෘත කිරීමට භාවිතා කරන Python හි ගොඩනඟන ලද ශ්‍රිතයක් වන අතර එයට තර්ක දෙකක් අවශ්‍ය වේ: ගොනුවේ නම string එකක් ලෙස) සහ ගොනුව විවෘත කළ යුතු ආකාරය. මෙම අවස්ථාවෙහිදී, ගොනු නාමය "data.txt" වන අතර, මාදිලිය "r" වේ, එය කියවීමේ මාදිලිය වේ. මෙම මාදිලිය කියවීම සඳහා ගොනුව විවෘත කිරීමට ඉඩ සලසයි, එය ගොනුවේ අන්තර්ගතය වෙනස් නොකරයි.

The other options are incorrect because they do not meet the requirements for opening the file in read mode. Option 2 ("w") opens the file in write mode, which can overwrite the file's contents. Option 3 uses an invalid function read() that does not exist for opening files. Option 4 lacks the "r" mode, which defaults to read mode but is not explicit. Option 5 uses an undefined load() function. Thus, only option 1 correctly opens the file in read mode, ensuring that its contents can be safely accessed without any risk of alteration.
කියවීමේ මාදිලියේ ගොනුව විවෘත කිරීම සඳහා අවශ්‍යතා සපුරාලන්නේ නැති නිසා අනෙකුත් විකල්ප වැරදිය. විකල්ප 2 ("w") ගොනුව ලිවීමේ ආකාරයෙන් විවෘත කරයි, එමඟින් ගොනුවේ අන්තර්ගතය උඩින් ලිවිය හැක. විකල්ප 3 ගොනු විවෘත කිරීම සඳහා නොපවතින වලංගු නොවන read() භාවිතා කරයි. විකල්ප 4 හි "r" මාදිලිය නොමැත, එය කියවීමට පෙරනිමි වන නමුත් පැහැදිලි නොවේ. විකල්ප 5 නිර්වචනය නොකළ load() ශ්‍රිතයක් භාවිතා කරයි. මේ අනුව, 1 විකල්පය පමණක් කියවීමේ මාදිලියේ ගොනුව නිවැරදිව විවෘත කරයි, එහි අන්තර්ගතය වෙනස් කිරීමේ අවදානමකින් තොරව ආරක්ෂිතව ප්‍රවේශ විය හැකි බව සහතික කරයි.

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

8. Consider the following statements about specialties of python programming language.
පයිතන් ක්‍රමලේඛන භාෂාවේ විශේෂතා පිළිබඳ පහත ප්‍රකාශයන් සලකා බලන්න.

A. Ability of interactive mode programming.
අන්තර්ක්‍රියාකාරී මාදිලියේ වැඩසටහන් කිරීමේ හැකියාව.

B. Availability of many numbers of reserved words.
මූල පද බොහෝ ගණනක් තිබීම.

C. Portability.
සුවහනීයතාවය.

Which of the above statement(s) is/are true?
ඉහත ප්‍රකාශවලින් කවරක් සත්‍ය ද?

1) A only. A පමණි.
2) B only. B පමණි.
3) A and B only. A හා B පමණි.
4) A and C only. A හා C පමණි.
5) B and C only. B හා C පමණි.

Answer: 4)

Explanation / පැහැදිලි කිරීම:

Statement / ප්‍රකාශය A
This statement is True. Python supports an interactive mode, allowing you to execute code directly in the interpreter. This is invaluable for testing code snippets, learning the language, and debugging.
මෙම ප්‍රකාශය සත්‍ය වේ. Python අන්තර්ක්‍රියාකාරී මාදිලියකට සහය දක්වයි, ඔබට කේත පරිවර්තකය තුළ කෙලින්ම ක්‍රියාත්මක කිරීමට ඉඩ සලසයි. කේත කොටස් පරීක්ෂා කිරීම, භාෂාව ඉගෙනීම සහ දෝෂහරණය කිරීම සඳහා මෙය ඉතා අගනේය.

Statement / ප්‍රකාශය B
This statement is False. Python has a relatively small set of reserved keywords. This simplicity contributes to its readability and ease of learning.
මෙම ප්‍රකාශය අසත්‍යයකි. Python සතුව සාපේක්ෂව කුඩා වෙන් කර ඇති මූල පද කට්ටලයක් ඇත. මෙම සරල බව එහි කියවීමේ හැකියාව සහ ඉගෙනීමේ පහසුව සඳහා දායක වේ.

Statement / ප්‍රකාශය C
This statement is True. Python is highly portable, meaning you can write Python code on one platform (like Windows) and run it on another (like macOS or Linux) with minimal or no modifications.
මෙම ප්‍රකාශය සත්‍යයකි. පයිතන් ඉතා අතේ ගෙන යා හැකි ය, එනම් ඔබට එක් වේදිකාවක (වින්ඩෝස් වැනි) පයිතන් කේතය ලිවිය හැකි අතර අවම හෝ වෙනස් කිරීම් නොමැතිව තවත් (මැකෝස් හෝ ලිනක්ස් වැනි) එය ධාවනය කළ හැකිය.

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

9. If the user inputs 20242526, what will the output be?
පරිශීලකයා 20242526 ඇතුළත් කළහොත්, ප්‍රතිදානය කුමක් වේවිද?

```
num = int(input("Enter a number: "))
reversed_num = int(str(num)[::-1])
print("Reversed number:", reversed_num)
```

1) Reversed number: 62524202
2) Reversed num: 62524202524
3) Reversed number: 62524524
4) Reversed number: 62202524
5) Reversed number 62524202

Answer: 1)

Explanation / පැහැදිලි කිරීම:

The correct answer is 1) Reversed number: 62524202. In this code, the user inputs the number 20242526.
නිවැරදි පිළිතුර වන්නේ 1) Reversed number: 62524202 වේ. මෙම කේතයෙහි, පරිශීලකයා 20242526 අංකය ඇතුළත් කරයි.

The code converts this integer into a string and then reverses the string using slicing [::-1], which effectively rearranges the digits in reverse order. After reversing, it converts the result back into an integer, so any leading zeros are removed.
කේතය මෙම පූර්ණ සංඛ්‍යාව string එකක් බවට පරිවර්තනය කර, පසුව string එක ප්‍රතිලෝම කර [::-1] slicing භාවිතා කරයි, එමඟින් ඉලක්කම් ප්‍රතිලෝම අනුපිළිවෙලින් ප්‍රතිසංවිධානය කරයි. ආපසු හැරවීමෙන් පසු, එය ප්‍රතිඵලය නැවත පූර්ණ සංඛ්‍යාවක් බවට පරිවර්තනය කරයි, එබැවින් ඕනෑම ප්‍රමුඛ ශුන්‍යයන් ඉවත් කරනු ලැබේ.

The output is 62524202 because reversing the input 20242526 results in this exact sequence of digits. Other options are incorrect because they do not match the exact reversed order of the input, nor do they correctly represent the reversed integer without extra or missing digits.
ප්‍රතිදානය 62524202 වන්නේ ආදානය 20242526 ආපසු හැරවීමෙන් මෙම නියම ඉලක්කම් අනුක්‍රමය ලැබෙන බැවිනි. වෙනත් විකල්ප වැරදි වන්නේ ඒවා ආදානයේ නිශ්චිත ප්‍රතිලෝම අනුපිළිවෙලට නොගැළපෙන නිසා හෝ අමතර හෝ නැතිවූ ඉලක්කම් නොමැතිව ප්‍රතිලෝම පූර්ණ සංඛ්‍යාව නිවැරදිව නියෝජනය නොකරන බැවිනි.

10. Which of the following is the most efficient way to reverse a string in Python?
පයිතන් හි string එකක් ප්‍රතිවර්තනය කිරීමේ වඩාත් කාර්යක්ෂම ක්‍රමය පහත ඒවායින් කුමක්ද?

1) Using a loop to iterate through the string and append characters to a new string in reverse order.
String එක හරහා පුනරාවර්තනය කිරීමට සහ ප්‍රතිලෝම අනුපිළිවෙලට නව string එකකට අක්ෂර එකතු කිරීමට ලූපයක් භාවිතා කිරීම.
2) Using the reversed() function and converting the result to a string.
reversed() ශ්‍රිතය භාවිතා කිරීම සහ ප්‍රතිඵලය string එකකට පරිවර්තනය කිරීම.
3) Using slice notation with a step of -1.
-1 step අගයක් සහිත slice අංකනය භාවිතා කිරීම.
4) Using the join() method with a reversed list of characters.
ප්‍රතිලෝම අක්ෂර ලැයිස්තුවක් සමඟ join() ක්‍රමය භාවිතා කිරීම.
5) Using a recursive function.
ආවර්තික ශ්‍රිතයක් භාවිතා කිරීම.

Answer: 3)

Here's a brief explanation of each option
මෙන්න එක් එක් විකල්පය පිළිබඳ කෙටි පැහැදිලි කිරීමක්

Option / විකල්ප 1)
This method involves manually iterating through the string and constructing a new string character by character. While straightforward, it is inefficient due to the repeated creation of new strings, which can lead to increased time complexity, especially for longer strings.
මෙම ක්‍රමයට string එක හරහා අතින් පුනරාවර්තනය කිරීම සහ අක්ෂරයෙන් නව string අක්ෂරයක් ගොඩනැගීම ඇතුළත් වේ. සරල වුවත්, නව string නැවත නැවත නිර්මාණය කිරීම හේතුවෙන් එය අකාර්යක්ෂම වේ, එය කාල සංකීර්ණතාව වැඩි කිරීමට හේතු විය හැක, විශේෂයෙන් දිගු string සඳහා.

Option / විකල්ප 2)
The reversed() function creates an iterator that yields characters from the original string in reverse order. However, it requires an additional step to convert this iterator back into a string using .join(), making it slightly less efficient than direct slicing.
reversed() ශ්‍රිතය මඟින් ප්‍රතිලෝම අනුපිළිවෙලින් මුල් string එකෙන් අක්ෂර ලබා දෙන පුනරාවර්තකයක් නිර්මාණය කරයි. කෙසේ වෙතත්, .join() භාවිතයෙන් මෙම පුනරාවර්තකය නැවත string එකක් බවට පරිවර්තනය කිරීමට අමතර පියවරක් අවශ්‍ය වේ, එය සෘජු slicing වඩා තරමක් අඩු කාර්යක්ෂම වේ.

Option / විකල්ප 3)
This is the most efficient method, as it allows for reversing a string in a single line of code (reversed_string = original_string[::-1]). It leverages Python's built-in capabilities and is both concise and fast, making it the preferred approach for reversing strings.
මෙය වඩාත් කාර්යක්ෂම ක්‍රමය වන අතර, එය තනි කේත පේළියක (reversed_string = original_string[::1]) string එකක් ආපසු හැරවීමට ඉඩ සලසයි. එය පයිතන් හි ගොඩනඟන ලද හැකියාවන් උපයෝගී කර ගන්නා අතර එය සංක්ෂිප්ත සහ වේගවත් වන අතර, එය නූල් ප්‍රතිවර්තනය කිරීම සඳහා වඩාත් කැමති ප්‍රවේශය බවට පත් කරයි.

Option / විකල්ප 4)
This method involves first converting the string into a list of characters, reversing that list, and then joining it back into a string. While effective, it is less efficient than slicing because it involves multiple operations (conversion to list, reversal, and joining).
මෙම ක්‍රමයට මුලින්ම string එක අක්ෂර ලැයිස්තුවක් බවට පරිවර්තනය කිරීම, එම ලැයිස්තුව ආපසු හැරවීම සහ එය නැවත string එකක් බවට සම්බන්ධ කිරීම ඇතුළත් වේ. ඵලදායී වන අතර, එය බහු මෙහෙයුම් (ලැයිස්තුවට පරිවර්තනය කිරීම, ආපසු හැරවීම සහ සම්බන්ධ වීම) ඇතුළත් වන නිසා එය slicing වලට වඩා අඩු කාර්යක්ෂම වේ.

Option / විකල්ප 5)
This method defines a function that calls itself to reverse the string. While it can work for small strings, recursion can lead to high memory usage and stack overflow errors for larger strings due to the depth of recursive calls, making it inefficient compared to other methods.
මෙම ක්‍රමය මඟින් string ප්‍රතිලෝම කිරීමට තමන්ව කැඳවන ශ්‍රිතයක් නිර්වචනය කරයි. එය කුඩා string සඳහා ක්‍රියා කළ හැකි අතර, ප්‍රත්‍යාවර්තනය ඉහළ මතක භාවිතයට හේතු විය හැකි අතර ප්‍රත්‍යාවර්ති ඇමතුම්වල ගැඹුර හේතුවෙන් විශාල string සඳහා overflow දෝෂ ඇති විය හැකි අතර එය අනෙකුත් ්‍රමවලට සාපේක්ෂව අකාර්යක්ෂම වේ.

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