සියලු ම හිමිකම් ඇවිරිණි /All Rights Reserved]

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

Information and Communication Technology

තොරතුරු හා සන්නිවේදන තාක්ෂණය

2026 QUIZ NO 416 MARKING

2026/01/30
ictfromabc
Ravindu Bandaranayake

1. Why is the use of strong and unique passwords crucial in protecting against cyber attacks?
සයිබර් ප්‍රහාර වලින් ආරක්ෂා වීමේදී ශක්තිමත් සහ අනන්‍ය මුරපද භාවිතය තීරණාත්මක වන්නේ ඇයි?

1) Strong passwords prevent opening emails containing cyberattacks.
ශක්තිමත් මුරපද සයිබර් ප්‍රහාර සහිත emails විවෘත කිරීම වළක්වයි.

2) Strong passwords help reduce the risk of account compromise if cyber attack occurs.
ප්‍රබල මුරපද, සයිබර් ප්‍රහාර සිදු වුවහොත් ගිණුම් සම්මුතියේ අවදානම අඩු කිරීමට උපකාරී වේ.

3) Strong passwords ensure automatic detection of websites with cyber attacks.
ශක්තිමත් මුරපද සයිබර් ප්‍රහාර සහිත වෙබ් අඩවි ස්වයංක්‍රියව අනාවරණය කර ගැනීම සහතික කරයි.

4) Unique passwords guarantee that no one will attempt cyber attacks.
කිසිවෙකු සයිබර් ප්‍රහාර වලට උත්සාහ නොකරන බවට අනන්‍ය මුරපද සහතික කරයි.

5) Strong passwords encrypt sensitive data directly, making cyber attacks impossible.
ශක්තිමත් මුරපද සෘජුවම සංවේදී දත්ත ගුප්ත කේතනය කරයි, සයිබර් ප්‍රහාර කළ නොහැක.

Answer : 2)

Explanation / පැහැදිලි කිරීම

The use of strong and unique passwords is crucial in protecting against cyberattacks because they help reduce the risk of account compromise if a cyberattack occurs. Strong passwords are designed to be complex and difficult to guess, which makes it harder for attackers to gain unauthorized access to accounts or systems. This added layer of protection reduces the likelihood of attackers successfully exploiting weak credentials to breach sensitive information.
ශක්තිමත් සහ අද්විතීය මුරපද භාවිතය සයිබර් ප්‍රහාරයන්ගෙන් ආරක්ෂා වීම සඳහා තීරණාත්මක වේ, මන්ද ඒවා සයිබර් ප්‍රහාරයක් සිදුවුවහොත් ගිණුම් සම්මුතියේ අවදානම අඩු කිරීමට උපකාරී වේ. ශක්තිමත් මුරපද සංකීර්ණ සහ අනුමාන කිරීමට අපහසු වන පරිදි නිර්මාණය කර ඇති අතර, ප්‍රහාරකයන්ට ගිණුම් හෝ පද්ධති වෙත අනවසර ප්‍රවේශය ලබා ගැනීමට අපහසු වේ. මෙම එකතු කරන ලද ආරක්ෂණ ස්ථරය ප්‍රහාරකයින් සංවේදී තොරතුරු කඩ කිරීම සඳහා දුර්වල අක්තපත්‍ර සාර්ථකව භාවිතා කිරීමේ සම්භාවිතාව අඩු කරයි.

Additionally, unique passwords ensure that even if one account is compromised, other accounts remain secure. Reusing passwords across multiple accounts can amplify the damage of a cyberattack. By combining strength and uniqueness, passwords act as a fundamental defense mechanism, making it more challenging for attackers to infiltrate systems or access critical data. Thus, they play a key role in minimizing the risks associated with cyberattacks.
අතිරේකව, අද්විතීය මුරපද මගින් එක් ගිණුමක් අවදානමට ලක්වුවද, අනෙකුත් ගිණුම් සුරක්ෂිතව පවතින බව සහතික කරයි. බහු ගිණුම් හරහා මුරපද නැවත භාවිතා කිරීමෙන් සයිබර් ප්‍රහාරයක හානිය වැඩි කළ හැක. ශක්තිය සහ සුවිශේෂත්වය ඒකාබද්ධ කිරීමෙන්, මුරපද මූලික ආරක්ෂක යාන්ත්‍රණයක් ලෙස ක්‍රියා කරයි, ප්‍රහාරකයන්ට පද්ධතිවලට රිංගා ගැනීම හෝ තීරණාත්මක දත්ත වෙත ප්‍රවේශ වීම වඩාත් අභියෝගාත්මක කරයි. මේ අනුව, ඔවුන් සයිබර් ප්‍රහාර හා සම්බන්ධ අවදානම් අවම කිරීම සඳහා ප්‍රධාන කාර්යභාරයක් ඉටු කරයි.

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

2. Two's complement of -56 is the correct answer?
-56 හි දෙකෙහි අනුපූරකය නිවැරදිව දක්වා ඇති පිළිතුර වන්නේ?

1) 11001110_2 2) 11001000_2 3) 10111101_2 4) 11111100_2 5) 10110010_2

Answer : 2)

- First convert the decimal number to a binary number.
දශමය සංඛ්‍යාව පලමුව ද්වීමය සංඛ්‍යාවක් බවට පරිවර්ථනය කරන්න.

-56 = 111000_2

- Add 0 to the beginning until the binary number is 8 digits.
ද්වීමය සංඛ්‍යාව සංඛ්‍යාංක 8ක් වන තුරු මුලට 0 එක් කරන්න.

00111000_2

- Convert 1's to 0's and 0's to 1's since the number is negative.
සංඛයාව සෘණ බැවින් බිටු 8කින් සමන්විත සංඛයාව 1, 0 බවටත් 0, 1 බවටත් පරිවර්ථනය කරන්න.

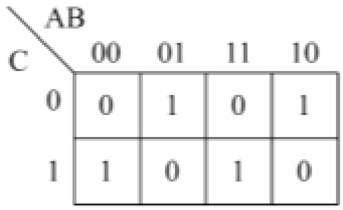
00111000_2
11000111_2 1's

- Adding 1 to the 1's complement represents the 2's complement.
1හි අනුපූරක නිරූපණයට 1ක් එකතු කල විට 2හි අනුපූරකය නිරූපනය වේ.

11000111_2 1's
1 2's
11001000_2

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

3. The following Karnaugh diagram represents the operation of which logic gate?
පහත දැක්වෙන කානෝ සිතියමෙන් නිරූපණය වන්නේ කුමන තාර්කික ද්වාරයේ ක්‍රියාකාරීත්වයද?



1) AND gate 2) OR gate 3) XOR gate 4) NAND gate 5) NOR gate

Answer : 3)

First let's draw the truth table for this Karnaugh map.
පළමුව මෙම කානෝ සිතියමට අදාළ සත්‍යතා වගුව ඇඳ ගනිමු.

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

The outputs of this truth table is 1 only when the number of truth inputs is an odd number. Or the output remains 0 when the number of truth inputs is even. That is, this represents a three-input XOR gate.
මෙම සත්‍යතා වගුවේ ප්‍රතිදාන 1 වී ඇත්තේ සත්‍ය ආදාන ගණන ඔත්තේ සංඛ්‍යාවක් වන විට පමණි. එසේත් නැතිනම් සත්‍ය ආදාන ගණන ඉරට්ටේ වන විට ප්‍රතිදාන 0 වී පවතී. එනම් මෙමගින් ආදාන තුනේ XOR ද්වාරයක් නිරූපණය වේ.

So, the correct answer number is 3.
එබැවින්, නිවැරදි පිළිතුරු අංකය 3 වේ.

4. What does the operating system use to run an application that requires more space than the maximum memory space of a computer's primary memory?
පරිගණකයක ප්‍රාථමික මතකයේ උපරිම මතක අවකාශයට වඩා වැඩි අවකාශයක් අවශ්‍ය වන යෙදුමක් ධාවනය කිරීම සඳහා මෙහෙයුම් පද්ධතිය කුමක් උපයෝගි කර ගනිද?

1) Extended memory / විස්තෘත මතකය.
2) Read only memory / පඨන මාත්‍ර මතකය.
3) Virtual memory / අතථ්‍ය මතකය.
4) Random Access Memory / සසම්භාවි ප්‍රවේශ මතකය.
5) Cache memory / නිහිත මතකය.

Answer : 3)

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

- **Virtual Memory / අතථ්‍යරූපී මතකය**
  Virtual memory is a memory management technique used by operating systems to extend the apparent available memory beyond the physical limits of the computer's primary memory (RAM). It does this by using a portion of the hard drive or SSD as if it were additional RAM. When an application requires more memory than is physically available, parts of the data are temporarily moved to this disk space, allowing the system to handle larger workloads than the physical RAM would allow.
  අතථ්‍ය මතකය යනු පරිගණකයේ ප්‍රාථමික මතකයේ (RAM) භෞතික සීමාවෙන් ඔබ්බට පවතින දෘශ්‍යමාන මතකය දිගු කිරීමට මෙහෙයුම් පද්ධති විසින් භාවිතා කරන මතක කළමනාකරණ ක්‍රමයකි. එය අතිරේක RAM එකක් ලෙස දෘඪ තැටියේ හෝ SSD කොටසක් භාවිතා කිරීමෙන් මෙය සිදු කරයි. යෙදුමකට භෞතිකව ලබා ගත හැකි ප්‍රමාණයට වඩා වැඩි මතකයක් අවශ්‍ය වූ විට, දත්තවල කොටස් තාවකාලිකව මෙම තැටි අවකාශය වෙත ගෙන යන අතර, භෞතික මතකයේ ඉඩ ප්‍රමාණයට වඩා විශාල ක්‍රියාවලියක් හැසිරවීමට පද්ධතියට ඉඩ සලසයි.

- **Extended memory / විස්තෘත මතකය**
  This term was more relevant in older computing systems, referring to memory beyond the first 1MB in early PCs. It is not a technique for managing memory beyond the capacity of physical RAM in modern systems.
  මෙම යෙදුම පැරණි පරිගණක පද්ධතිවල වඩාත් අදාළ වූ අතර, මුල් පරිගණකවල පළමු 1MB ඉක්මවා ඇති මතකය ගැන සඳහන් කරයි. නවීන පද්ධතිවල භෞතික මතකයේ ධාරිතාවෙන් ඔබ්බට මතකය කළමනාකරණය කිරීමේ තාක්ෂණයක් නොවේ.

- **Reading Only Memory (ROM) / පඨන මාත්‍ර මතකය**
  ROM is non-volatile memory used to store firmware or permanent software. It is not involved in running applications or extending memory capacity.
  ROM යනු ස්ථිරාංග හෝ ස්ථිර මෘදුකාංග ගබඩා කිරීමට භාවිතා කරන නශ්‍ය නොවන මතකයකි. එය යෙදුම් ධාවනයට හෝ මතක ධාරිතාව වැඩි කිරීමට සම්බන්ධ නොවේ.

- **Random Access Memory (RAM) / සසම්භාවි ප්‍රවේශ මතකය**
  RAM is the physical memory in a computer where applications and data currently being used are stored. It does not refer to a technique for extending memory beyond physical limits.
  RAM යනු දැනට භාවිතා කරන යෙදුම් සහ දත්ත ගබඩා කර ඇති පරිගණකයක භෞතික මතකයයි. එය භෞතික සීමාවෙන් ඔබ්බට මතකය දිගු කිරීමේ තාක්ෂණයක් වෙත යොමු නොවේ.

- **Cache memory / නිහිත මතකය**
  Cache memory is a small, fast type of volatile memory used to store frequently accessed data for quick retrieval. It speeds up processes but does not extend the memory capacity.
  නිහිත මතකය යනු ඉක්මනින් ලබා ගැනීම සඳහා නිතර ප්‍රවේශ වන දත්ත ගබඩා කිරීමට භාවිතා කරන කුඩා, වේගවත් නශ්‍ය මතක වර්ගයකි. එය ක්‍රියාවලි වේගවත් කරන නමුත් මතක ධාරිතාව දිගු නොකරයි.

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

5. Which of the following is a primary goal of Time-Sharing Operating Systems compared to Multi-Programmed Batch Systems?
බහුක්‍රමලේඛිත කාණ්ඩ පද්ධති හා සසඳන විට කාල විභජන මෙහෙයුම් පද්ධතිවල මූලික ඉලක්කය වන්නේ පහත ඒවායින් කුමක්ද?

1) Maximizing CPU utilization by keeping multiple jobs in memory, switching between them efficiently.
බහු කාර්යය මතකයේ තබාගෙන ඒවා අතර කාර්යක්ෂමව මාරු වීමෙන් CPU භාවිතය උපරිම කිරීම.

2) Providing an interactive environment where users can directly interact with the system and receive immediate feedback.
පරිශීලකයින්ට සෘජුවම පද්ධතිය සමග අන්තර්ක්‍රියා කළ හැකි සහ ක්ෂණික ප්‍රතිපෝෂණ ලබා ගත හැකි අන්තර්ක්‍රියාකාරී පරිසරයක් සැපයීම.

3) Minimizing the turnaround time for batch jobs by executing them sequentially without any user intervention. කිසිදු පරිශීලක මැදිහත්වීමකින් තොරව අනුක්‍රමිකව ක්‍රියාත්මක කිරීමෙන් කණ්ඩායම් කාර්යයන් සඳහා හැරවුම් කාලය අවම කිරීම.

4) Reducing the number of jobs in the system to avoid context switching and overhead.
සන්දර්භය මාරු කිරීම සහ බාධා වළක්වා ගැනීම සඳහා පද්ධතියේ කාර්යයන් ගණන අඩු කිරීම.

5) Ensuring that the system does not have any idle time by using batch processing techniques.
කණ්ඩායම් සැකසුම් ශිල්පීය ක්‍රම භාවිතා කිරීමෙන් පද්ධතියට නිෂ්ක්‍රිය කාලයක් නොමැති බව සහතික කිරීම.

Answer : 2)

Time-Sharing Operating Systems are designed to allow multiple users to interact with the system simultaneously by providing each user with a small time slice of the CPU. The goal is to provide an interactive environment where users get immediate feedback. This type of system is ideal for environments where many users are working on the system at the same time (like in a university or office).
කාල-විභාජන මෙහෙයුම් පද්ධති නිර්මාණය කර ඇත්තේ එක් එක් පරිශීලකයාට CPU හි කුඩා කාල කොටසක් ලබා දීමෙන් බහු පරිශීලකයින්ට එකවර පද්ධතිය සමග අන්තර් ක්‍රියා කිරීමට ඉඩ සැලසෙන පරිදි ය. ඉලක්කය වන්නේ පරිශීලකයින්ට ක්ෂණික ප්‍රතිපෝෂණ ලැබෙන අන්තර්ක්‍රියාකාරී පරිසරයක් සැපයීමයි. බොහෝ පරිශීලකයින් එකවර පද්ධතිය මත වැඩ කරන (විශ්ව විද්‍යාලයක හෝ කාර්යාලයක මෙන්) පරිසරය සඳහා මෙම ආකාරයේ පද්ධතිය වඩාත් සුදුසු වේ.

Multi-Programmed Batch Systems, on the other hand, focus on maximizing CPU utilization by keeping multiple jobs in memory and switching between them, but without the need for user interaction during job execution.
බහු-වැඩසටහන් කණ්ඩායම් පද්ධති, අනෙක් අතට, බහු රැකියා මතකයේ තබාගෙන ඒවා අතර මාරු වීමෙන් රැකියා ක්‍රියාත්මක කිරීමේදී පරිශීලක අන්තර්ක්‍රියා අවශ්‍යතාවයකින් තොරව. CPU භාවිතය උපරිම කිරීම කෙරෙහි අවධානය යොමු කරයි.

Let's briefly review the other options / අනෙකුත් විකල්ප කෙටියෙන් සමාලෝචනය කරමු.

Let's review each option.
අපි එක් එක් විකල්පය සමාලෝචනය කරමු.

- **Option / විකල්පය 1)**
  This describes Multi-Programmed Batch Systems, not Time-Sharing systems.
  මෙය කාණ්ඩ පද්ධතිවල ලක්ෂණයක් මිස කාල විභාජන පද්ධති නොවේ.

- **Option / විකල්පය 3)**
  This is characteristic of Batch Systems, not Time-Sharing systems.
  මෙය විස්තර කරන්නේ බහු-වැඩසටහන් කළ කණ්ඩායම් පද්ධති මිස කාලය විභාජන පද්ධති නොවේ.

- **Option / විකල්පය 4)**
  This is not a goal of Time-Sharing Systems, which actually focus on handling multiple jobs.
  මෙය සැබවින්ම බහු කාර්යය නියමකරණය හැසිරවීම කෙරෙහි අවධානය යොමු කරන කාලය විභාජන පද්ධතිවල ඉලක්කයක් නොවේ.

- **Option / විකල්පය 5)**
  This describes the Batch Processing approach, not Time-Sharing systems.
  මෙය විස්තර කරන්නේ කාණ්ඩ සැකසුම් ප්‍රවේශය මිස කාලය විභාජන පද්ධති නොවේ.

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

6. What is the purpose of `SELECT * FROM tableName`?
`SELECT* FROM tableName` හි අරමුණ කුමක්ද?

    1) Viewing details of a structure of a table / වගුවක ව්‍යුහයක විස්තර බැලීම.
    2) Using an available database / පවතින දත්ත සමුදායක් භාවිතා කිරීම.
    3) Creating a new database / නව දත්ත සමුදායක් නිර්මාණය කිරීම.
    4) Viewing all the data of a table / වගුවක සියලුම දත්ත බැලීම.
    5) Deleting a table / වගුවක් මකා දැමීම.

Answer : 4)

Correct answer 4) Viewing all the data of a table
නිවැරදි පිළිතුර 4) Viewing all the data of a table

Explanation / පැහැදිලි කිරීම

For the following actions, we used these code lines,

1) Viewing details of a structure of a table
වගුවක ව්‍යුහයක විස්තර බැලීම - DESCRIBE *tableName*

2) Using an available database
පවතින දත්ත සමුදායක් භාවිතා කිරීම - USE *databaseName* / CONNECT *databaseName*

3) Creating a new database
නව දත්ත සමුදායක් නිර්මාණය කිරීම - CREATE DATABASE *databaseName* / CREATE SCHEMA *databaseName*

4) Viewing all the data of a table
වගුවක සියලුම දත්ත බැලීම - SELECT*FROM *tableName*

5) Deleting a table
වගුවක් මකා දැමීම – DROP TABLE *tableName*

The final answer is 4)
අවසාන පිළිතුර වන්නේ 4)

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

```
a={'A':1,'B':2,'C':3}
b='hello'
for x in a:
    print(b+x)
```

1)	2)	3)	4)	5)

`{hello1,hello2,hello3}`	`hello`
	`A`
	`B`
	`C`
		`helloA`
		`helloB`
		`helloC`
			`hello1`
			`hello2`
			`hello3`
				`syntax error`

Answer : 3)

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

```
a={'A':1,'B':2,'C':3}
```

This line creates a dictionary named `a` with three key-value pairs.
මෙම පේළිය යතුරු-අගය යුගල තුනක් සහිත `a` නමින් dictionary එකක් නිර්මාණය කරයි.

- 'A' is associated with the value 1. / 'A' අගය 1 සමඟ සම්බන්ධ වේ.
- 'B' is associated with the value 2. / 'B' අගය 2 සමඟ සම්බන්ධ වේ.
- 'C' is associated with the value 3. / 'C' අගය 3 සමඟ සම්බන්ධ වේ.

```
b='hello'
```

This line defines a string b with the value 'hello'.
මෙම පේළිය 'hello' අගය සහිත b නමින් string එකක් අර්ථ දක්වයි.

```
for x in a:
    print(b+x)
```

- The loop iterates over the keys of the dictionary a.
  ලූපය a dictionary එකෙහි යතුරු මත පුනරාවර්තනය වේ.

- In each iteration, x takes the value of the current key in the dictionary.
  එක් එක් පුනරාවර්තනයකදී, x dictionary එකෙහි වත්මන් යතුරේ අගය ගනී.

- Inside the loop, the string b is concatenated with the current key x (which is a string) using the + operator.
  ලූපය තුළ, b string එක + මෙහෙයවනය භාවිතයෙන් වත්මන් යතුර x (එය string එකක්) සමඟ සම්බන්ධ වේ.

- The result of the concatenation (b + x) is then printed.
  පසුව සංයුක්ත කිරීමේ ප්‍රතිඵලය (b + x) ප්‍රතිදානය කරනු ලැබේ.

- In the first iteration, x is 'A', so it prints b + 'A', which is 'helloA'.
  පළමු පුනරාවර්තනයේ දී, x යනු 'A' වේ, එබැවින් එය b + 'A' ප්‍රතිදානය කරයි, එය 'helloA' වේ.

- In the second iteration, x is 'B', so it prints b + 'B', which is 'helloB'.
  දෙවන පුනරාවර්තනයේ දී, x යනු 'B' වේ, එබැවින් එය b + 'B' ප්‍රතිදානය කරයි, එය 'helloB' වේ.

- In the third iteration, x is 'C', so it prints b + 'C', which is 'helloC'.
  තුන්වන පුනරාවර්තනයේ දී, x යනු 'C' වේ, එබැවින් එය b + 'C' ප්‍රතිදානය කරයි, එය 'helloC' වේ.

The code will output the following.
කේතය පහත සඳහන් දේ ප්‍රතිදානය කරයි.

```
helloA
helloB
helloC
```

So, the correct answer number here is 3.
එමනිසා, මෙහි නිවැරදි පිළිතුරු අංකය 3 වේ.

8. Which of the following Python functions is used to convert a Unicode code point into its corresponding character?
යුනිකෝඩ් කේත ලක්ෂ්‍යයක් එහි අනුරූප අක්ෂරය බවට පරිවර්තනය කිරීමට භාවිතා කරන පහත සඳහන් කුමන පයිතන් ශ්‍රිතයද?

1) `ord()` 2) `chr()` 3) `str()` 4) `ascii()` 5) `hex()`

Answer: 2)

Explanation of each option / එක් එක් විකල්පය පැහැදිලි කිරීම:

1) **`ord()`:**
Converts a character into its Unicode code point.
අක්ෂරයක් එහි යුනිකෝඩ් කේත ලක්ෂ්‍යය බවට පරිවර්තනය කරයි.

```
print(ord('A')) # Output: 65
```

2) **`chr()`:**
Takes an integer representing a Unicode code point and returns the corresponding character.
යුනිකෝඩ් කේත ලක්ෂ්‍යයක් නියෝජනය කරන පූර්ණ සංඛ්‍යාවක් ගෙන අනුරූප අක්ෂරය ආපසු ලබා දෙයි.

```
print(chr(65))  # Output: 'A'
```

3) **`str()`:**
Converts data into a string.
දත්ත string එකක් බවට පරිවර්තනය කරයි.

```
print(str(123)) # Output: '123'
```

4) **`ascii()`:**
Returns a string containing printable characters, with non-ASCII characters escaped.
ASCII නොවන අක්ෂර ගැලවී ගොස් ඇති මුද්‍රණය කළ හැකි අක්ෂර අඩංගු string එකක් ආපසු ලබා දෙයි.

```
print(ascii('Hello'))  # Output: 'Hello' (no change since all are ASCII)
print(ascii('Hëllo'))  # Output: 'H\\xebllo' ('ë' is escaped as \xeb)
```

5) **`hex()`:**
Converts an integer to its hexadecimal string representation.
නිඛිලයක් එහි ෂඩ් දශම strings නිරූපණයට පරිවර්තනය කරයි.

```
print(hex(255)) # Output: '0xff' (255 in hexadecimal)
```

Therefore, the Python function used to convert a Unicode code point into its corresponding character is `chr()`.
එබැවින්, යුනිකෝඩ් කේත ලක්ෂ්‍යයක් එහි අනුරූප අක්ෂරය බවට පරිවර්තනය කිරීමට භාවිතා කරන පයිතන් ශ්‍රිතය `chr()` වේ.

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

```
x = [1,2,3]
y = x.copy()
y.append(4)
y.pop(2)
print(y)
```

1) `[1, 2, 3, 4]` 2) `[1, 2, 3]` 3) `[1, 2, 4]` 4) `1, 2, 3, 4` 5) `'1','2','4'`

Answer : 3)

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

`x = [1,2,3]`
This creates a list `x` with the elements `[1, 2, 3]`.
මෙය මූලාංග `[1, 2, 3]` සමඟ `x` ලැයිස්තුවක් නිර්මාණය කරයි.

`y = x.copy()`
This creates a copy of the list `x` and assigns it to the new variable `y`. Now, `y` is a separate list with the same elements as `x`, i.e., `[1, 2, 3]`.
මෙය `x` ලැයිස්තුවේ පිටපතක් සාදා එය නව විචල්‍ය `y` වෙත පවරයි. දැන්, `y` යනු `x` ට සමාන මූලාංග සහිත වෙනම ලැයිස්තුවකි, එනම් `[1, 2, 3]`.

`y.append(4)`
This appends the value 4 to the end of the list `y`. After this operation, `y` becomes `[1, 2, 3, 4]`.
මෙය `y` ලැයිස්තුවේ අවසානයට 4 අගය එකතු කරයි. මෙම මෙහෙයුමෙන් පසුව, `y`, `[1, 2, 3, 4]` බවට පත් වේ.

`y.pop(2)`
The `pop()` method removes and returns the element at the specified index. Here, it removes the element at index 2 in the list `y`. The element at index 2 is 3. After removing it, `y` becomes
`[1, 2, 4]`.
`pop()` ක්‍රමය මඟින් නිශ්චිත දර්ශකයේ ඇති මූලාංගය ඉවත් කර ආපසු ලබා දේ. මෙන්න, එය `y` ලැයිස්තුවේ 2 දර්ශකයේ මූලාංගය ඉවත් කරයි. 2 දර්ශකයෙහි ඇති මූලාංගය 3 වේ. එය ඉවත් කිරීමෙන් පසු `y`, `[1, 2, 4]` බවට පත් වේ.

`print(y)`
This prints the final list `y`, which is `[1, 2, 4]`.
මෙය අවසාන `y` ලැයිස්තුව ප්‍රතිදානය කරයි, එනම් `[1, 2, 4]`.

So, the correct answer number is 3.
එබැවින්, නිවැරදි පිළිතුරු අංකය 3 වේ.

10. You have a link on your webpage, but it is not working. Which of the following could be the reason?
ඔබට ඔබේ වෙබ් පිටුවේ සබැඳියක් ඇත, නමුත් එය ක්‍රියා නොකරයි. පහත සඳහන් ඒවායින් කුමන හේතුවක් විය හැකිද?

1) The <a> tag is missing the href attribute
<a> උසුලනය href ගුණාංගය මග හැරී ඇත.
2) The link is styled with text-decoration: none;.
සබැඳිය පෙළ-අලංකාරයෙන් text-decoration: none; ලෙස හැඩගස්වා ඇත
3) The link is inside a <button> tag
සබැඳිය <button> උසුලනයක් තුළ ඇත.
4) The link is wrapped in a <span>
සබැඳිය <span>. එකකින් ඔතා ඇත.
5) The link is using a relative path instead of an absolute path.
සබැඳිය නිරපේක්ෂ මාර්ගයක් වෙනුවට සාපේක්ෂ මාර්ගයක් භාවිතා කරයි.

Answer : 1)

Explanation / පැහැදිලි කිරීම:

Correct Answer is 1) The `<a>` tag is missing the `href` attribute.
නිවැරදි පිළිතුර : <a> උසුලනයට href ගුණාංගය මග හැරී ඇත.

- `<a>` without `href` doesn't function as a link.
`href` නොමැතිව `<a>` සබැඳියක් ලෙස ක්‍රියා නොකරයි.

Why not the others? / ඇයි අනිත් ඒවා වැරදි?

- 2) `text-decoration: none;` removes the underline but doesn't break the link.
`text-decoration: none;` යටි ඉරි ඉවත් කරන නමුත් සබැඳිය බිඳ නොදමයි.
- 3 & 4) Wrapping in `<button>` or `<span>` doesn't break the link unless styled improperly.
3 සහ 4) `<button>` හෝ `<span>` තුළ Wrap කිරීම නුසුදුසු ලෙස හැඩගස්වා ඇත්නම් මිස සබැඳිය නොකැඩී යයි.