සියලු ම හිමිකම් ඇවිරිණි / All Rights Reserved]

Information and Communication Technology
තොරතුරු හා සන්නිවේදන තාක්ෂණය

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

2026 QUIZ NO 292 MARKING

2025/09/28
ictfromabc
Ravindu Bandaranayake

1. Which event is considered by many to mark the beginning of the personal computer revolution?
පුද්ගලික පරිගණක විප්ලවයේ ආරම්භය සනිටුහන් කිරීම සඳහා බොහෝ දෙනා සලකන්නේ කුමන සිදුවීමද?

   1) The release of the Apple Lisa.
   Apple Lisa නිකුත් කිරීම.
   2) The introduction of the IBM PC.
   IBM පරිගණකය හඳුන්වා දීම.
   3) The development of the ARPANET (precursor to the internet).
   ARPANET සංවර්ධනය (අන්තර්ජාලයේ පුරෝගාමියා).
   4) The invention of the microprocessor.
   ක්ෂුද්‍ර සකසනය සොයා ගැනීම.
   5) The creation of the first graphical user interface (GUI).
   පළමු චිත්‍රක පරිශීලක අතුරුමුහුණත (GUI) නිර්මාණය කිරීම.

   Answer: 2)

   Explanation / පැහැදිලි කිරීම:

   The introduction of the IBM PC in 1981 is widely considered the beginning of the personal computer revolution. It brought computing power to individuals and small businesses, setting the stage for widespread adoption of personal computers.
   1981 දී IBM PC එක හඳුන්වාදීම පුද්ගලික පරිගණක විප්ලවයේ ආරම්භය ලෙස පුළුල් ලෙස සැලකේ. එය පුද්ගල සහ කුඩා ව්‍යාපාර සඳහා පරිගණක බලය ගෙන ආ අතර, පුද්ගල පරිගණක පුළුල් ලෙස භාවිතා කිරීම සඳහා වේදිකාව සකස් කළේය.

   Incorrect options / වැරදි විකල්ප:

   1) While the Apple Lisa was significant for its introduction of a graphical user interface (GUI), it did not have the same widespread impact as the IBM PC.
   Apple Lisa එහි චිත්‍රක පරිශීලක අතුරුමුහුණතක් (GUI) හඳුන්වාදීම සඳහා වැදගත් වූ අතර, එය IBM PC මෙන් පුළුල් බලපෑමක් ඇති කළේ නැත.

   3) The ARPANET was a precursor to the internet but focused on research and military purposes, rather than personal computing.
   ARPANET අන්තර්ජාලයේ පූර්වගාමියා වූ නමුත් පුද්ගලික පරිගණනයට වඩා පර්යේෂණ සහ හමුදාමය අරමුණු කෙරෙහි අවධානය යොමු කළේය.

   4) The invention of the microprocessor was crucial for computing but did not directly mark the beginning of the personal computer revolution.
   ක්ෂුද්‍ර සකසනයේ සොයා ගැනීම පරිගණනය සඳහා ඉතා වැදගත් වූ නමුත් පුද්ගලික පරිගණක විප්ලවයේ ආරම්භය සෘජුව සනිටුහන් කළේ නැත.

   5) While important, the creation of the first GUI, demonstrated by the Xerox Alto, did not directly initiate the personal computer revolution.
   මෙය වැදගත් වුවද, Xerox Alto විසින් ප්‍රදර්ශනය කරන ලද පළමු GUI නිර්මාණය පුද්ගලික පරිගණක විප්ලවය සෘජුව ආරම්භ කළේ නැත.

2. The BIOS program activates as soon as the computer is switched on. The BIOS program is located in,
පරිගණකය සක්‍රිය වූ වහාම BIOS වැඩසටහන ක්‍රියාත්මක වේ. BIOS වැඩසටහන පිහිටා ඇත්තේ,

1) Cache Memory / වාරක මතකය.
2) Random Access Memory / පඨන මාත්‍ර මතකය (ROM)
3) CMOS RAM / CMOS මතකය.
4) Northbridge Chipset / නෝර්ත්බ්‍රිජ් චිප්සෙට්.
5) Southbridge Chipset / සවුත්බ්‍රිජ් චිප්සෙට්.

Answer: 2)

Explanation of each option / එක් එක් විකල්පය පැහැදිලි කිරීම:

1) Cache Memory / වාරක මතකය.
Cache memory is a small, high-speed volatile memory located close to the CPU, designed to speed up access to frequently used data and instructions. It is not used for storing the BIOS. Cache memory loses its contents when the computer is powered off, making it unsuitable for storing firmware like the BIOS, which needs to be persistent.
වාරක මතකය යනු නිතර භාවිතා කරන දත්ත සහ උපදෙස් වෙත ප්‍රවේශය වේගවත් කිරීම සඳහා නිර්මාණය කර ඇති CPU වෙත සමීපව පිහිටා ඇති කුඩා, අධිවේගී නශ්‍ය මතකයකි. එය BIOS ගබඩා කිරීම සඳහා භාවිතා නොවේ. පරිගණකය ක්‍රියා විරහිත වූ විට වාරක මතකය එහි අන්තර්ගතය නැති කර ගනී, එය BIOS වැනි ස්ථිරාංග ගබඩා කිරීමට නුසුදුසු වන අතර එය නොනැසී පැවතිය යුතුය.

2) Random Access Memory / පඨන මාත්‍ර මතකය (ROM)
The BIOS (Basic Input/Output System) is a firmware program that is essential for initializing and testing the hardware components of a computer during the booting process. When a computer is powered on, he BIOS is the first program to run, and it is responsible for performing the POST (Power-On Self-Test), initializing hardware components, and loading the operating system from the boot device.
BIOS (මූලික ආදාන/ප්‍රතිදාන පද්ධතිය) යනු පරිගණකයේ ආරම්භ කිරීමේ ක්‍රියාවලියේදී පරිගණකයේ දෘඪාංග සංරචක ආරම්භ කිරීම සහ පරීක්ෂා කිරීම සඳහා අත්‍යවශ්‍ය වන ස්ථිරාංග වැඩසටහනකි. පරිගණකයක් ක්‍රියාත්මක වන විට, BIOS යනු ක්‍රියාත්මක වන පළමු වැඩසටහන වන අතර, එය (Power-On Self-Test) දෘඪාංග සංරචක ආරම්භ කිරීම සහ ඇරඹුම් උපාංගයෙන් මෙහෙයුම් පද්ධතිය පූරණය සඳහා වගකීම දරයි.

The BIOS is stored in Read-Only Memory (ROM). ROM is a type of non-volatile memory, meaning that its contents are retained even when the computer is powered off. This makes ROM an ideal location for storing firmware like the BIOS, as it ensures that the BIOS program is always available to execute when the computer starts up. Traditionally, ROM chips were used to store the BIOS, but in modern computers, EEPROM (Electrically Erasable Programmable Read-Only Memory) or flash memory, which are types of ROM, are commonly used. Regardless of the specific type, the BIOS is fundamentally stored in ROM.
BIOS ගබඩා කර ඇත්තේ කියවීමට පමණක් වන මතකය ROM තුළය. ROM යනු නශ්‍ය නොවන මතක වර්ගයකි, එනම් පරිගණකය ක්‍රියා විරහිත වූ විට පවා එහි අන්තර්ගතය රඳවා තබා ගනී. මෙය BIOS වැනි ස්ථිරාංග ගබඩා කිරීම සඳහා ROM සුදුසු ස්ථානයක් බවට පත් කරයි, එය පරිගණකය ආරම්භ වන විට BIOS වැඩසටහන ක්‍රියාත්මක කිරීමට සැමවිටම පවතින බව සහතික කරයි. සම්ප්‍රදායිකව, BIOS ගබඩා කිරීම සඳහා ROM චිප් භාවිතා කරන ලද, නමුත් නවීන පරිගණකවල, ROM වර්ග වන EEPROM (විද්‍යුත් වශයෙන් මකා දැමිය හැකි වැඩසටහන්ගත කළ හැකි කියවීමට පමණක් මතකය) හෝ ෆ්ලෑෂ් මතකය බහුලව භාවිතා වේ. වර්ගය කුමක් වුවත්, BIOS මූලික වශයෙන් ඝෘඩව් හි ගබඩා කර ඇත.

3) CMOS RAM
CMOS RAM (Complementary Metal-Oxide-Semiconductor RAM) is a small amount of memory used to store configuration settings for the BIOS, such as system time and hardware settings. CMOS RAM is powered by a small battery to retain its data when the computer is off, but it does not store the BIOS program itself. Instead, it stores the settings that the BIOS reads during initialization.
CMOS RAM (Complementary Metal-Oxide-Semiconductor RAM) යනු පද්ධති කාලය සහ දෘඪාංග සැකසුම් වැනි BIOS සඳහා වින්‍යාස සැකසුම් ගබඩා කිරීමට භාවිතා කරන කුඩා මතක ප්‍රමාණයකි. CMOS RAM පරිගණකය ක්‍රියා විරහිත වූ විට එහි දත්ත රඳවා ගැනීමට කුඩා බැටරියකින් බල ගැන්වෙන නමුත් එය BIOS වැඩසටහනම ගබඩා නොකරයි. ඒ වෙනුවට, එය ආරම්භයේදී BIOS කියවන සැකසුම් ගබඩා කරයි.

4) Northbridge Chipset
The Northbridge chipset is part of the motherboard's chipset and is responsible for handling communications between the CPU, RAM, and high-speed peripherals such as the graphics card. It does not store the BIOS program. The Northbridge plays a critical role in performance but is not involved in storing firmware.
Northbridge චිප්සෙට් එක මවුපුවරුවේ චිප්සෙට් එකේ කොටසක් වන අතර CPU, RAM සහ ග්‍රැෆික් කාඩ් වැනි අධිවේගී උපාංග අතර සන්නිවේදනය හැසිරවීමේ වගකීම දරයි. එය BIOS වැඩසටහන ගබඩා නොකරයි. Northbridge කාර්ය සාධනය සඳහා තීරණාත්මක කාර්යභාරයක් ඉටු කරන නමුත් ස්ථිරාංග ගබඩා කිරීමට සම්බන්ධ නොවේ.

5) Southbridge Chipset
The Southbridge chipset handles lower-speed peripheral communications, such as USB, audio, and storage interfaces. Like the Northbridge, it does not store the BIOS. The Southbridge works in conjunction with the Northbridge to manage the flow of data between the CPU and peripheral devices but is not related to firmware storage.
සවුත්බ්‍රිජ් චිප්සෙට් එක USB, ශ්‍රව්‍ය සහ ගබඩා අතුරුමුහුණත් වැනි අඩු-වේග පර්යන්ත සන්නිවේදනයන් හසුරුවයි. Northbridge මෙන්ම එය BIOS ගබඩා නොකරයි. CPU සහ පර්යන්ත උපාංග අතර දත්ත ප්‍රවාහය කළමනාකරණය කිරීම සඳහා Southbridge, Northbridge සමඟ එක්ව ක්‍රියා කරන නමුත් ස්ථිරාංග ගබඩාවට සම්බන්ධ නොවේ.

So, the BIOS is stored in ROM because ROM provides the necessary non-volatile storage that ensures the BIOS is always available upon powering on the computer. The other options (cache memory, CMOS RAM, Northbridge, and Southbridge chipsets) serve different roles in the computer's architecture and are not suitable for storing the BIOS firmware.
එබැවින්, BIOS, ROM හි ගබඩා කර ඇත්තේ ROM මඟින් අවශ්‍ය නශ්‍ය නොවන ආචයනය සපයන නිසා පරිගණකයේ බල ගැන්වීමේදී BIOS සැමවිටම පවතින බව සහතික කරයි. අනෙකුත් විකල්ප (cache memory, CMOS RAM, Northbridge, and Southbridge chipsets) පරිගණක ආකෘතියේ විවිධ භූමිකාවන් ඉටු කරන අතර BIOS ස්ථිරාංග ගබඩා කිරීම සඳහා සුදුසු නොවේ.

3. Given the 8-bit 2's complement binary number 1110 1101, what is its decimal equivalent?
8 bit (2) හි අනුපූරක ද්විමය අංකය (1110) (1101) ලබා දී ඇති අතර, එහි දශම සමානය කුමක්ද?

1) -19 2) -35 3) -29 4) -27 5) -39

Answer: 1)

To find the decimal equivalent of the 8-bit 2's complement binary number 11101101, follow these steps:
8-bit 2 හි අනුපූරක ද්විමය අංක 11101101 හි දශම තුල්‍ය සොයා ගැනීමට, මෙම පියවර අනුගමනය කරන්න.

(1) **Identify if it's a negative number / එය සෘණ අංකයක් දැයි හඳුනා ගැනීම.**

The leftmost bit (1) indicates that it's a negative number since it's 1 in 2's complement.
වම්පස ඇති බිටුව (1) පෙන්නුම් කරන්නේ එය 2 හි අනුපූරකයේ 1 වන බැවින් එය සෘණ අංකයක් බවයි.

(2) **Find the 2's complement / 2 හි අනුපූරකය සොයන්න.**

- Invert the bits: 11101101 becomes 00010010.
  බිටු ප්‍රතිලෝමය: 11101101 = 00010010 බවට පත් වේ.
- Add 1 to this inverted result: 00010010 + 1 = 00010011 (which is 19 in decimal).
  මෙම ප්‍රතිලෝම ප්‍රතිඵලයට 1 ක් එකතු කරන්න: 00010010 + 1 = 00010011 (එය දශමයෙන් 19 කි).

(3) **Make it negative / එය සෘණ අගයක් කරන්න**

Since it was a negative number originally, the decimal equivalent is -19.
එය මුලින් සෘණ අංකයක් වූ බැවින්, දශම තුල්‍ය වන්නේ -19 වේ.

So, 11101101 in 2's complement is -19 in decimal.
එබැවින්, 11101101 හි 2 හි අනුපූරකය දශමයෙන් -19 වේ.

4. Which of the following answer contains respective SOP and POS expressions of the following expression?
පහත සඳහන් ප්‍රකාශනයට අදාළ SOP සහ POS ප්‍රකාශන අඩංගු වන්නේ පහත කුමන පිළිතුරෙහිද?

$\mathrm{CB} + \overline{(\overline{\mathrm{DD}} + \overline{0\mathrm{C}})\mathrm{D}}$

1) $\underline{\mathrm{CB} + \overline{\mathrm{D}}, (\overline{\mathrm{D}} + \mathrm{C})(\overline{\mathrm{D}} + \mathrm{B})}$
2) $\mathrm{CB} + \overline{\mathrm{CD}}, (\overline{\mathrm{D}} + \mathrm{C})(\overline{\mathrm{D}} + \mathrm{B})$
3) $\mathrm{CD} + \overline{\mathrm{B}}, (\overline{\mathrm{D}} + \mathrm{C})(\overline{\mathrm{D}} + \mathrm{B})$
4) $\mathrm{CB} + \overline{\mathrm{D}}, (\overline{\mathrm{D}} + \mathrm{C})(\overline{\mathrm{B}} + \mathrm{C})$
5) $\mathrm{CD} + \overline{\mathrm{B}}, (\overline{\mathrm{D}})(\overline{\mathrm{D}} + \mathrm{B})$

Answer: 1)

Let's simply the above expression by using Boolean laws.
බූලියානු නීති භාවිතයෙන් ඉහත ප්‍රකාශනය සරල කරමු.

$= \mathrm{CB} + \overline{(\overline{\mathrm{DD}} + \overline{0\mathrm{C}})\mathrm{D}}$	
$= \mathrm{CB} + \overline{\overline{\mathrm{DD}} + \overline{0\mathrm{C}}} + \overline{\mathrm{D}}$	De Morgan's law ද මුවාර් ප්‍රමේයය
$= \mathrm{CB} + \overline{\overline{\mathrm{DD}}}\ \overline{\overline{0\mathrm{C}}} + \overline{\mathrm{D}}$	De Morgan's law ද මුවාර් ප්‍රමේයය
$= \mathrm{CB} + \mathrm{DD}\ \overline{\overline{0\mathrm{C}}} + \overline{\mathrm{D}}$	Double complement law ද්විත්ව ප්‍රතිලෝම න්‍යාය
$= \mathrm{CB} + \mathrm{DD}\ 0\mathrm{C} + \overline{\mathrm{D}}$	Double complement law ද්විත්ව ප්‍රතිලෝම න්‍යාය
$= \mathrm{CB} + 0 + \overline{\mathrm{D}}$	Identity law සර්වසාම්‍ය න්‍යාය
$= \mathrm{CB} + \overline{\mathrm{D}}$	Identity law සර්වසාම්‍ය න්‍යාය

Now we convert above SOP expression into POS Expression.
දැන් අපි ඉහත SOP ප්‍රකාශනය POS ප්‍රකාශනය බවට පරිවර්තනය කරමු.

$= \overline{\mathrm{CB} + \overline{\mathrm{D}}}$	Taking inverse of the equation ප්‍රකාශනයේ ප්‍රතිලෝමය ගැනීම
$= (\overline{\mathrm{CB}})\overline{\overline{\mathrm{D}}}$	De Morgan's law ද මුවාර් ප්‍රමේයය
$= (\overline{\mathrm{CB}})\mathrm{D}$	Double complement law ද්විත්ව ප්‍රතිලෝම න්‍යාය
$= (\overline{\mathrm{C}} + \overline{\mathrm{B}})\mathrm{D}$	De Morgan's law ද මුවාර් ප්‍රමේයය
$= \mathrm{D}\overline{\mathrm{C}} + \overline{\mathrm{B}}\mathrm{D}$	Distributive law විඝටන න්‍යාය
$= \overline{\mathrm{D}\overline{\mathrm{C}} + \overline{\mathrm{B}}\mathrm{D}}$	Taking inverse of the equation ප්‍රකාශනයේ ප්‍රතිලෝමය ගැනීම
$= \overline{(\mathrm{D}\overline{\mathrm{C}})}\ \overline{(\overline{\mathrm{B}}\mathrm{D})}$	De Morgan's law ද මුවාර් ප්‍රමේයය
$= (\overline{\mathrm{D}} + \overline{\overline{\mathrm{C}}})\ (\overline{\overline{\mathrm{B}}} + \overline{\mathrm{D}})$	De Morgan's law ද මුවාර් ප්‍රමේයය
$= (\overline{\mathrm{D}} + \mathrm{C})\ (\mathrm{B} + \overline{\mathrm{D}})$	Double complement law ද්විත්ව ප්‍රතිලෝම න්‍යාය

So, the correct answer is 1.
එබැවින්ල නිවැරදි පිළිතුර 1 වේ.

5. Which real-life application best exemplifies the function of a combinational circuit ?
සංයෝජන පරිපථයක ක්‍රියාකාරිත්වය වඩාත් හොඳින් නිරූපණය කරන සැබෑ ජීවිත යෙදුම කුමක්ද ?

1) <u>Traffic light controller / මාර්ග සංඥා පාලකය.</u>
2) Water filtration system / ජල පෙරීමේ පද්ධතිය.
3) Digital counter / අංකිත ගණකය.
4) Solar panels / සූර්ය පැනල.
5) Microwave oven / ක්ෂුද්‍ර තරංග උදුන.

Answer: 1)

Combinational circuits are digital circuits where the output is determined solely by the current inputs, without any dependence on previous inputs or states. These circuits perform operations based on a combination of inputs and produce immediate outputs.
සංයෝජන පරිපථ යනු අංකිත පරිපථ වන අතර එහිදී ප්‍රතිදානය පෙර යෙදවුම් හෝ ප්‍රාන්ත මත යැපීමෙන් තොරව වත්මන් යෙදවුම් මගින් පමණක් තීරණය වේ. මෙම පරිපථ ආදාන සංයෝගයක් මත පදනම්ව මෙහෙයුම් සිදු කරන අතර ක්ෂණික ප්‍රතිදානයන් නිපදවයි.

An ideal real-life example of a combinational circuit is the Arithmetic Logic Unit (ALU) within a CPU. The ALU performs a variety of arithmetic and logical operations, such as addition, subtraction, and logical comparisons, based on the current input data. The result is produced instantaneously, exemplifying the essence of a combinational circuit where outputs change directly with the inputs.
සංයුක්ත පරිපථයක සැබෑ උදාහරණය වන්නේ CPU තුළ ඇති අංක ගණිතමය තාර්කික ඒකකය ALU ය. ALU වත්මන් ආදාන දත්ත මත පදනම්ව එකතු කිරීම, අඩු කිරීම සහ තාර්කික සැසඳීම් වැනි විවිධ අංක ගණිතමය සහ තාර්කික මෙහෙයුම් සිදු කරයි. ප්‍රතිපලය ක්ෂණිකව නිපදවනු ලැබේ, ප්‍රතිදානයන් යෙදවුම් සමඟ සෘජුවම වෙනස් වන සංයෝජන පරිපථයක සාරය විදහා දක්වයි.

1) Traffic Light Controller / රථවාහන ආලෝක පාලකය
A traffic light controller requires a sequence of operations where the state of the lights (red, green, yellow) depends on both current inputs (like sensor signals) and the previous state of the system. This reliance on previous states and the need to transition through a sequence of states makes it a sequential circuit rather than a combinational one.
රථවාහන ආලෝක පාලකයට ක්‍රියා අනුපිළිවෙලක් අවශ්‍ය වන අතර එහිදී විදුලි පහන් වල තත්වය (රතු, කොළ, කහ) වත්මන් යෙදවුම් (සංවේදක සංඥා වැනි) සහ පද්ධතියේ පෙර තත්ත්වය යන දෙකම මත රඳා පවතී. පෙර පැවති තත්වයන් මත මෙම රඳා පැවතීම සහ ප්‍රාන්තවල අනුපිළිවෙලක් හරහා සංක්‍රමණය වීමේ අවශ්‍යතාවය එය සංයෝජන එකකට වඩා අනුක්‍රමික පරිපථයක් බවට පත් කරයි.

2) Water Filtration System / ජල පෙරීමේ පද්ධතිය
Water filtration systems involve continuous monitoring and control processes, which may include adjusting filtration stages based on sensor inputs over time. These processes are more aligned with sequential circuits because they require maintaining and updating states based on ongoing inputs and system conditions.
ජල පෙරීමේ පද්ධති අඛණ්ඩ අධීක්ෂණ සහ පාලන ක්‍රියාවලින් ඇතුළත් වන අතර, කාලයත් සමඟ සංවේදක යෙදවුම් මත පදනම්ව පෙරීමේ අදියර සකස් කිරීම ඇතුළත් විය හැකිය. මෙම ක්‍රියාවලි අනුක්‍රමික පරිපථ සමඟ වඩාත් සමපාත වේ, මන්ද ඒවාට අඛණ්ඩ යෙදවුම් සහ පද්ධති තත්ව මත පදනම්ව තත්වයන් නඩත්තු කිරීම සහ යාවත්කාලීන කිරීම අවශ්‍ය වේ.

3) Digital Counter / අංකිත ගණකය
A digital counter keeps track of the number of pulses or events that occur, updating its count based on the input signals. This process inherently involves memory, as the current count must be stored and updated with each new pulse. Hence, digital counters are sequential circuits due to their dependence on prior inputs and states.
අංකිත ගණකයක් ආදාන සංඥා මත පදනම්ව එහි ගණන යාවත්කාලීන කරමින්, සිදුවන ස්පන්දන හෝ සිදුවීම් ගණන නිරීක්ෂණය කරයි. මෙම ක්‍රියාවලිය සඳහා මතකය සම්බන්ධ වේ, මන්ද වත්මන් ගණන එක් එක් නව ස්පන්දනය සමඟ ගබඩා කර යාවත්කාලීන කළ යුතුය. එබැවින්, අංකිත ගණකය පූර්ව යෙදවුම් සහ තත්වයන් මත යැපීම හේතුවෙන් අනුක්‍රමික පරිපථයක් වේ.

4) Solar Panels / සූර්ය පැනල
Solar panels convert sunlight into electrical energy through photovoltaic cells, a process that does not involve digital logic circuits directly. If any digital control systems are used (such as for tracking the sun's position), these systems would likely be part of a broader control mechanism involving sequential logic to monitor and adjust positions based on feedback over time.
සූර්ය පැනල ප්‍රකාශ වෝල්ටීයතා සෛල හරහා සූර්යාලෝකය විද්‍යුත් ශක්තිය බවට පරිවර්තනය කරයි, එය අංකිත තාර්කික පරිපථ සෘජුව සම්බන්ධ නොවන ක්‍රියාවලියකි. කිසියම් අංකිත පාලන පද්ධති භාවිතා කරන්නේ නම් (සූර්යයාගේ පිහිටීම නිරීක්ෂණය කිරීම වැනි), මෙම පද්ධති කාලයත් සමඟ ප්‍රතිපෝෂණ මත පදනම්ව ස්ථාන නිරීක්ෂණය කිරීම සහ සකස් කිරීම සඳහා අනුක්‍රමික තර්කනය ඇතුළත් පුළුල් පාලන යාන්ත්‍රණයක කොටසක් විය හැකිය.

5) Microwave Oven / ක්ෂුද්‍ර තරංග උදුන
A microwave oven incorporates digital controls to set cooking times and power levels. However, its operation involves executing timed sequences and adjusting based on user inputs and internal states. This use of timing and state-based control classifies it as a sequential circuit.
මයික්‍රෝවේව් උදුනක පිසීමේ වේලාවන් සහ බල මට්ටම් සැකසීමට අංකිත පාලන ඇතුළත් වේ. කෙසේ වෙතත්, එහි ක්‍රියාකාරිත්වය කාලානුරූපී අනුපිළිවෙල ක්‍රියාත්මක කිරීම සහ පරිශීලක යෙදවුම් සහ අභ්‍යන්තර තත්වයන් මත පදනම්ව සකස් කිරීම ඇතුළත් වේ. මෙම කාල නිර්ණය සහ තත්ව පාදක පාලනය භාවිතා කිරීම අනුක්‍රමික පරිපථයක් ලෙස වර්ගීකරණය කරයි.

While the provided options each incorporate some form of digital control, they all involve elements of sequential logic, making them unsuitable as examples of pure combinational circuits. The ALU in a CPU remains the quintessential real-life example of a combinational circuit, performing immediate operations based on current inputs without any reliance on previous states.
ලබා දී ඇති විකල්පයන් එක් එක් ආකාරයේ අංකිත පාලනයක් ඇතුළත් වන අතර, ඒවා සියල්ලම අනුක්‍රමික තාර්කික මූලාංග ඇතුළත් වන අතර ඒවා සංයෝජන පරිපථවල උදාහරණ ලෙස නුසුදුසු වේ. CPU එකක ALU සංයෝජක පරිපථයක සැබෑ උදාහරණය ලෙස පවතී, පෙර තත්ත් වයන් මත කිසිදු විශ්වාසයකින් තොරව වත්මන් යෙදවුම් මත පදනම්ව ක්ෂණික මෙහෙයුම් සිදු කරයි.

So, the answer is 1).
ඒ අනුව, පිළිතුර වන්නේ 1) වේ.

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

```
set1 = {1, 3, 5}
set2 = {2, 4, 6}
print(len(set1 + set2))
```

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

Answer: 5)

Sets in Python / Python හි Sets:
Sets are unordered collections of unique elements. They are defined using curly braces `{}`. Operations that can be performed on sets include union, intersection, difference, etc., but not direct addition.
Sets යනු unique elements වල unordered collection එකකි. ඒවා curly braces `{}` භාවිතයෙන් අර්ථ දක්වා ඇත. Sets මත සිදු කළ හැකි මෙහෙයුම් වලට union, intersection, difference ආදිය ඇතුළත් වේ, නමුත් direct addition කිරීමක් නොවේ.

The + operator with sets / Sets සහිත + ක්‍රියාකරු:
The `+` operator is not defined for sets in Python. Attempting to use the `+` operator with sets will raise a `TypeError`.
Python හි sets සඳහා `+` ක්‍රියාකරු (operator) අර්ථ දක්වා නැත. `+` ක්‍රියාකරු (operator) sets සමඟ භාවිතා කිරීමට උත්සාහ කිරීම `TypeError` එකක් මතු කරයි.

When the code `print(len(set1 + set2))` is executed, Python will raise a `TypeError` because you cannot use the `+` operator to concatenate or add sets. The correct way to combine sets would be to use the union method or the `|` operator.
කේතය `print(len(set1 + set2))` ක්‍රියාත්මක කළ විට, ඔබට sets එකතු කිරීමට හෝ එකතු කිරීමට `+` ක්‍රියාකරු භාවිත කළ නොහැකි නිසා `TypeError` එකක් මතු කරයි. sets ඒකාබද්ධ කිරීමේ නිවැරදි ක්‍රමය වනුයේ යුනියන් ක්‍රමය හෝ `|` ක්‍රියාකරු (operator) භාවිතා කිරීමයි.

Therefore, the correct answer for the output of the given code is 5) Error.
ලබා දී ඇති කේතයේ ප්‍රතිදානය සඳහා නිවැරදි පිළිතුර වන්නේ 5) Error.

Correct way to combine sets / Sets ඒකාබද්ධ කිරීමට නිවැරදි මාර්ගය:
If you wanted to combine the sets and then find the length of the combined set, you could do:
ඔබට sets ඒකාබද්ධ කර ඒකාබද්ධ setඑකේ දිග සොයා ගැනීමට අවශ්‍ය නම්, ඔබට කළ හැක්කේ:

```
set1 = {1, 3, 5}
set2 = {2, 4, 6}
combined_set = set1 | set2  # or set1.union(set2)
print(len(combined_set))  # This would output 6
```

7. Which of the following code blocks will print Hello?
Hello මුද්‍රණය කරනු ලබන්නේ පහත සඳහන් කුමන කේත බණ්ඩය මගින් ද?

(A)
```
num = 10
if num >= 5:
    print("Hello")
elif num <= 15:
    print("World")
```

(B)
```
num = 10
if num >= 5:
    print("Hello")
else:
    if num <= 15:
        print("World")
```

(C)
```
num = 10
if num >= 5:
    print("Hello")
elif num <= 15:
    print("World")
else:
    print ("!")
```

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) All of the above. ඉහත සියල්ලම.

Answer: 5)

Option / විකල්පය (A):
num = 10. The first condition num >= 5 is True, so it will print "Hello".
num = 10. පළමු කොන්දේසිය num >= 5 සත්‍ය වේ. එබැවින්, එය "Hello" ලෙස මුද්‍රණය කරනු ඇත.

Option / විකල්පය (B):
num = 10. The first condition num >= 5 is True, so it will print "Hello". The else block will not execute because the first condition is already satisfied.
num = 10. පළමු කොන්දේසිය num >= 5 සත්‍ය වේ. එබැවින්, එය "Hello" ලෙස මුද්‍රණය කරනු ඇත. පළමු කොන්දේසිය දැනටමත් තෘප්තිමත් බැවින් else කොටස ක්‍රියාත්මක නොවේ.

Option / විකල්පය (C):
num = 10. The first condition num >= 5 is True, so it will print "Hello".
num = 10. පළමු කොන්දේසිය num >= 5 සත්‍ය වේ. එබැවින්, එය "Hello" ලෙස මුද්‍රණය කරනු ඇත.

All three code blocks will print **"Hello"** because the condition num >= 5 is true in each case. Therefore, the correct answer is 5).
කේත බ්ලොක් තුනම **"Hello"** ලෙස මුද්‍රණය කරනු ඇත, මන්ද කොන්දේසිය num >= 5 සෑම අවස්ථාවකදීම සත්‍ය වේ. එබැවින්, නිවැරදි පිළිතුර 5) වේ.

8. Which of the following data type is immutable in Python?
පයිතන් හි වෙනස් කළ නොහැකි දත්ත වර්ගයක් වන්නේ පහත සඳහන් කුමක්ද?

1) List
2) Dictionary
3) Set
4) Tuple
5) None above this. ඉහත කිසිවක් නොවේ.

Answer: 4)

Python has both mutable and immutable data types. List, Dictionary, and Set are all mutable, meaning their elements can be changed, added, or removed after the object is created. However, a Tuple is immutable, meaning that once a tuple is created, its values cannot be altered. The elements within the tuple remain fixed. None of the above implies that none of the options are immutable, but in this case, the Tuple is indeed immutable.
පයිතන් සතුව විකෘති සහ වෙනස් කළ නොහැකි දත්ත වර්ග දෙකම ඇත. ලැයිස්තුව, ශබ්දකෝෂය සහ කට්ටලය යන සියල්ලම විකෘති වේ, එනම් වස්තුව නිර්මාණය කිරීමෙන් පසු ඒවායේ මූලද්‍රව්‍ය වෙනස් කිරීමට, එකතු කිරීමට හෝ ඉවත් කිරීමට හැකිය. කෙසේ වෙතත්, Tuple එකක් වෙනස් කළ නොහැකි ය, එනම් tuple එකක් නිර්මාණය කළ පසු එහි අගයන් වෙනස් කළ නොහැක. ටුපල් තුළ ඇති මූලද්‍රව්‍ය ස්ථාවරව පවතී. ඉහත කිසිවක් අඟවන්නේ විකල්ප කිසිවක් වෙනස් කළ නොහැකි බවයි, නමුත් මෙම අවස්ථාවේ දී, ඔමවකෑ ඇත්ත වශයෙන්ම වෙනස් කළ නොහැකි ය.

Therefore, the correct answer is option 4) Tuple, as it is the only data type in the list that cannot be modified after creation.
එබැවින්, නිවැරදි පිළිතුර වන්නේ විකල්ප 4) Tuple, එය නිර්මාණය කිරීමෙන් පසු වෙනස් කළ නොහැකි ලැයිස්තුවේ ඇති එකම දත්ත වර්ගය වන බැවිනි.

9. What is the correct Python script to delete a file?
ගොනුවක් මැකීම සඳහා නිවැරදි පයිතන් ක්‍රමලේඛණය කුමක් ද?

1) `del(fp)`
2) `fp.delete()`
3) `os.remove('file')`
4) `os.delete('file')`
5) `fp.delete('fb.txt')`

Answer: 3)

The correct Python script to delete a file is `os.remove('file')`. This function, provided by the '`os`' module, removes the specified file from the filesystem. Unlike options 1 and 2, which do not use correct methods or functions for file deletion, and option 4, which does not exist, `os.remove` directly performs the file deletion operation. Option 5 is incorrect because `fp.delete('fb.txt')` is not a valid Python syntax for file deletion.
ගොනුවක් මකා දැමීමට නිවැරදි Python ස්ක්‍රිප්ට් එක වන්නේ `os.remove('file')`. os මොඩියුලය මඟින් සපයන මෙම ශ්‍රිතය, ගොනු පද්ධතියෙන් නිශ්චිත ගොනුව ඉවත් කරයි. ගොනු මකාදැමීම සඳහා නිවැරදි ක්‍රම හෝ ශ්‍රිත භාවිතා නොකරන විකල්ප 1) සහ 2) සහ නොපවතින විකල්ප 4) මෙන් නොව, `os.remove` ගොනු මකාදැමීමේ මෙහෙයුම සෘජුවම සිදු කරයි. `fp.delete('fb.txt')` ගොනු මකාදැමීම සඳහා වලංගු Python වාක්‍ය ඛණ්ඩයක් නොවන නිසා විකල්ප 5) වැරදියි.

So, the correct answer is 3).
ඒබැවින්, නිවැරදි පිළිතුර 3) වේ.

10. Consider the following python statement.
පහත පයිතන් ප්‍රකාශය සලකා බලන්න.

```
DataList=[52,90,67,99]
for i in DataList:
    if i< 90:
        print(i)
        break
print('end')
```

Which of the following is the correct output?
පහත ඒවායින් නිවැරදි ප්‍රතිදානය කුමක්ද?

1)	2)	3)	4)	5)
52	52	52	52	52
end	90	90	90	end
	67	67	67	90
		end	90	end
			end	99
				end

Answer: 1)

Let's break down the code and explain what happens step by step:
අපි කේතය බිඳ දමා පියවරෙන් පියවර පැහැදිලි කරමු:

`DataList = [52, 90, 67, 99]`
This line creates a list called `DataList` containing the elements `[52, 90, 67, 99]`.
මෙම පේළිය මූලද්‍රව්‍ය `[52, 90, 67, 99]` අඩංගු `DataList` නමින් ලැයිස්තුවක් නිර්මාණය කරයි.

`for i in DataList`
This line starts a loop that will iterate through each element in the DataList. On each iteration, the variable `i` will hold the value of the current element in the list.
මෙම රේඛාව DataList හි එක් එක් මූලද්‍රව්‍ය හරහා පුනරාවර්තනය වන ලූපයක් ආරම්භ කරයි. එක් එක් පුනරාවර්තනයකදී, `i` විචල්‍යය ලැයිස්තුවේ වත්මන් මූලද්‍රව්‍යයේ අගය රඳවා ගනු ඇත.

`if i < 90`

Inside the loop, there's an `if` statement that checks whether the current value of `i` is less than 90.
ලූපය ඇතුළත, `i` හි වත්මන් අගය 90 ට වඩා අඩු දැයි පරීක්ෂා කරන `if` ප්‍රකාශයක් ඇත.

`print(i)`

If the condition `i < 90` is true, the value of `i` is printed
`i < 90` කොන්දේසිය සත්‍ය නම්, `i` අගය මුද්‍රණය කෙරේ.

`break`

After printing the value of i, the break statement immediately exits the loop, meaning no further elements in `DataList` will be checked or processed.
i අගය මුද්‍රණය කිරීමෙන් පසු, break ප්‍රකාශය වහාම ලූපයෙන් පිටවෙයි, එයින් අදහස් වන්නේ `DataList` හි තවත් මූලද්‍රව්‍ය පරීක්ෂා කිරීම හෝ සකසනු නොලැබේ.

`print('end')`

This line is outside the loop and will execute after the loop has finished, printing the word `'end'`
මෙම පේළිය ලූපයෙන් පිටත වන අතර ලූපය අවසන් වූ පසු `'end'` යන වචනය මුද්‍රණය කිරීමෙන් ක්‍රියාත්මක වේ.

Execution / ක්‍රියාත්මක කිරීම:

On the first iteration, `i` is `52`. The condition `i < 90` is true (since 52 is less than 90), so `52` is printed, and the `break` statement is triggered, causing the loop to exit. The loop stops after printing `52`, and no further elements (90, 67, 99) are checked or printed. After the loop exits, `end` is printed.
පළමු පුනරාවර්තනයේ දී, `i 52` වේ. `i < 90` කොන්දේසිය සත්‍ය වේ (52 90 ට වඩා අඩු බැවින්), එබැවින් `52` මුද්‍රණය කර ඇති අතර, `break` ප්‍රකාශය ක්‍රියා විරහිත වේ, ලූපය පිටවීමට හේතු වේ. `52` මුද්‍රණය කිරීමෙන් පසු ලූපය නතර වේ, සහ තවත් මූලද්‍රව්‍ය (90, 67, 99) පරීක්ෂා කර හෝ මුද්‍රණය නොකෙරේ. ලූපය පිටවීමෙන් පසු, `end` මුද්‍රණය වේ.

So, the output will be / එබැවින්, ප්‍රතිදානය,

```
52
end
```

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