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 197 MARKING

2025/06/25
ictfromabc
Ravindu Bandaranayake

1. Which of the following storage types offers the fastest access time?
පහත සඳහන් ගබඩා වර්ග අතුරින් වේගවත්ම ප්‍රවේශ කාලය ලබා දෙන්නේ කුමක්ද?

1) Hard disk drive / දෘඪ තැටි ධාවකය
2) Solid State Drive (SSD) / ඝන තත්ව ධාවකය
3) Read-Only Memory (ROM) / පඨන මාත්‍ර මතකය
4) Random Access Memory (RAM) / සසම්භාවී ප්‍රවේශ මතකය
5) Magnetic Tape / චුම්බක පටි

Answer: 4)

Explanation of each option / එක් එක් විකල්පය පැහැදිලි කිරීම:

Option / විකල්ප 1)
Relatively slow due to mechanical parts (spinning platters, read/write heads).
යාන්ත්‍රික කොටස් (භ්‍රමණය වන තැටි, කියවීමේ/ලිවීමේ හිස්) නිසා සාපේක්ෂව මන්දගාමී වේ.

Option / විකල්ප 2)
Much faster than HDDs as it uses flash memory and has no moving parts, but still slower than RAM.
සැනෙලි මතකය භාවිතා කරන අතර, චලනය වන කොටස් නොමැති බැවින් HDD වලට වඩා බොහෝ වේගවත් වේ, නමුත් තවමත් RAM වලට වඩා මන්දගාමී වේ.

Option / විකල්ප 3)
Accessing data from ROM is slower than from RAM, because ROM is not meant for frequent or rapid reading during normal operations. Unlike RAM, which allows fast, direct access to data needed by the CPU in real time, ROM is only accessed occasionally and is typically much slower in access time.
ROM වලින් දත්ත ප්‍රවේශ කිරීම RAM වලින් දත්ත ප්‍රවේශ කිරීමට වඩා මන්දගාමී වේ, මන්ද, ROM සාමාන්‍ය මෙහෙයුම් වලදී නිතර හෝ වේගවත් කියවීම සඳහා අදහස් නොකරයි. CPU මගින් තත්‍ය කාලීනව අවශ්‍ය දත්ත වෙත වේගවත්, සෘජු ප්‍රවේශයක් ලබා දෙන RAM මෙන් නොව, ROM වෙත ප්‍රවේශ වන්නේ ඉඳහිට පමණක් වන අතර සාමාන්‍යයෙන් ප්‍රවේශ කාලය තුළ එය බෙහෙවින් මන්දගාමී වේ.

Option / විකල්ප 4)
This is the fastest type of volatile memory, directly accessible by the CPU, and used for actively running programs and data.
මෙය CPU මගින් සෘජුවම ප්‍රවේශ විය හැකි වේගවත්ම නශ්‍ය මතක වර්ගය වන අතර, ක්‍රියාකාරීව ක්‍රියාත්මක වන වැඩසටහන් සහ දත්ත සඳහා භාවිතා කරයි.

Option / විකල්ප 5)
These are very slow in sequential access, primarily used for archival storage.
මේවායේ අනුක්‍රමික ප්‍රවේශය ඉතා මන්දගාමී වන අතර, ප්‍රධාන වශයෙන් සංරක්ෂිත ගබඩා කිරීම සඳහා භාවිතා වේ.

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

2. Which of the following statements is true about 2's complement?
2හි අනුපූරක සම්බන්ධයෙන් සත්‍ය වන්නේ පහත කුමන ප්‍රකාශයද?

1) Used in first generation computers to perform logical operations.
තාර්කික මෙහෙයුම් සිදු කිරීම සඳහා පළමු පරම්පරාවේ පරිගණකවල භාවිතා විය.

2) Addition and subtraction are two different operations.
එකතු කිරීම සහ අඩු කිරීම එකිනෙකට වෙනස් මෙහෙයුම් දෙකක් ලෙස සිදු කෙරේ.

3) An extra bit is used to represent the sign.
ලකුණ නිරූපනය කිරීමට අමතර බිටුවක් භාවිතා කෙරේ.

4) It helps to develop faster, cheaper hardware to perform arithmetic operations.
අංක ගණිත මෙහෙයුම් ඉටු කිරීම සඳහා වඩා වේගවත්, මිල අඩු දෘඩාංග නිපදවීමට ඉවහල් වේ.

5) Usually represented by the hexadecimal number system.
සාමාන්‍යයෙන් ෂඩ් දශම සංඛ්‍යා පද්ධතිය මගින් නිරූපණය කෙරේ.

Answer: 4)

Explanation / පැහැදිලි කිරීම:
2's complement is widely used in computer systems to represent signed integers. One of its key advantages is that it allows the same hardware circuits to be used for both addition and subtraction, simplifying the design and making it faster and cheaper to implement.
2 හි අනුපූරකය පරිගණක පද්ධතිවල ලකුණුවත් කරන ලද පූර්ණ සංඛ්‍යා නියෝජනය කිරීම සඳහා බහුලව භාවිතා වේ. එහි එක් ප්‍රධාන වාසියක් නම්, එය එකතු කිරීම සහ අඩු කිරීම යන දෙකටම එකම දෘඩාංග පරිපථ භාවිතා කිරීමට ඉඩ සලසා දීම, සැලසුම සරල කිරීම සහ එය ක්‍රියාත්මක කිරීම වේගවත් හා ලාභදායී කිරීමයි.

Explanation of other options / අනෙකුත් විකල්ප පැහැදිලි කිරීම:

Option / විකල්ප 1)
This statement is incorrect. 2's complement was not used in first-generation computers, which often used different methods for representing signed numbers. It became more standard in later generations.
මෙම ප්‍රකාශය අසත්‍ය වේ. 2 හි අනුපූරකය පළමු පරම්පරාවේ පරිගණකවල භාවිතා නොකළ අතර, එහිදී බොහෝ විට ලකුණුවත් කරන ලද අංක නියෝජනය කිරීම සඳහා විවිධ ක්‍රම භාවිතා කරන ලදී. පසු පරම්පරාවල එය වඩාත් සම්මත විය.

Option / විකල්ප 2)
This statement is incorrect. In 2's complement arithmetic, subtraction can be performed using addition by adding the 2's complement of the number, which simplifies the hardware design.
මෙම ප්‍රකාශය අසත්‍ය වේ. 2 හි අනුපූරක අංක ගණිතයේදී, සංඛ්‍යාවේ 2 හි අනුපූරකය එකතු කිරීමෙන්, එකතු කිරීම භාවිතයෙන් අඩු කිරීම සිදුකළ නිසා දෘඩාංග නිර්මාණය සරල කළ හැක.

Option / විකල්ප 3)
This statement is incorrect. In 2's complement representation, the most significant bit (MSB) doubles as the sign bit and as part of the magnitude of the number. There's no separate "extra" bit for the sign.
මෙම ප්‍රකාශය අසත්‍ය වේ. 2 හි අනුපූරක නිරූපණයෙහි, වැඩිම වෙසෙසි බිටු (MSB) සංකේත බිටු ලෙස සහ සංඛ්‍යාවේ විශාලත්වයේ කොටසක් ලෙස දෙගුණ වේ. ලකුණ සඳහා වෙනම "අමතර" බිටු එකක් නොමැත.

Option / විකල්ප 5)
This statement is incorrect. 2's complement numbers are typically represented in binary. Hexadecimal can be used to represent any binary number compactly, but it is not specifically tied to 2's complement.
මෙම ප්‍රකාශය අසත්‍ය වේ. 2 හි අනුපූරක සංඛ්‍යා සාමාන්‍යයෙන් ද්විමය වශයෙන් නිරූපණය කෙරේ. ඕනෑම ද්විමය සංඛ්‍යාවක් සංයුක්තව නිරූපණය කිරීමට ෂඩ් දශමය භාවිතා කළ හැකි නමුත් එය 2 හි අනුපූරකයට නිශ්චිතව බැඳී නැත.

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

3. $(\overline{\overline{A}B}1(\overline{\overline{B}}A + A)A)$ The simplified answer is, / සරල කළ පිළිතුර වන්නේ,

1) 1 2) B 3) A+B 4) A 5) 0

Answer: 4)

Let's simplify the given Boolean expression.
ලබා දී ඇති බූලියානු ප්‍රකාශනය සරල කරමු.

$= (\overline{\overline{A}B}1(\overline{\overline{B}}A + A)A)$
$= (\overline{\overline{A}B}1(BA + A)A)$ Double inverse law / ද්විත්ව ප්‍රතිලෝම න්‍යාය
$= (\overline{\overline{A}B}(BA + A)A)$ Identity law / සර්වසාම්‍ය න්‍යාය
$= ((\overline{\overline{A}} + \overline{B})(BA + A)A)$ De Morgan's law / ද මුවාර් ප්‍රමේය
$= ((A + \overline{B})(BA + A)A)$ Double inverse law / ද්විත්ව ප්‍රතිලෝම න්‍යාය
$= (A + \overline{B})A.A$ Redundancy law / සමතිරික්ත න්‍යාය
$= (A + \overline{B})A$ Idempotent law / තද්ේවහාවි න්‍යාය
$= AA + A\overline{B}$ Distributive law / විෂටන න්‍යාය
$= A + A\overline{B}$ Idempotent law / තද්ේවහාවි න්‍යාය
$= A$ Redundancy law / සමතිරික්ත න්‍යාය

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

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

```
my_list = [10, 20, 30, 40, 50]
print(my_list[1:4])
```

1) `[10, 20, 30]` 2) `[20, 30, 40]` 3) `[10, 20, 30, 40]`
4) `[20, 30, 40, 50]` 5) `[10, 20]`

Answer: 2)

Explanation / පැහැදිලි කිරීම:

`my_list = [10, 20, 30, 40, 50]`
This line initializes a list named `my_list` with five integer elements. In Python, list elements are indexed starting from `0`.
මෙම පේළිය නිඛිල සංඛ්‍යා මූලාංග පහක් සහිත `my_list` නම් ලැයිස්තුවක් ආරම්භ කරයි. පයිතන්හි, ලැයිස්තු මූලාංග `0` සිට සුවිගත කර ඇත.

`print(my_list[1:4])`
This line uses list slicing. The syntax `[start:end]` returns a new list containing elements from the `start` index up to the `end` index (not including the `end` index).
මෙම පේළිය ලැයිස්තු slicing කිරීම භාවිතා කරයි. `[start:end]` ඛණ්ඩය ආරම්භක දර්ශකයේ සිට අවසාන දර්ශකය (අවසාන දර්ශකය ඇතුළත් නොකර) දක්වා මූලාංග අඩංගු නව ලැයිස්තුවක් ආපසු ලබා දෙයි.

So, the slice will include elements at indices `1, 2` and `3`.
එමනිසා, slice එකේ දර්ශක `1, 2` සහ `3` හි මූලාංග ඇතුළත් වේ.

```
my_list[1] = 20
my_list[2] = 30
my_list[3] = 40
```

The `print()` function will output the sub-list `[20, 30, 40]` .
`print()` ශ්‍රිතය `[20, 30, 40]` යන උප-ලැයිස්තුව ප්‍රතිදානය කරයි.

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

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

```
Movies = [2005, 2010, 2015, 2020]
sorted_movies = list(reversed(sorted(Movies)))
print(sorted_movies)
```

1) `(2005, 2015, 2010, 2020)`
2) `['2005','2015', '2010', '2020']`
3) `[2005, 2015, 2010, 2020]`
4) `[2020, 2015, 2010, 2005]`
5) `[2005, 2010, 2015, 2020]`

Answer: 4)

Explanation / පැහැදිලි කිරීම:

The code first sorts the list `Movies` using `sorted(Movies)`, which arranges the elements `[2005, 2010, 2015, 2020]` in ascending order. This operation doesn't change the order of the elements since they are already in ascending order, resulting in `[2005, 2010, 2015, 2020].`
කේතය ප්‍රථමයෙන් `sorted(Movies)` භාවිතයෙන් `Movies` ලැයිස්තුව අනුපිළිවෙලට සකසයි, එමඟින් මූලාංග `[2005, 2010, 2015, 2020]` ආරෝහණ අනුපිළිවෙලට සකසයි. මෙම මෙහෙයුම මූලාංගවල අනුපිළිවෙල වෙනස් නොකරන්නේ ඒවා දැනටමත් ආරෝහණ අනුපිළිවෙලෙහි පවතින බැවිනි. එහි ප්‍රතිඵලයක් ලෙස `[2005, 2010, 2015, 2020]` ලැබේ.

Next, the `reversed()` function is applied to the sorted list. This function creates an iterator that yields the elements in reverse order. When `reversed(sorted(Movies))` is used, the sorted list `[2005, 2010, 2015, 2020]` is reversed to `[2020, 2015, 2010, 2005].`
ඊළඟට, වර්ග කළ ලැයිස්තුවට `reversed()` ශ්‍රිතය යොදනු ලැබේ. මෙම ශ්‍රිතය ප්‍රතිලෝම අනුපිළිවෙලින් මූලාංග ලබා දෙන පුනරාවර්තකයක් නිර්මාණය කරයි. `reversed(sorted(Movies))` භාවිතා කරන විට, අනුපිළිවෙලට සකසන ලද ලැයිස්තුව `[2005, 2010, 2015, 2020]` යන්න `[2020, 2015, 2010, 2005]` වෙත ආපසු හරවනු ලැබේ.

Finally, `list(reversed(sorted(Movies)))` converts this reversed iterator into a list. The resulting list `[2020, 2015, 2010, 2005]` is then printed. Thus, the output of `print(sorted_movies)` is `[2020, 2015, 2010, 2005]`.
අවසාන වශයෙන්, `list(reversed(sorted(Movies)))` මගින් මෙම ප්‍රතිලෝම පුනරාවර්තකය ලැයිස්තුවක් බවට පරිවර්තනය කරයි. ප්‍රතිඵලය වන ලැයිස්තුව `[2020, 2015, 2010, 2005]` පසුව මුද්‍රණය කෙරේ. මේ අනුව, `print(sorted_movies)` හි ප්‍රතිදානය `[2020, 2015, 2010, 2005]` වේ.

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