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 249 MARKING

2025/08/16
ictfromabc
Ravindu Bandaranayake

1. Consider the following features related to memory.
මතකය සම්බන්ධ පහත විශේෂාංග සලකා බලන්න.

   A. It is part of the central processing unit.
   එය මධ්‍ය සැකසුම් ඒකකයේ කොටසකි.

   B. Used in logical operations to temporarily store the results.
   තාර්කික ක්‍රියා වලදී ප්‍රතිඵල තාවකාලිකව ගබඩා කිරීම සඳහා භාවිතා වේ.

   C. It has the highest data transmission speed compared to others.
   එයට අනෙක් ඒවාට සාපේක්ෂව ඉහළම දත්ත සම්ප්‍රේෂණ වේගයක් ඇත.

   What is the memory component that matches with the above features?
   ඉහත විශේෂාංග සමඟ ගැළපෙන මතක සංරචකය කුමක්ද?

   1) L1 Cache memory / L1 චාරක මතකය.
   2) L2 Cache memory / L2 චාරක මතකය.
   3) Static Random Access Memory / **ස්ථිතික සසම්භාවී ප්‍රවේශ මතකය.**
   4) Main memory / **ප්‍රධාන මතකය.**
   5) Register / **රෙජිස්තර.**

   Answer: 5)

   The memory component that matches with the above features is (5) Register.
   ඉහත විශේෂාංග සමඟ ගැලපෙන මතක සංරචකය (5) රෙජිස්තර වේ.

   Explanation / පැහැදිලි කිරීම:

   A register is a small amount of storage available as part of a CPU. The CPU uses registers to store data that needs to be accessed quickly during the execution of a program.
   රෙජිස්තර යනු CPU එකක කොටසක් ලෙස පවතින කුඩා මතක උපාංගයකි. වැඩසටහනක් ක්‍රියාත්මක කිරීමේදී ඉක්මනින් ප්‍රවේශ විය යුතු දත්ත ගබඩා කිරීමට CPU රෙජිස්තර භාවිතා කරයි.

   It is part of the central processing unit (CPU): Registers are located within the CPU itself, making them the fastest type of memory in a computer. They are used to store data that the CPU needs to access quickly.
   එය මධ්‍ය සැකසුම් ඒකකයේ (CPU) කොටසකි: රෙජිස්තර CPU තුළම පිහිටා ඇති අතර, ඒවා පරිගණකයක වේගවත්ම මතක වර්ගය වේ. CPU එකට ඉක්මනින් ප්‍රවේශ වීමට අවශ්‍ය දත්ත ගබඩා කිරීමට ඒවා භාවිතා වේ.

   Used in logical operations to temporarily store the results: Registers are used to hold intermediate results of computations and control information. They are essential for the CPU to perform logical operations and calculations.
   ප්‍රතිඵල තාවකාලිකව ගබඩා කිරීම සඳහා තාර්කික මෙහෙයුම් වලදී භාවිතා වේ: ගණනය කිරීම් සහ පාලන තොරතුරු වල අතරමැදි ප්‍රතිඵල රඳවා ගැනීමට රෙජිස්තර භාවිතා කරයි. ඒවා තාර්කික මෙහෙයුම් සහ ගණනය කිරීම් සිදු කිරීමට CPU සඳහා අත්‍යවශ්‍ය වේ.

It has the highest data transmission speed compared to others: Since registers are part of the CPU, the data transmission speed is the highest compared to other types of memory. This is because the data doesn't need to travel over the system bus, which can be a bottleneck for data transmission.
එය අනෙක් ඒවාට සාපේක්ෂව ඉහළම දත්ත සම්ප්‍රේෂණ වේගයක් ඇත: රෙජිස්තර CPU හි කොටසක් බැවින්, දත්ත සම්ප්‍රේෂණ වේගය, අනෙකුත් මතක වර්ග හා සසඳන විට ඉහළම වේ. මෙයට හේතුව දත්ත සම්ප්‍රේෂණය සඳහා බාධාවක් විය හැකි පද්ධති බසය හරහා දත්ත ගමන් කිරීමට අවශ්‍ය නොවන බැවිනි.

In contrast, other types of memory like L2 Cache, L3 Cache, Static Random Access Memory (SRAM), and Main Memory are not located inside the CPU, and thus, they do not have the same level of data transmission speed as registers.
ඊට ප්‍රතිවිරුද්ධව, L2 වාරක මතකය, L3 වාරක මතකය, ස්ථිතික සසම්භාවී ප්‍රවේශ මතකය සහ ප්‍රධාන මතකය. වැනි අනෙකුත් මතක වර්ග CPU තුළ පිහිටා නොමැති අතර, ඒ අනුව, ඒවාට රෙජිස්තර වලට සමාන දත්ත සම්ප්‍රේෂණ වේගයක් නොමැත.

2. Below are some statements about Central Processing Unit and Motherboard, which statement(s) is/are true?
පහතින් දැක්වෙන්නේ මධ්‍ය සැකසුම් ඒකකය සහ මව්පුවරුව සම්බන්ද ප්‍රකාශන කිහිපයකි, මේවා අතරින් සත්‍ය වන්නේ කුමන ප්‍රකාශන(ය)ද?

A. When purchasing a motherboard you should check what type of CPU socket your motherboard has.
මව්පුවරුවක් මිලදී ගැනීමේදී එහි ඇත්තේ කුමන ආකාරයේ CPU කෙවෙනියක් දැයි පරීක්ෂා කළ යුතුය.

B. The motherboard's chipset is also important as it determines many of the features and I/O ports you can use on your motherboard.
මව්පුවරුවේ chipset එක ද වැදගත් වන්නේ එය ඔබගේ මව්පුවරුවේ භාවිතා කළ හැකි බොහෝ විශේෂාංග සහ I/O කෙවෙනි තීරණය කරන බැවිනි.

C. Some motherboards may require a BIOS update to support newer CPUs.
සමහර මව්පුවරු වලට නව CPU සඳහා සහය දැක්වීමට BIOS යාවත්කාලීනයක් අවශ්‍ය විය හැක.

1) A only. **A පමණි.**
2) B only. **B පමණි.**
3) C only. **C පමණි.**
4) A and C only. **A හා C පමණි.**
5) <u>All of the above.</u> **ඉහත සියල්ලම.**

Answer: 5)

Evaluating each statement / **එක් එක් ප්‍රකාශය පැහැදිලි කිරීම:**

A. When purchasing a motherboard you should check what type of CPU socket your motherboard has.
මව්පුවරුවක් මිල දී ගැනීමේදී ඔබේ මව්පුවරුවේ ඇත්තේ කුමන වර්ගයේ CPU කෙවෙනියක්ද යන්න පරීක්ෂා කළ යුතුය.

This statement is true because the CPU socket on the motherboard must match the socket type of the CPU you intend to use. If they are not compatible, the CPU cannot be installed on the motherboard.
මෙම ප්‍රකාශය සත්‍ය වන්නේ මව්පුවරුවේ ඇති CPU කෙවෙනිය ඔබ භාවිතා කිරීමට අදහස් කරන CPU හි කෙවෙනි වර්ගයට ගැළපිය යුතු බැවිනි. ඒවා නොගැළපේ නම්, CPU මව්පුවරුවේ ස්ථාපනය කළ නොහැක.

B. The motherboard's chipset is also important as it determines many of the features and I/O ports you can use on your motherboard.
මව්පුවරුවේ චිප්සෙට් එක වැදගත් වන්නේ එය ඔබගේ මව්පුවරුවේ භාවිතා කළ හැකි බොහෝ විශේෂාංග සහ ආදාන ප්‍රතිදාන කෙවෙනි (I/O ports) තීරණය කරන බැවිනි.

This statement is true. The chipset on the motherboard controls the communication between the CPU and other components. Different chipsets offer different features and connectivity options, so choosing the right chipset is crucial for getting the desired functionality from your motherboard.
මෙම ප්‍රකාශය සත්‍ය වේ. මව්පුවරුවේ ඇති චිප්සෙට් එක CPU සහ අනෙකුත් සංරචක අතර සන්නිවේදනය පාලනය කරයි. විවිධ චිප්සෙට් විවිධ විශේෂාංග සහ සම්බන්ධතා විකල්ප ලබා දෙයි, එබැවින් ඔබේ මව්පුවරුවෙන් අපේක්ෂිත ක්‍රියාකාරීත්වය ලබා ගැනීම සඳහා නිවැරදි චිප්සෙට් එක තෝරා ගැනීම ඉතා වැදගත් වේ.

C. Some motherboards may require a BIOS update to support newer CPUs.
සමහර මව්පුවරු වලට නව CPU සඳහා සහය දැක්වීමට BIOS යාවත්කාලීනයක් අවශ්‍ය විය හැක.

This statement is also true. Motherboards often need BIOS updates to support newer CPUs released after the motherboard's manufacturing date. Without the update, the motherboard may not recognize or properly function with the newer CPU.
මෙම ප්‍රකාශය සත්‍ය වේ . මව්පුවරුවේ නිෂ්පාදන දිනයට පසුව නිකුත් කරන ලද නව CPU වලට සහය දැක්වීම සඳහා මව්පුවරුවලට බොහෝ විට BIOS යාවත්කාලීන අවශ්‍ය වේ. යාවත්කාලීන කිරීමකින් තොරව, මව්පුවරුව නව CPU සමඟ හඳුනා ගැනීමට හෝ නිසි ලෙස ක්‍රියා කිරීමට නොහැකි විය හැක.

So, all of the statements are true and collectively provide important considerations when dealing with CPUs and motherboards.
එබැවින්, සියලුම ප්‍රකාශයන් සත්‍ය වන අතර CPU සහ මව්පුවරු සමඟ කටයුතු කිරීමේදී සාමූහිකව වැදගත් කරුණු සපයයි.

The correct answer is 5) All of the above.
නිවැරදි පිළිතුර 5) ඉහත සියල්ල.

3. $110111_2 + 101100_2$ is equal to,
$110111_2 + 101100_2$ ට තුල්‍ය වන්නේ,

A. 99_{10} B. 1100011_2 C. 147_8

1) A only.
A පමණි.
2) B only.
B පමණි.
3) C only.
C පමණි.
4) A and B only.
A හා B පමණි.
5) All of the above.
ඉහත සියල්ලම.

Answer: 4)

Explanation / පැහැදිලි කිරීම:

First let's add the two binary numbers here.
පළමුව මෙහි ඇති ද්විමය සංඛ්‍යා දෙක එකතු කර සිටිමු.

$$\begin{array}{r} 110111_2 \\ +101100_2 \\ \hline 1100011_2 \end{array}$$

Then it appears that this value is equal to the value of B.
එවිට මෙම අගය B හි අගයට සමාන බව පෙනියයි.

Now let's convert this binary value into decimal.
දැන් මෙම ද්විමය අගය දශමය බවට හරවා බලමු.

1100011_2						
1	1	0	0	0	1	1
2^6	2^5	2^4	2^3	2^2	2^1	2^0
64	32	16	8	4	2	1
64 + 32 + 2 + 1						
99_{10}						

Hence this value is equal to the value of A.
ඒ අනුව මෙම අගය A හි අගයට තුල්‍ය වේ.

Now let's convert this binary value into octal.
දැන් මෙම ද්විමය අගය අෂ්ටමය බවට හරවා බලමු.

1100011_2		
001	100	011
1	4	3
143_8		

Hence this value is not equal to the value of C.
ඒ අනුව මෙම අගය C හි අගයට තුල්‍ය නොවේ.

Accordingly, the correct answer is only A and B.
ඒ අනුව නිවැරදි පිළිතුර වන්නේ A සහ B පමණි යන්නයි.

4. Which number is represented by the following data representation represented by sign bit representation?
ලකුණුවත් නිරූපණය මගින් නිරූපණය කර ඇති පහත දත්ත නිරූපණයෙන් නිරූපණය වන සංඛ්‍යාව කුමක්ද?

1	0	0	0	1	1	0	0

1) 140 2) -140 3) 12 4) <u>-12</u> 5) 1100

Answer: 4)

Explanation / පැහැදිලි කිරීම:

In 8-bit signed magnitude representation, the leftmost bit (most significant bit) represents the sign of the number, where 0 denotes positive and 1 denotes negative. The remaining 7 bits represent the magnitude of the number.
බිටු 8හි ලකුණුවත් ප්‍රමාණය කරන ලද විශාලත්වය නියෝජනය කිරීමේදී, වම්පස ඇති බිටුව සංඛ්‍යාවේ සලකුණ නියෝජනය කරයි, එහිදී 0 ධන සහ 1 සෘණ නිරූපණය කරයි. ඉතිරි බිටු 7 අංකයේ විශාලත්වය නියෝජනය කරයි.

1	1	0	0
2^3	2^2	2^1	2^0
8	4	2	1
8x1	4x1	2x0	1x0
8	4	0	0
8 + 4 = 12			

Since the leftmost bit is 1, the answer is a negative number. So, the answer here is -12.
මෙහි වම් පස පළමු බිටුව 1 බැවින් මෙහි පිළිතුර සෘණ සංඛ්‍යාවක් වේ. එබැවින්, මෙහි පිළිතුර -12 වේ.

- Questions 5 and 6 are based on the below scenario.
ප්‍රශ්න 5 සහ 6 පහත සිදුවීම මත පදනම් වේ.

Two inputs P and Q are simultaneously fed to an AND gate and an OR gate then the output of the AND gate is gone through an XOR gate with the P input and the output of the OR gate is gone through another XOR gate with the Q input and the output of the last two gates are fed to an OR gate.
P සහ Q ආදාන දෙකක් එකවර AND ද්වාරයකට සහ OR ද්වාරයකට ආදානය කෙරේ. එවිට AND ද්වාරයේ ප්‍රතිදානය P ආදානය සමඟ XOR ද්වාරයක් හරහා ගොස් OR ද්වාරයෙහි ප්‍රතිදානය Q ආදානය සමඟ වෙනත් XOR ද්වාරයක් හරහා යයි. අවසාන ද්වාර දෙකේ ප්‍රතිදානය OR ද්වාරයකට ආදානය වේ.

5. What will be the output of the above logic circuit scenario?
ඉහත තාර්කික පරිපථ සිදුවීමෙහි ප්‍රතිදානය කුමක් වේද?

1) $((P.Q) \oplus Q) + ((P+Q) \oplus P)$
2) $((P.Q) \oplus P) . ((P+Q) \oplus Q)$
3) $\underline{((P.Q) \oplus P) + ((P+Q) \oplus Q)}$
4) $((P.Q) \oplus Q) + ((P.Q) \oplus P)$
5) $((P.Q) \oplus P) + ((P.Q) \oplus Q)$

Answer: 3)

Explanation / පැහැදිලි කිරීම:

The first AND gate will give us (P.Q) and then this is gone through an XOR gate with P input this will result in $((P.Q) \oplus P)$, then the first OR gate will give us (P+Q) then this is fed into another XOR gate with the Q input, which will result in $((P+Q) \oplus Q)$, finally both these outputs will go through an OR gate so the output will be $((P.Q) \oplus P) + ((P+Q) \oplus Q)$
පළමු AND ද්වාරය අපට (P.Q) ලබා දෙන අතර පසුව මෙය P ආදානය සහිත XOR ද්වාරයක් හරහා ගිය විට $((P.Q) \oplus P)$ ලෙස ප්‍රතිඵලයක් ලැබෙනු ඇත. පළමු OR ද්වාරය අපට ලබා දෙනු ඇත්තේ (P+Q) වේ. පසුව මෙය Q ආදානය සමඟ වෙනත් XOR ද්වාරයකට ආදානය කරනු ලැබේ, එහි ප්‍රතිඵලය වනු ඇත්තේ $((P+Q) \oplus Q)$ වේ. අවසානයේ මෙම ප්‍රතිදානය දෙකම OR ද්වාරයක් හරහා යන බැවින් ප්‍රතිදානය $((P.Q) \oplus P) + ((P+Q) \oplus Q)$ වේ.

The correct answer is 3).
නිවැරදි පිළිතුර 3) වේ.

6. What is the simplified expression of the above logic circuit scenario?
ඉහත තාර්කික පරිපථ සිදුවීමෙහි සරල කළ ප්‍රකාශනය කුමක්ද?

1) $P . Q$ 2) $P + Q$ 3) $P \oplus Q$ 4) $\underline{P . \overline{Q}}$ 5) $P + \overline{Q}$

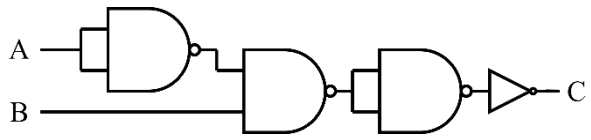
Answer: 4)

Explanation / පැහැදිලි කිරීම:

$= \left((P. Q) \oplus P\right) + \left((P + Q) \oplus Q\right) = \left((P. Q). \overline{P} + \left(\overline{P. Q}\right). P\right) + ((P + Q). \overline{Q} + \left(\overline{P + Q}\right). Q)$
$= \left(P. \overline{P}. Q + \left(\overline{P. Q}\right). P\right) + (P. \overline{Q} + Q. \overline{Q} + \left(\overline{P + Q}\right). Q)$ Distributive law / විෂටන න්‍යාය
$= \left(\left(\overline{P. Q}\right). P\right) + (P. \overline{Q} + \left(\overline{P + Q}\right). Q)$ Inverse law / ප්‍රතිලෝම න්‍යාය
$= \left(\overline{P} + \overline{Q}\right). P + P. \overline{Q} + \overline{P}. \overline{Q}. Q$ De Morgan's law / ද මුවාර් ප්‍රමේයය
$= \left(\overline{P} + \overline{Q}\right). P + P. \overline{Q}$ Inverse law / ප්‍රතිලෝම න්‍යාය
$= \overline{P}. P + \overline{Q}. P + P. \overline{Q}$ Distributive law / විෂටන න්‍යාය
$= \overline{Q}. P + P. \overline{Q}$ Inverse law / ප්‍රතිලෝම න්‍යාය
$= P. \overline{Q}$ Idempotent law / තද්ේවභාවී න්‍යාය

The correct answer is 4).
නිවැරදි පිළිතුර 4) වේ.

7. C equals to,
C සමාන වන්නේ,

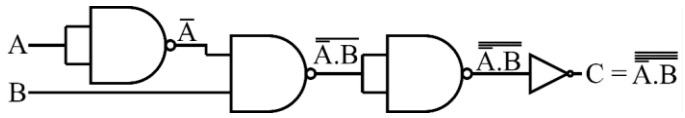


1) $\overline{A} + B$ 2) $\underline{A + \overline{B}}$ 3) $A+B$ 4) $A\overline{B}$ 5) AB

Answer: 2)

Explanation / පැහැදිලි කිරීම:

Let's obtain expressions from the above logical expression.
ඉහත තාර්කික පරිපථයෙන් අපි ප්‍රකාශනය ලබා ගනිමු.



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

$C = (\overline{A}.B)$ Double complement law / ද්විත්ව ප්‍රතිලෝම න්‍යාය

$C = (\overline{\overline{A}} + \overline{B})$ De Morgan's law / ද මුවාර් ප්‍රමේයය

$C = (A + \overline{B})$ Double complement law / ද්විත්ව ප්‍රතිලෝම න්‍යාය

So the correct answer is 2.
එබැවින් නිවැරදි පිළිතුර 2 වේ.

8. What is the correct way to access and modify a global variable inside a function in Python?
පයිතන් හි ශ්‍රිතයක් තුළදී ගෝලීය විචල්‍ය වෙත ප්‍රවේශ වීමේ හා වෙනස් කිරීමේ නිවැරදි ක්‍රමය කුමක්ද?

1) `global x = 1`
2) `x = 1`
3) `declare global x = 1`
4) In any function, use global x
5) `x := 1`

Answer: 4)

Explanation / පැහැදිලි කිරීම:

To declare a global variable in Python, you don't need to use any special syntax outside of functions. However, when you need to modify a global variable within a function, you use the `global` keyword to tell Python that you intend to use the global variable, not a local one.
පයිතන් හි global variable එකක් ප්‍රකාශ කිරීමට, ඔබට ශ්‍රිතයෙන් පිටත කිසිදු විශේෂ වාක්‍ය ඛණ්ඩයක් භාවිතා කිරීමට අවශ්‍ය නොවේ. කෙසේ වෙතත්, ඔබට ශ්‍රිතයක් තුළ ගෝලීය විචල්‍යක් වෙනස් කිරීමට අවශ්‍ය වූ විට, ඔබ local එකක් නොව ගෝලීය විචල්‍යක් භාවිතා කිරීමට අදහස් කරන බව පයිතන් ට පැවසීමට ඔබ `global` මූල පදය භාවිතා කරයි.

Option / විකල්පය 1)
Incorrect. The `global` keyword is used within functions to refer to a global variable, but you cannot use `global` in this way outside a function.
වැරදියි. ගෝලීය විචල්‍යයක් වෙත යොමු කිරීම සඳහා `global` මූල පදය ශ්‍රිත තුළ භාවිතා වේ, නමුත් ඔබට ශ්‍රිතයකින් පිටත මේ ආකාරයෙන් `global` භාවිතා කළ නොහැක.

Option / විකල්පය 2)
Incorrect. This assigns `1` to `x` globally if done outside of any function. However, this is not the complete answer since it doesn't explain how to modify a global variable within a function.
වැරදියි. මෙය කිසියම් function එකකින් පරිබාහිරව සිදු කළහොත් ගෝලීය වශයෙන් `1` සිට `x` වෙත පවරයි. කෙසේ වෙතත්, function එක තුළ ගෝලීය විචල්‍යක් වෙනස් කරන්නේ කෙසේද යන්න පැහැදිලි නොකරන බැවින් මෙය සම්පූර්ණ පිළිතුර නොවේ.

Option / විකල්පය 3)
Incorrect. This is not valid Python syntax.
වැරදියි. මෙය වලංගු පයිතන් syntax එකක් නොවේ.

Option / විකල්පය 4)
Correct. This is how declare that `x` refers to a global variable within a function.
නිවැරදියි. `x` යනු ශ්‍රිතයක් තුළ ඇති ගෝලීය විචල්‍යක් බව ප්‍රකාශ කරන ආකාරය මෙයයි.

Option / විකල්පය 5)
Incorrect. The `:=` is the walrus operator in Python, used for assignment expressions, but it is not used for declaring global variables.
වැරදියි. `:=` යනු පයිතන් හි walrus operator වන අතර, assignment expressions සඳහා භාවිතා කරයි, නමුත් එය ගෝලීය විචල්‍යයන් ප්‍රකාශ කිරීම සඳහා භාවිතා නොවේ.

The correct answer is Option 4) In any function, use `global x`.
නිවැරදි පිළිතුර විකල්පය 4) ඕනෑම ශ්‍රිතයකදී, `global x` භාවිතා කරන්න යි.

9. Fill in the line of the following Python code for calculating the factorial of a number.
සංඛ්‍යාවක ක්‍රමාරෝපිතය ගණනය කිරීම සඳහා පහත පයිතන් කේතයේ පේළිය පුරවන්න.

```
def fact(num):
    if num == 0:
        return 1
    else:
        return _________
```

1) `num*fact(num-1)` 2) `num*fact(num-2)` 3) `(num-1)*(num-2)`
4) `num*(num-1)` 5) `fact(num)*fact(num-1)`

Answer: 1)

Explanation / පැහැදිලි කිරීම:

The function `fact(num)` calculates the factorial of a number using recursion. The factorial of a number `n` is defined as `n * (n-1) * (n-2) * ... * 1`. The recursive call `num*fact(num-1)` ensures the function multiplies `num` by the factorial of the number just below it, gradually decreasing until it reaches the base case `num == 0`, where it returns 1.
`fact(num)` ශ්‍රිතයේ ප්‍රත්‍යාවර්තනය භාවිතයෙන් සංඛ්‍යාවක ක්‍රමාරෝපිතය ගණනය කරයි. `n` සංඛ්‍යාවක සාධක ලෙස `n * (n-1) * (n-2) * ... * 1` අර්ථ දක්වා ඇත. පුනරාවර්තන ඇමතුම් `num*fact(num-1)` ශ්‍රිතය ඊට මදක් පහළින් ඇති සංඛ්‍යාවේ සාධක මගින් සංඛ්‍යාව ගුණ කිරීම සහතික කරයි, එය ක්‍රමයෙන් අඩු වේ. පාදක `num == 0` වෙත ළඟා වේ, එහිදී එය 1 ලබා දෙයි.

Option / විකල්පය 1)
Correct. This is the standard recursive formula for calculating factorials.
නිවැරදියි. ක්‍රමාරෝපිතය ගණනය කිරීමේ සම්මත පුනරාවර්තන සූත්‍රය මෙයයි.

Option / විකල්පය 2)
Incorrect. `num*fact(num-2)` skips intermediate numbers, leading to incorrect results.
වැරදියි. `num*fact(num-2)` අතරමැදි සංඛ්‍යා මඟ හරියි, වැරදි ප්‍රතිඵලවලට මඟ පාදයි.

Option / විකල්පය 3)
Incorrect. `(num-1)*(num-2)` is not recursive and only multiplies two values, not the entire sequence.
වැරදියි. `(num-1)*(num-2)`පුනරාවර්තන නොවන අතර අගයන් දෙකක් පමණක් ගුණ කරයි, සම්පූර්ණ අනුපිළිවෙල නොවේ.

Option / විකල්පය 4)
Incorrect. `num*(num-1)` computes only the product of two consecutive numbers, not the factorial.
වැරදියි. `num*(num-1)` ගණනය කරන්නේ අනුක්‍රමික සංඛ්‍යා දෙකක ගුණිතය පමණි, ක්‍රමාරෝපිතය නොවේ.

Option / විකල්පය 5)
Incorrect. `fact(num)*fact(num-1)` creates multiple recursive calls without correctly calculating the factorial.
වැරදියි. `fact(num)*fact(num-1)` විසින් ක්‍රමාරෝපිතය අගය නිවැරදිව ගණනය නොකර බහු පුනරාවර්තන ඇමතුම් නිර්මාණය කරයි.

The correct answer is 1).
නිවැරදි පිළිතුර 1) වේ.

10. What is the output of below python code?
පහත පයිතන් කේතයේ ප්‍රතිදානය කුමක්ද?

```
count={}
count[(1,2,4)] = 5
count[(4,2,1)] = 7
count[(1,2)] = 6
count[(4,2,1)] = 2
tot = 0
for i in count:
    tot=tot+count[i]
print(len(count)+tot)
```

1) 5
2) 10
3) 16
4) 25
5) Tuples can't be made keys of a dictionary.
Dictionary වල keys වලින් tuples නිර්මාණය කළ නොහැක.

Answer: 3)

Explanation / පැහැදිලි කිරීම:

Tuples are immutable, so they can be used as dictionary keys. The process is as follows:
Tuples වෙනස් කළ නොහැකි බැවින් ඒවා dictionary එකක keys ලෙස භාවිතා කළ හැකිය. ක්‍රියාවලිය පහත පරිදි වේ:

- **Dictionary Creation / Dictionary එකක් නිර්මාණය**

  `count[(1,2,4)]` = 5
  adds the `key (1,2,4)` with value 5 / 5 අගය `key (1,2,4)` එක සමඟ එකතු කරයි.

  `count[(4,2,1)]` = 7
  adds the `key(4,2,1)` with value 7 / 7 අගය `key(4,2,1)` එක සමඟ එකතු කරයි.

  `count[(1,2)]` = 6
  adds the `key(1,2)` with value 6 / 6 අගය `key(1,2)` එක සමඟ එකතු කරයි.

`count[(4,2,1)]` = 2
updates the value of the key (4,2,1) to 2 / key (4,2,1) එක අගය 2 ලෙස යාවත්කාලීන කරයි.

Final dictionary: `{(1,2,4): 5, (4,2,1): 2, (1,2): 6}`

- **Total calculation (tot)**

  Loop iterates through the keys of the dictionary, adding the values:
  ලූපය dictionary එකෙහි keys හරහා පුනරාවර්තනය වන අතර, අගයන් එකතු කරයි:

  `5 (from (1,2,4)) + 2 (from (4,2,1)) + 6 (from (1,2)) = 13`

- **Final Output / අවසාන ප්‍රතිදානය**

  `len(count)` returns the number of keys, which is 3.
  `len(count)` keys ගණන ආපසු ලබා දෙයි, එය 3 වේ.

  Final output / අවසාන ප්‍රතිදානය : `len(count) + tot = 3 + 13 = 16.`

The correct answer is 3).
නිවැරදි පිළිතුර 3) වේ.