G.C.E. O/L ICT · 10 ශ්‍රේණිය

පාඩම 3: පරිගණක පද්ධතිවල දත්ත නිරූපණය

- 3.1 Binary – අවස්ථා දෙකකින් දත්ත නිරූපණය
- 3.2 දශම සිට Binary, Octal සහ Hexadecimal
- 3.3 Binary ↔ දශම, Octal සහ Hexadecimal හැරවීම්
- 3.4 දත්ත ගබඩා ඒකක – Bit, Byte, KB, MB, GB, TB
- 3.5 කේතන පද්ධති – BCD, EBCDIC, ASCII සහ Unicode

© 2026 schoolict.net (Ruwan Mangala Suraweera)
සියලු හිමිකම් ඇවිරිණි.

G.C.E. O/L ICT · 10 ශ්‍රේණිය · පාඩම 3

3.1 Binary – අවස්ථා දෙකකින් දත්ත නිරූපණය

පරිගණකයක් ඇතුළේ හැම දෙයක්ම – අංක, අකුරු, ඡායාරූප, සින්දු, වීඩියෝ – ගබඩා කරන්නේ සංකේත දෙකකින් පමණයි: **1 සහ 0**. පරිගණක අවස්ථා දෙකක් භාවිතා කරන්නේ ඇයි, **ද්විමය (binary) සංඛ්‍යා පද්ධතිය** කුමක්ද, ධන දශම සංඛ්‍යාවක් binary වලින් ලියන හැටි මේ සටහනෙන් ඉගෙන ගමු.

1. දත්ත සහ අවස්ථා දෙක

පරිගණකයක් සෑදී ඇත්තේ **transistors** නම් කුඩා ඉලෙක්ට්‍රොනික ස්විච මිලියන (බිලියන) ගණනකින්. එක් ස්විචයකට තිබිය හැක්කේ **අවස්ථා දෙකක්** පමණයි: ON හෝ OFF. ON = **1**, OFF = **0** ලෙස ලියනවා. අවස්ථා දෙකක් පමණක් ඇති නිසා ඒවා ගබඩා කිරීම සහ යැවීම පහසුයි, විශ්වාසදායකයි.

අවස්ථාව 1	අවස්ථාව 0	දකින තැන
ON	OFF	ස්විචය / transistor
TRUE / ඔව්	FALSE / නැත	තර්කය, තීරණ
ඉහළ වෝල්ටීයතාව	අඩු වෝල්ටීයතාව	පරිපථ, වයර්
එක් දිශාවකට චුම්බක	අනෙක් දිශාවට චුම්බක	Hard disk
ආලෝකය / pit	වෙනසක් නැත	CD / DVD
ආරෝපණය ඇත	ආරෝපණය නැත	SSD, USB flash, memory card

💡 0 සහ 1 වෙන් වෙන් අගය ලෙස ගබඩා වූ දත්ත = **digital** දත්ත. තනි 0 හෝ 1 කට **bit** (binary digit) කියනවා.

2. පරිගණක binary භාවිතා කරන්නේ ඇයි?

- ඉලෙක්ට්‍රොනික ස්විචවලට ඇත්තේ අවස්ථා දෙකයි – ON/OFF හොඳින්ම ගැළපේ.
- අවස්ථා දෙක වෙන් කර හඳුනා ගැනීම පහසුයි, ඒ නිසා දෝෂ අඩුයි.
- පරිපථ සරලයි, ලාභයි, වේගවත්.
- හැම දත්ත වර්ගයක්ම (පෙළ, රූප, ශබ්ද, වීඩියෝ) 1 සහ 0 රටා බවට හැරවිය හැක.

3. ද්විමය (binary) සංඛ්‍යා පද්ධතිය

අපි සාමාන්‍යයෙන් ඉලක්කම් දහයක් (0–9) ඇති **දශම** (base 10) පද්ධතිය භාවිතා කරනවා. **Binary** පද්ධතියේ **පාදය 2** යි; ඉලක්කම් දෙකයි: **0 සහ 1**. සෑම ස්ථානයක්ම දකුණු පස ස්ථානය මෙන් **2 ගුණයක්** වටිනවා: 1, 2, 4, 8, 16, 32, 64, 128 …

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1

සංඛ්‍යාව කුමන පද්ධතියේද පෙන්වීමට පාදය කුඩා අංකයක් ලෙස ලියන්න: 1101_2 (binary), 13_{10} (දශම). පාදය නැතිනම් 10 කියන්නේ දහයද දෙකද කියා නොදැනේ!

4. ධන දශම නිඛිලයක් binary වලින් ලිවීම

ක්‍රමය 1 – ස්ථානීය අගය: ගැළපෙන විශාලතම ස්ථානීය අගය සොයා එහි 1 දමා අඩු කර දිගටම කරන්න. උදා: 13 = 8 + 4 + 1.

8	4	2	1
1	1	0	1

💡 එනම් 13_{10} = **1101_2**. පරීක්ෂාව: 8 + 4 + 0 + 1 = 13.

ක්‍රමය 2 – 2න් නැවත නැවත බෙදීම: 2න් බෙදා ශේෂය ලියන්න; ලබ්ධිය 0 වන තුරු බෙදන්න. ශේෂ **පහළ සිට ඉහළට** කියවන්න. උදා: 25_{10}.

බෙදීම	ලබ්ධිය	ශේෂය
25 ÷ 2	12	1
12 ÷ 2	6	0
6 ÷ 2	3	0
3 ÷ 2	1	1
1 ÷ 2	0	1

💡 ශේෂ පහළ සිට ඉහළට: 25_{10} = **11001_2**. (16 + 8 + 1 = 25)

5. bit n කට අගයන් කීයක් නිරූපණය කළ හැකිද?

bit n කින් වෙනස් රටා **2^n** ක් සෑදිය හැක. bit 1 → 2, bit 2 → 4, bit 3 → 8, bit 8 → 256 (0 සිට 255). bit n කින් ලිවිය හැකි විශාලතම අංකය **$2^n - 1$**.

Bits (n)	රටා (2^n)	පරාසය
1	2	0 – 1
2	4	0 – 3
3	8	0 – 7
4	16	0 – 15
8	256	0 – 255

2026 නව **2026 සම්බන්ධය:** ඔබේ phone එකේ ඡායාරූපයක් කුඩා තිත් (pixels) මිලියන ගණනක්; එක් එක් තිතේ රතු, කොළ, නිල් bits ලෙස ගබඩා වේ. Megapixel 12 ඡායාරූපයක් සම්පීඩනයට පෙර bytes මිලියන 36ක් පමණ!

ප්‍රධාන පද

Binary (ද්විමය) – 0 සහ 1 පමණක් ඇති පාදය 2 සංඛ්‍යා පද්ධතිය.

Bit – තනි binary ඉලක්කමක් (0 හෝ 1) – කුඩාම දත්ත ඒකකය.

අවස්ථා දෙකේ පද්ධතිය – ON/OFF හෝ TRUE/FALSE වැනි අවස්ථා දෙකක් පමණක් ඇති පද්ධතිය.	**Transistor** – Chip එකක් ඇතුළේ ON (1) හෝ OFF (0) වන කුඩා ඉලෙක්ට්‍රොනික ස්විචය.
Decimal (දශම) – 0–9 ඉලක්කම් ඇති පාදය 10 පද්ධතිය.	**පාදය (radix)** – සංඛ්‍යා පද්ධතියක භාවිතා වන වෙනස් ඉලක්කම් ගණන.
ස්ථානීය අගය – ඉලක්කමේ ස්ථානයේ අගය, උදා: binary හි 1, 2, 4, 8.	**නැවත නැවත බෙදීම** – පාදයෙන් නැවත නැවත බෙදා ශේෂ ඉහළට කියවීම.
ශේෂය – බෙදීමෙන් පසු ඉතිරිය; පාදය 2 හි 0 හෝ 1.	**Digital දත්ත** – 0 සහ 1 වැනි වෙන් වූ අගය ලෙස ගබඩා වූ දත්ත.
MSB – Most Significant Bit – ඉහළම ස්ථානීය අගය ඇති වම් කෙළවරේ bit එක.	**LSB** – Least Significant Bit – ස්ථානීය අගය 1 ඇති දකුණු කෙළවරේ bit එක.

ආදර්ශ ප්‍රශ්න සහ පිළිතුරු

1. පරිගණක binary පද්ධතිය භාවිතා කරන්නේ ඇයි?

> **පිළිතුර:** ඉලෙක්ට්‍රොනික ස්විචවලට (transistors) ඇත්තේ ON, OFF අවස්ථා දෙකයි; ඒවා 1 සහ 0 ට ගැළපේ. අවස්ථා දෙක විශ්වාසදායක ලෙස හඳුනාගත හැකි අතර පරිපථ සරලයි.

2. 1 සහ 0 නිරූපණය කළ හැකි අවස්ථා යුගල තුනක් දෙන්න.

> **පිළිතුර:** ON/OFF, TRUE/FALSE, ඉහළ/අඩු වෝල්ටීයතාව (ඔව්/නැත, චුම්බක/නොවේ ද).

3. 13_{10} binary වලට හරවන්න.

> **පිළිතුර:** 13 = 8 + 4 + 1 → 1101_2.

4. Bit එකක් යනු කුමක්ද?

> **පිළිතුර:** Binary digit – තනි 0 හෝ 1; කුඩාම දත්ත ඒකකය.

5. bit 8 කින් වෙනස් අගයන් කීයක් නිරූපණය කළ හැකිද?

> **පිළිතුර:** 2^8 = අගයන් 256 (0 සිට 255).

6. නැවත නැවත බෙදීමේදී ශේෂ කියවන්නේ කුමන අනුපිළිවෙළටද?

> **පිළිතුර:** අවසාන ශේෂයේ (පහළ) සිට පළමු ශේෂය (ඉහළ) දක්වා. අවසාන ශේෂය MSB වේ.

බහුවරණ ප්‍රශ්න

1. Binary සංඛ්‍යා පද්ධතිය ඉලක්කම් කීයක් භාවිතා කරයිද?

A. 2
B. 8
C. 16
D. 10

2. පරිගණකයක භාවිතා වන අවස්ථා දෙක හොඳින්ම විස්තර කරන්නේ?

A. රතු සහ නිල්
B. ON සහ OFF
C. ඉහළ, පහළ, මැද
D. 0 සිට 9

3. තනි binary ඉලක්කමක් හඳුන්වන්නේ:

A. Word
B. Nibble
C. Bit
D. Byte

4. 13_{10} binary වලින් කුමක්ද?

A. 1101
B. 1001
C. 1110
D. 1011

5. 25_{10} binary වලින් කුමක්ද?

A. 11001
B. 10101
C. 11010
D. 10011

6. Binary හි දකුණේ සිට 5 වන bit එකේ ස්ථානීය අගය?

A. 5
B. 32
C. 16
D. 10

7. bit 4 කින් වෙනස් රටා කීයක් සෑදිය හැකිද?

A. 15
B. 16
C. 8
D. 4

8. bit 8 කින් ගබඩා කළ හැකි විශාලතම අංකය:

A. 256
B. 128
C. 8
D. 255

9. 1010_2 ට සමාන දශම අංකය?

A. 12
B. 5
C. 10
D. 1010

10. 2න් නැවත නැවත බෙදීමේදී පිළිතුර කියවන්නේ:

A. ශේෂ ඉහළ සිට පහළට
B. අවසාන ලබ්ධිය පමණයි
C. ලබ්ධි ඉහළ සිට පහළට
D. ශේෂ පහළ සිට ඉහළට

11. Binary ඉලෙක්ට්‍රොනික පරිපථවලට විශ්වාසදායක ඇයි?

A. අවස්ථා දෙකක් පමණක් වෙන් කර හඳුනාගැනීම පහසුයි
B. මිනිසුන් භාවිතා කරන පාදයයි
C. වෝල්ටීයතා මට්ටම් දහයක් භාවිතා කරයි
D. විදුලිය අවශ්‍ය නැත

12. Binary අංකයක් විය නොහැක්කේ කුමක්ද?

A. 1001
B. 1021
C. 10
D. 1111

13. 10110_2 හි MSB ඇති ස්ථානයේ අගය:

A. 32
B. 1
C. 2
D. 16

14. 6_{10} binary වලින් ලිවීමට අවම bit ගණන?

A. 2
B. 3
C. 6
D. 4

ව්‍යුහගත ප්‍රශ්න

1. (i) අවස්ථා දෙකේ පද්ධතියක් යනු කුමක්ද? (ලකුණු 1)
(ii) 1 සහ 0 නිරූපණයට භාවිතා වන අවස්ථා දෙකේ උදාහරණ දෙකක් දෙන්න. (ලකුණු 2) *(ලකුණු 3)*

(i) ඇත්තේ අවස්ථා දෙකක් පමණක් වන පද්ධතියක්.
(ii) ON/OFF, TRUE/FALSE, ඉහළ/අඩු වෝල්ටීයතාව, චුම්බක/නොවේ (ඕනෑම දෙකක්).

2. පරිගණක binary සංඛ්‍යා පද්ධතිය භාවිතා කිරීමට හේතු තුනක් දෙන්න. *(ලකුණු 3)*

Transistors වලට අවස්ථා දෙකයි; අවස්ථා දෙක හඳුනාගැනීම පහසු, දෝෂ අඩුයි; පරිපථ සරල/ලාභ/වේගවත්; හැම දත්ත වර්ගයක්ම 0, 1 ලෙස කේත කළ හැක.

3. නැවත නැවත බෙදීම භාවිතයෙන් 45_{10} binary වලට හරවන්න. ක්‍රියාවලිය පෙන්වන්න. (ලකුණු 3) *(ලකුණු 3)*

45÷2=22 ශේ 1; 22÷2=11 ශේ 0; 11÷2=5 ශේ 1; 5÷2=2 ශේ 1; 2÷2=1 ශේ 0; 1÷2=0 ශේ 1. ඉහළට කියවන්න: $45_{10} = 101101_2$.

4. ස්ථානීය අගය භාවිතයෙන් 100_{10} binary වලින් ලියන්න. (ලකුණු 3) *(ලකුණු 3)*

> 100 = 64 + 32 + 4 → 64 32 16 8 4 2 1 = 1 1 0 0 1 0 0 → 1100100_2.

5. (i) bit 5 කින් වෙනස් අගයන් කීයක් ගබඩා කළ හැකිද? (ලකුණු 1)
(ii) bit 5 කින් ගබඩා කළ හැකි විශාලතම දශම අංකය? (ලකුණු 1)
(iii) එය binary වලින් ලියන්න. (ලකුණු 1) *(ලකුණු 3)*

> (i) 2^5 = 32 (ii) 31 (iii) 11111_2

6. පන්තියක smart bulb එකක් app එකකින් පාලනය වේ. (i) Bulb එකේ ON/OFF තත්ත්වය bit එකක් ලෙස ගබඩා කරන්නේ කෙසේද? (ලකුණු 1) (ii) Bulbs අටක් පේළියක ඇත. ON/OFF රටා කීයක් තිබිය හැකිද? (ලකුණු 2) *(ලකුණු 3)*

> (i) ON = 1, OFF = 0. (ii) එක් bulb = bit 1, bit 8 → 2^8 = රටා 256.

සාරාංශය

- පරිගණක සියලු දත්ත අවස්ථා දෙකකින් ගබඩා කරයි: 1 සහ 0.
- අවස්ථා දෙක: ON/OFF, TRUE/FALSE, ඉහළ/අඩු වෝල්ටීයතාව.
- Bit = එක් binary ඉලක්කමක්; කුඩාම දත්ත ඒකකය.
- Binary පාදය 2; ස්ථානීය අගයන් 1, 2, 4, 8, 16, 32, 64, 128.
- දශම → binary: ස්ථානීය අගය හෝ 2න් නැවත නැවත බෙදීම.
- ශේෂ පහළ සිට ඉහළට කියවන්න.
- bit n → රටා 2^n; විශාලතම අගය $2^n - 1$.
- පාදය කුඩා අංකයෙන් ලියන්න, උදා: $1101_2 = 13_{10}$.

MCQ පිළිතුරු – 3.1

1. **A** – Binary පාදය 2: ඉලක්කම් 0 සහ 1.
2. **B** – ස්විචයක් ON (1) හෝ OFF (0) වේ.
3. **C** – Bit = binary digit.
4. **A** – 13 = 8 + 4 + 1 = 1101_2.
5. **A** – 25 = 16 + 8 + 1 = 11001_2.
6. **C** – ස්ථානීය අගයන් 1, 2, 4, 8, 16 → 2^4 = 16.
7. **B** – 2^4 = 16.
8. **D** – $2^8 - 1$ = 255 (11111111_2).
9. **C** – 8 + 2 = 10.
10. **D** – අවසාන ශේෂය MSB නිසා ඉහළට කියවන්න.
11. **A** – ශබ්ද (noise) තිබුණත් මට්ටම් දෙක හඳුනාගැනීම පහසුයි.
12. **B** – Binary හි ඇත්තේ 0 සහ 1 පමණයි; '2' නොමැත.
13. **D** – bit 5: 16, 8, 4, 2, 1 → MSB = 16.
14. **B** – 6 = 110_2 → bit 3.

© 2026 schoolict.net (Ruwan Mangala Suraweera). සියලු හිමිකම් ඇවිරිණි.

G.C.E. O/L ICT · 10 ශ්‍රේණිය · පාඩම 3

3.2 දශම සිට Binary, Octal සහ Hexadecimal

පරිගණක binary භාවිතා කළත්, 1, 0 දිගු පේළි මිනිසුන්ට කියවීම අපහසුයි. ඒ නිසා programmers **octal (පාදය 8)** සහ **hexadecimal (පාදය 16)** ද භාවිතා කරයි. සංඛ්‍යා පද්ධති හතර, ඒවායේ **ස්ථානීය අගයන්** සහ **නැවත නැවත බෙදීම** මගින් දශම අංකයක් ඒවාට හරවන හැටි මේ සටහනෙන් ඉගෙන ගමු.

1. සංඛ්‍යා පද්ධති හතර

පද්ධතිය	පාදය	භාවිත ඉලක්කම්	උදාහරණය
Binary (ද්විමය)	2	0, 1	1011_2
Octal (අෂ්ටමය)	8	0 – 7	157_8
Decimal (දශම)	10	0 – 9	249_{10}
Hexadecimal (ෂඩ් දශමය)	16	0 – 9, A – F	$3F_{16}$

පාදය පද්ධතියේ ඉලක්කම් ගණන කියයි. විශාලතම ඉලක්කම හැම විටම **පාදය − 1** (binary හි 1, octal හි 7, දශම හි 9, hex හි F).

2. Hexadecimal ඉලක්කම් A–F

Hex වලට තනි ඉලක්කම් 16ක් අවශ්‍ය නිසා 9න් පසු අකුරු භාවිතා කරයි:

Hex	A	B	C	D	E	F
දශම	10	11	12	13	14	15

3. ස්ථානීය අගයන්

හැම පද්ධතියකම දකුණු කෙළවරේ ස්ථානය 1 (පාදය^0). වමට යන සෑම ස්ථානයක්ම **පාදය ගුණයක්** වැඩියි.

පාදය	4th place	3rd place	2nd place	1st place
2	$2^3 = 8$	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$
8	$8^3 = 512$	$8^2 = 64$	$8^1 = 8$	$8^0 = 1$
10	$10^3 = 1000$	$10^2 = 100$	$10^1 = 10$	$10^0 = 1$
16	$16^3 = 4096$	$16^2 = 256$	$16^1 = 16$	$16^0 = 1$

> 💡 උදා: 234_8 හි 2 = 2 × 64, 3 = 3 × 8, 4 = 4 × 1.

4. නැවත නැවත බෙදීමේ ක්‍රමය

- දශම අංකය **නව පාදයෙන්** (2, 8 හෝ 16) බෙදන්න.
- **ශේෂය** ලියන්න.
- **ලබ්ධිය** නැවත බෙදන්න; ලබ්ධිය 0 වන තුරු කරන්න.
- ශේෂ **පහළ සිට ඉහළට** කියවන්න. Hex හි 10–15 → A–F.

උදාහරණය 1: 47_{10} → binary

බෙදීම	ලබ්ධිය	ශේෂය
47 ÷ 2	23	1
23 ÷ 2	11	1
11 ÷ 2	5	1
5 ÷ 2	2	1
2 ÷ 2	1	0
1 ÷ 2	0	1

💡 47_{10} = **101111_{2}**

උදාහරණය 2: 156_{10} → octal

බෙදීම	ලබ්ධිය	ශේෂය
156 ÷ 8	19	4
19 ÷ 8	2	3
2 ÷ 8	0	2

💡 156_{10} = **234_{8}**. පරීක්ෂාව: 2×64 + 3×8 + 4×1 = 128 + 24 + 4 = 156.

ඉලක්කම	2	3	4
ස්ථානීය අගය	8^2 = 64	8^1 = 8	8^0 = 1
ඉලක්කම × ස්ථානීය අගය	2 × 64 = 128	3 × 8 = 24	4 × 1 = 4

උදාහරණය 3: 500_{10} → hexadecimal

බෙදීම	ලබ්ධිය	ශේෂය
500 ÷ 16	31	4
31 ÷ 16	1	15 = F
1 ÷ 16	0	1

💡 පහළ සිට ඉහළට: 1, F, 4 → 500_{10} = **$1F4_{16}$**. පරීක්ෂාව: 1×256 + 15×16 + 4 = 500.

💡 **විභාග උපක්‍රමය:** සෑම ඉලක්කමක්ම ස්ථානීය අගයෙන් ගුණ කර එකතු කර පිළිතුර පරීක්ෂා කරන්න. මුල් අංකය ලැබුණොත් නිවැරදියි!

5. එකම අංකය සැසඳීම

Decimal	Binary	Octal	Hex
8	1000	10	8
10	1010	12	A

15	1111	17	F
16	10000	20	10
64	1000000	100	40
100	1100100	144	64
255	11111111	377	FF

2026 නව **අද hex දකින තැන්:** web වර්ණ (#FF0000 = රතු), Wi-Fi routers වල MAC ලිපින (උදා: 3C:5A:B4:...), IPv6 ලිපින සහ phone/PC දෝෂ කේත.

ප්‍රධාන පද

Octal (අෂ්ටමය) – 0–7 ඉලක්කම් ඇති පාදය 8 පද්ධතිය.

Hexadecimal (ෂඩ් දශමය) – 0–9 සහ A–F ඇති පාදය 16 පද්ධතිය.

පාදය – සංඛ්‍යා පද්ධතියක ඉලක්කම් ගණන (2, 8, 10, 16).

ස්ථානීය අගය – ස්ථානයක අගය: පාදයේ බලයක් (පාදය^0, පාදය^1 ...).

ලබ්ධිය – බෙදීමක පූර්ණ සංඛ්‍යා පිළිතුර.

ශේෂය – බෙදීමෙන් පසු ඉතිරිය; හැම විටම පාදයට වඩා කුඩායි.

නැවත නැවත බෙදීම – ලබ්ධිය 0 වන තුරු නව පාදයෙන් බෙදීමේ ක්‍රමය.

පාද අංකය (subscript) – අගයකට පසු පාදය පෙන්වන කුඩා අංකය, උදා: 75_8.

Hex ඉලක්කම් A–F – Hexadecimal හි 10–15 සඳහා අකුරු.

පාදයේ බලය – පාදය තමා විසින්ම ගුණ කිරීම, උදා: $16^2 = 256$.

Decimal (දශම) – මිනිසුන් දිනපතා භාවිතා කරන පාදය 10 පද්ධතිය.

වර්ණ කේතය – තිරයේ වර්ණයක් සඳහා hex ඉලක්කම් 6ක අගය, උදා: #00FF00.

ආදර්ශ ප්‍රශ්න සහ පිළිතුරු

1. Octal පද්ධතියේ පාදය සහ ඉලක්කම් මොනවාද?

පිළිතුර: පාදය 8; ඉලක්කම් 0–7.

2. Hexadecimal හි A සිට F දක්වා අකුරු නිරූපණය කරන්නේ?

පිළිතුර: A=10, B=11, C=12, D=13, E=14, F=15.

3. 100_{10} octal වලට හරවන්න.

පිළිතුර: 100÷8=12 ශේ 4; 12÷8=1 ශේ 4; 1÷8=0 ශේ 1 → 144_8.

4. 250_{10} hexadecimal වලට හරවන්න.

පිළිතුර: 250÷16=15 ශේ 10(A); 15÷16=0 ශේ 15(F) → FA_{16}.

5. Hexadecimal හි පළමු ස්ථානීය අගයන් හතර?

පිළිතුර: 1, 16, 256, 4096 (16^0 සිට 16^3).

6. Programmers hexadecimal භාවිතා කරන්නේ ඇයි?

> **පිළිතුර:** Binary ට වඩා කෙටියි, හැරවීම පහසුයි (hex ඉලක්කම් 1 = bit 4), උදා: වර්ණ කේත, මතක ලිපින.

බහුවරණ ප්‍රශ්න

1. Hexadecimal පද්ධතියේ පාදය?

A. 6
B. 16
C. 8
D. 10

2. Octal පද්ධතියේ භාවිතා නොවන ඉලක්කම?

A. 0
B. 7
C. 5
D. 8

3. Hexadecimal හි D අකුර නිරූපණය කරන්නේ:

A. 13
B. 11
C. 14
D. 12

4. 156_{10} octal වලින්?

A. 243
B. 234
C. 324
D. 432

5. 250_{10} hexadecimal වලින්?

A. FA
B. AF
C. F10
D. FB

6. 47_{10} binary වලින්?

A. 110111
B. 101101
C. 101111
D. 111101

7. Octal හි දකුණේ සිට 3 වන ඉලක්කමේ ස්ථානීය අගය:

A. 64
B. 512
C. 16
D. 24

8. Hexadecimal හි දකුණේ සිට 3 වන ඉලක්කමේ ස්ථානීය අගය:

A. 32
B. 48
C. 4096
D. 256

9. 255_{10} hexadecimal වලින්?

A. 100
B. FF
C. F5
D. EF

10. 64_{10} octal වලින්?

A. 100
B. 010
C. 80
D. 64

11. 200_{10} hexadecimal වලින්?

A. 8C
B. 128
C. C8
D. C2

12. Hexadecimal වලට හැරවීමේදී ශේෂය 11 ලියන්නේ:

A. 11
B. C
C. B
D. A

13. වලංගු hexadecimal අංකය කුමක්ද?

A. 1H
B. 2E9
C. G5
D. Z0

14. 16_{10} hexadecimal වලින්:

A. 16
B. F
C. G
D. 10

15. #FF0000 web වර්ණය භාවිතා කරන සංඛ්‍යා පද්ධතිය?

A. Octal
B. Binary
C. Hexadecimal
D. රෝම

ව්‍යුහගත ප්‍රශ්න

1. වගුව සම්පූර්ණ කරන්න:
පද්ධතිය | පාදය | ඉලක්කම්
Binary | ? | ?
Octal | ? | ?
Hexadecimal | ? | ? (ලකුණු 4) *(ලකුණු 4)*

> Binary – 2 – 0, 1
> Octal – 8 – 0 සිට 7
> Hexadecimal – 16 – 0 සිට 9 සහ A සිට F

2. නැවත නැවත බෙදීමෙන් 83_{10} octal වලට හරවන්න. ක්‍රියාවලිය පෙන්වන්න. (ලකුණු 3) *(ලකුණු 3)*

> 83÷8=10 ශේ 3; 10÷8=1 ශේ 2; 1÷8=0 ශේ 1 → 123_8. පරීක්ෂාව: 64+16+3=83.

3. 175_{10} hexadecimal වලට හරවන්න. ක්‍රියාවලිය පෙන්වන්න. (ලකුණු 3) *(ලකුණු 3)*

> 175÷16=10 ශේ 15(F); 10÷16=0 ශේ 10(A) → AF_{16}. පරීක්ෂාව: 10×16+15=175.

4. 29_{10} (i) binary (ii) octal (iii) hexadecimal වලට හරවන්න. (ලකුණු 3) *(ලකුණු 3)*

> (i) 11101_2 (ii) 35_8 (iii) $1D_{16}$

5. (i) Octal හි පළමු ස්ථාන හතරේ ස්ථානීය අගයන් ලියන්න. (ලකුණු 2)
(ii) පාදය 8 සහ 16 හි විශාලතම තනි ඉලක්කම? (ලකුණු 2) *(ලකුණු 4)*

> (i) 1, 8, 64, 512 (ii) පාදය 8 → 7; පාදය 16 → F (15)

6. Web නිර්මාණකරුවෙකුට රතු = 200, කොළ = 30, නිල් = 255 වර්ණය අවශ්‍යයි. වර්ණ කේත #RRGGBB ලෙස hex වලින් ලියයි. (i) 200, 30, 255 hex වලට හරවන්න. (ලකුණු 3) (ii) වර්ණ කේතය ලියන්න. (ලකුණු 1) *(ලකුණු 4)*

> (i) 200 = C8, 30 = 1E, 255 = FF (ii) #C81EFF

සාරාංශය

- සංඛ්‍යා පද්ධති හතර: binary (2), octal (8), දශම (10), hexadecimal (16).
- පද්ධතියක විශාලතම ඉලක්කම = පාදය − 1.
- Hex හි 10–15 සඳහා A–F.
- ස්ථානීය අගයන් පාදයේ බල: 1, පාදය, පාදය^2, පාදය^3 ...
- Octal ස්ථානීය අගයන්: 1, 8, 64, 512. Hex: 1, 16, 256, 4096.
- දශම → ඕනෑම පාදයක්: පාදයෙන් නැවත නැවත බෙදා ශේෂ ඉහළට කියවන්න.
- ඉලක්කම් ස්ථානීය අගයෙන් ගුණ කර එකතු කර පරීක්ෂා කරන්න.
- Hex වර්ණ කේත, MAC සහ IPv6 ලිපින සඳහා භාවිතා වේ.

MCQ පිළිතුරු – 3.2

1. **B** – Hexa (6) + decimal (10) = 16.
2. **D** – Octal ඉලක්කම් 0–7.
3. **A** – A10 B11 C12 D13.

4. **B** – ශේෂ 4, 3, 2 ඉහළට → 234_8.
5. **A** – 250 = 15×16 + 10 → FA.
6. **C** – 32 + 8 + 4 + 2 + 1 = 47.
7. **A** – $8^0=1$, $8^1=8$, $8^2=64$.
8. **D** – $16^2 = 256$.
9. **B** – 255 = 15×16 + 15 → FF.
10. **A** – $64 = 1\times8^2 \rightarrow 100_8$.
11. **C** – 200 = 12×16 + 8 → C8.
12. **C** – B = 11.
13. **B** – Hex ඉලක්කම් 0–9 සහ A–F පමණි.
14. **D** – $16 = 1\times16 + 0 \rightarrow 10_{16}$.
15. **C** – වර්ණ කේත hex ඉලක්කම් 6 (RRGGBB).

© 2026 schoolict.net (Ruwan Mangala Suraweera). සියලු හිමිකම් ඇවිරිණි.

G.C.E. O/L ICT · 10 ශ්‍රේණිය · පාඩම 3

3.3 Binary ↔ දශම, Octal සහ Hexadecimal හැරවීම්

දැන් අපි අනෙක් පැත්තට යමු. ස්ථානීය අගය භාවිතයෙන් **binary → දශම** ද, **bit 3 සහ 4 කාණ්ඩ කිරීමෙන් binary ↔ octal** සහ **binary ↔ hexadecimal** ද ඉක්මනින් හරවමු. අවසානයේ binary හරහා **octal ↔ hexadecimal** හරවමු. ධන පූර්ණ සංඛ්‍යා පමණක් භාවිතා වේ.

1. Binary → දශම (ස්ථානීය අගය ක්‍රමය)

දකුණේ සිට සෑම bit එකකටම යටින් ස්ථානීය අගය (1, 2, 4, 8 ...) ලියන්න. Bit එක **1** වන ස්ථානවල අගයන් එකතු කරන්න.

උදාහරණය 1: 101101_2 → දශම

ඉලක්කම	1	0	1	1	0	1
ස්ථානීය අගය	$2^5 = 32$	$2^4 = 16$	$2^3 = 8$	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$
ඉලක්කම × ස්ථානීය අගය	1 × 32 = 32	0 × 16 = 0	1 × 8 = 8	1 × 4 = 4	0 × 2 = 0	1 × 1 = 1

💡 32 + 0 + 8 + 4 + 0 + 1 = **45_{10}**

Octal සහ hex → දශම සඳහාද එම ක්‍රමයමයි: සෑම ඉලක්කමක්ම ස්ථානීය අගයෙන් ගුණ කර එකතු කරන්න.

උදාහරණය 2: $3A7_{16}$ → දශම

ඉලක්කම	3	A	7
ස්ථානීය අගය	$16^2 = 256$	$16^1 = 16$	$16^0 = 1$
ඉලක්කම × ස්ථානීය අගය	3 × 256 = 768	10 × 16 = 160	7 × 1 = 7

💡 768 + 160 + 7 = **935_{10}**

2. Binary ↔ octal (bit 3 කාණ්ඩ)

$8 = 2^3$ නිසා **octal ඉලක්කම් 1 = bit 3 හරියටම**. Bits **දකුණේ සිට** තුන බැගින් කාණ්ඩ කරන්න; අවශ්‍ය නම් වමට 0 එකතු කරන්න. සෑම කාණ්ඩයක්ම එක් ඉලක්කමකට හරවන්න.

Octal	0	1	2	3	4	5	6	7
Binary	000	001	010	011	100	101	110	111

උදාහරණය 3: 1101011_2 → octal

කාණ්ඩය	001	101	011
අගය	1	5	3

💡 1101011_2 = **153_8** (001 සෑදීමට වමට 0 දෙකක් එකතු කළා).

උදාහරණය 4: 572_8 → binary

ඉලක්කම	5	7	2
බිටු	101	111	010

💡 572_8 = **101111010_2**

3. Binary ↔ hexadecimal (bit 4 කාණ්ඩ)

16 = 2^4 නිසා **hex ඉලක්කම් 1 = bit 4 හරියටම** (nibble). දකුණේ සිට හතර බැගින් කාණ්ඩ කරන්න.

Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Binary	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111

උදාහරණය 5: 1011111010_2 → hexadecimal

කාණ්ඩය	0010	1111	1010
අගය	2	F	A

💡 1011111010_2 = **$2FA_{16}$**

උදාහරණය 6: $9C_{16}$ → binary

ඉලක්කම	9	C
බිටු	1001	1100

💡 $9C_{16}$ = **10011100_2**

4. Octal ↔ hexadecimal (binary හරහා)

8 සහ 16 අතර සෘජු කාණ්ඩ කිරීමක් නැත. Binary හරහා යන්න: **octal → binary (බැගින් bit 3) → 4 බැගින් නැවත කාණ්ඩ → hex**; hex → octal සඳහා ප්‍රතිවිරුද්ධව.

උදාහරණය 7: 357_8 → hexadecimal

ඉලක්කම	3	5	7
බිටු	011	101	111

කාණ්ඩය	1110	1111
අගය	E	F

💡 357_8 = 11101111_2 = **EF_{16}** (පරීක්ෂාව: දෙකම 239_{10}).

උදාහරණය 8: $2B_{16}$ → octal

ඉලක්කම	2	B
බිටු	0010	1011

කාණ්ඩය	101	011
අගය	5	3

💡 $2B_{16} = 101011_2 = \mathbf{53_8}$

💡 **පොදු වැරදි:** දකුණ වෙනුවට වමේ සිට කාණ්ඩ කිරීම; ඉදිරි 0 අමතක කිරීම; hex හි A වෙනුවට 10 ලිවීම.

ප්‍රධාන පද

කාණ්ඩ කිරීම – Binary අංකයක් දකුණේ සිට bit 3 හෝ 4 කාණ්ඩවලට බෙදීම.

Nibble – bit 4 ක කාණ්ඩය = hex ඉලක්කම් එකක්.

Triad (bit 3 කාණ්ඩය) – bit 3 ක කාණ්ඩය = octal ඉලක්කම් එකක්.

ඉදිරි ශූන්‍යය – කාණ්ඩයක් සම්පූර්ණ කිරීමට වමට එක් කරන 0; අගය වෙනස් නොවේ.

ස්ථානීය අගය ක්‍රමය – දශම ලබාගැනීමට සෑම ඉලක්කමක්ම ස්ථානීය අගයෙන් ගුණ කර එකතු කිරීම.

Binary සිට දශම – Bit 1 වන ස්ථානවල අගයන් එකතු කිරීම.

Binary සිට octal – Bits දකුණේ සිට 3 බැගින් කාණ්ඩ කර හැරවීම.

Binary සිට hexadecimal – Bits දකුණේ සිට 4 බැගින් කාණ්ඩ කර හැරවීම.

Octal සිට binary – සෑම octal ඉලක්කමක්ම bit 3 ලෙස ලිවීම.

Hexadecimal සිට binary – සෑම hex ඉලක්කමක්ම bit 4 ලෙස ලිවීම.

Octal සිට hexadecimal – Octal → binary, පසුව 4 බැගින් නැවත කාණ්ඩ කිරීම.

ධන නිඛිලය – බිංදුවට වැඩි පූර්ණ සංඛ්‍යාවක් (භාග නැත, ඍණ නැත).

ආදර්ශ ප්‍රශ්න සහ පිළිතුරු

1. 11010_2 දශම වලට හරවන්න.

පිළිතුර: $16 + 8 + 2 = 26_{10}$.

2. Octal සඳහා bits 3 බැගින් කාණ්ඩ කරන්නේ ඇයි?

පිළිතුර: $8 = 2^3$ නිසා සෑම octal ඉලක්කමක්ම හරියටම bit 3 කි.

3. 11110110_2 hexadecimal වලට හරවන්න.

පිළිතුර: 1111 | 0110 → F | 6 → $F6_{16}$.

4. 47_8 binary වලට හරවන්න.

පිළිතුර: 4 → 100, 7 → 111 → 100111_2.

5. Bits කාණ්ඩ කිරීම ආරම්භ කරන්නේ කුමන පැත්තෙන්ද?

පිළිතුර: දකුණෙන් (LSB). අවසාන කාණ්ඩය අඩු නම් වමට 0 එකතු කරන්න.

6. Octal → hexadecimal හරවන්නේ කෙසේද?

> **පිළිතුර:** සෑම octal ඉලක්කමක්ම bit 3 කර, දකුණේ සිට 4 බැගින් නැවත කාණ්ඩ කර hex ඉලක්කම් කරන්න.

බහුවරණ ප්‍රශ්න

1. 101101_2 දශම වලින්?

A. 45
B. 90
C. 54
D. 43

2. 1111_2 දශම වලින්?

A. 16
B. 1111
C. 15
D. 4

3. 1100100_2 දශම වලින්?

A. 100
B. 200
C. 98
D. 50

4. 1101011_2 octal වලින්?

A. 513
B. 153
C. 651
D. 326

5. 1011111010_2 hexadecimal වලින්?

A. 2EA
B. 2FA0
C. 2FA
D. BE2

6. 572_8 binary වලින්?

A. 101011010
B. 111101010
C. 101111001
D. 101111010

7. $9C_{16}$ binary වලින්?

A. 10010011
B. 11001001
C. 10011100
D. 10011010

8. Binary → hexadecimal හැරවීමේදී bits කාණ්ඩ කරන්නේ:

A. වමේ සිට අට බැගින්
B. වමේ සිට හතර බැගින්
C. දකුණේ සිට තුන බැගින්
D. දකුණේ සිට හතර බැගින්

9. 357_8 hexadecimal වලින්?

A. 357
B. EF
C. FE
D. DF

10. $2B_{16}$ octal වලින්?

A. 53
B. 52
C. 35
D. 43

11. $3A7_{16}$ දශම වලින්?

A. 937
B. 935
C. 3107
D. 967

12. 11110110_2 hexadecimal වලින්?

A. 6F
B. F5
C. F6
D. E6

13. Hexadecimal ඉලක්කම් එකක් bits කීයකට සමානද?

A. 4
B. 8
C. 16
D. 3

14. 75_8 දශම වලින්?

A. 117
B. 57
C. 75
D. 61

15. $7F_{16}$ ට සමාන binary අංකය?

A. 11110111
B. 1111111
C. 1111111111
D. 01110000

ව්‍යුහගත ප්‍රශ්න

1. 10110011_2 (i) දශම (ii) octal (iii) hexadecimal වලට හරවන්න. ක්‍රියාවලිය පෙන්වන්න. (ලකුණු 6) *(ලකුණු 6)*

(i) 128+32+16+2+1 = 179_{10}
(ii) 010 110 011 → 263_8
(iii) 1011 0011 → $B3_{16}$

2. 645_8 binary වලට, පසුව hexadecimal වලට හරවන්න. (ලකුණු 4) *(ලකුණු 4)*

6→110, 4→100, 5→101 → 110100101_2. නැවත කාණ්ඩ: 0001 1010 0101 → $1A5_{16}$.

3. $A5_{16}$ (i) binary (ii) octal (iii) දශම වලට හරවන්න. (ලකුණු 3) *(ලකුණු 3)*

(i) 1010 0101 → 10100101_2 (ii) 010 100 101 → 245_8 (iii) 10×16+5 = 165_{10}

4. සිසුවෙක් 1101101_2 hex වලට 1101 101 → D5 ලෙස හැරෙව්වා. (i) කළ වරද කුමක්ද? (ලකුණු 1) (ii) නිවැරදි පිළිතුර දෙන්න. (ලකුණු 2) *(ලකුණු 3)*

(i) දකුණ වෙනුවට වමේ සිට කාණ්ඩ කළා. (ii) 0110 1101 → $6D_{16}$.

5. දශම වලට හරවා ආරෝහණ පිළිවෙළට සකසන්න: 1110_2, 17_8, 10_{16}. (ලකුණු 3) *(ලකුණු 3)*

1110_2 = 14, 17_8 = 15, 10_{16} = 16 → 1110_2, 17_8, 10_{16}.

6. Binary හරහා $1F_{16}$ octal වලට හරවන්න. පියවර දෙකම පෙන්වන්න. (ලකුණු 3) *(ලකුණු 3)*

1→0001, F→1111 → 11111_2. 3 බැගින්: 011 111 → 37_8.

සාරාංශය

- Binary → දශම: bit 1 වන ස්ථානවල අගයන් එකතු කරන්න.
- Octal/hex → දශම: ඉලක්කම × ස්ථානීය අගය, පසුව එකතු කරන්න.
- Octal ඉලක්කම් 1 = bit 3 ($8 = 2^3$).
- Hex ඉලක්කම් 1 = bit 4 ($16 = 2^4$), nibble.
- හැම විටම දකුණේ සිට කාණ්ඩ කරන්න; වමට 0 එකතු කරන්න.
- Octal/hex → binary: සෑම ඉලක්කමක්ම bit 3 හෝ 4 ලෙස.
- Octal ↔ hex: binary හරහා ගොස් නැවත කාණ්ඩ කරන්න.
- දෙකම දශම වලට හරවා පිළිතුරු පරීක්ෂා කරන්න.

MCQ පිළිතුරු – 3.3

1. **A** – 32+8+4+1 = 45.
2. **C** – 8+4+2+1 = 15.
3. **A** – 64+32+4 = 100.
4. **B** – 001 101 011 → 1 5 3.
5. **C** – 0010 1111 1010 → 2 F A.
6. **D** – 5→101, 7→111, 2→010.
7. **C** – 9→1001, C→1100.
8. **D** – $16 = 2^4$; LSB වලින් පටන් ගන්න.
9. **B** – 357_8 = 11101111_2 = EF_{16}.

10. **A** – 2B = 101011_2 = 101 011 = 53_8.
11. **B** – 3×256 + 10×16 + 7 = 935.
12. **C** – 1111 0110 → F6.
13. **A** – 16 = 2^4 → bit 4.
14. **D** – 7×8 + 5 = 61.
15. **B** – 7→0111, F→1111 → 1111111.

© 2026 schoolict.net (Ruwan Mangala Suraweera). සියලු හිමිකම් ඇවිරිණි.

G.C.E. O/L ICT · 10 ශ්‍රේණිය · පාඩම 3

3.4 දත්ත ගබඩා ඒකක – Bit, Byte, KB, MB, GB, TB

ඡායාරූපයක්, සින්දුවක් හෝ ඔබේ phone එකේ ගබඩාව කොච්චර විශාලද? **ගබඩා ඒකක** – bit, byte, kilobyte, megabyte, gigabyte, terabyte – **1 KB = bytes 1024** භාවිතයෙන් **ඒවා අතර හැරවීම**, සහ cache, RAM, ROM, SSD/hard disk, CD/DVD, USB drives, memory cards සහ cloud ගබඩාවල **ධාරිතාව සැසඳීම** මේ සටහනෙන් ඉගෙන ගමු.

1. Bit සහ byte

Bit (0 හෝ 1) කුඩාම ඒකකයයි. **bit 8 = byte 1**. Byte එකකට 'A' වැනි එක් අක්ෂරයක් (ASCII) ගබඩා කළ හැක. bit 4 = **nibble** 1.

2. ගබඩා ඒකක පඩිපෙළ

ඒකකය	සමාන	Bytes වලින්	ආසන්නව දරන්නේ
Byte 1 (B)	bit 8	1	එක් අකුරක්
Kilobyte 1 (KB)	bytes 1024	2^{10} = 1024	කෙටි පණිවිඩයක් / පිටුවෙන් භාගයක්
Megabyte 1 (MB)	KB 1024	2^{20} = 1,048,576	විනාඩි 1 MP3 සින්දුවක්
Gigabyte 1 (GB)	MB 1024	2^{30} = 1,073,741,824	HD වීඩියෝ පැය 1ක් පමණ
Terabyte 1 (TB)	GB 1024	2^{40} = 1,099,511,627,776	ඡායාරූප 250,000ක් පමණ

💡 **නීතිය: විශාල** ඒකකයකට → **1024 න් බෙදන්න**. **කුඩා** ඒකකයකට → **1024 න් ගුණ කරන්න**. Bits ↔ bytes → **8** න් ගුණ/බෙදීම.

3. ගණනය කිරීම් උදාහරණ

ප්‍රශ්නය	ක්‍රියාවලිය	පිළිතුර
KB 2 bytes වලින්	2 × 1024	2048 bytes
MB 3 KB වලින්	3 × 1024	3072 KB
GB 16 MB වලින්	16 × 1024	16,384 MB
GB 2048 TB වලින්	2048 ÷ 1024	2 TB
Bytes 5 bits වලින්	5 × 8	40 bits
KB 4 bits වලින්	4 × 1024 × 8	32,768 bits
MB 512 bytes වලින්	512 × 1024 × 1024	536,870,912 bytes

උදාහරණය: ඡායාරූප කීයක් ගැළපේද?

GB 64 memory card එකක MB 4 බැගින් ඡායාරූප ගබඩා වේ. GB 64 = 64 × 1024 MB = MB 65,536. ඡායාරූප = 65,536 ÷ 4 = **ඡායාරූප 16,384**.

💡 **අමතර සටහන – KB සහ KiB:** O/L විභාගයේ KB 1 = bytes 1024. ජාත්‍යන්තර ප්‍රමිතියේ bytes 1024 = **kibibyte (KiB)** වන අතර **kB = bytes 1000**. ඒ නිසා '1 TB' disk එකක් Windows හි GB 931ක් පමණ පෙන්වයි.

4. ගබඩා උපාංග සැසඳීම

උපාංගය	සාමාන්‍ය ධාරිතාව (2026)	සටහන්
Cache මතකය	a few MB (e.g. 8–64 MB)	CPU එක තුළ; වේගවත්ම, කුඩාම
ROM (firmware)	a few MB (e.g. 16–32 MB)	BIOS/UEFI ආරම්භක වැඩසටහන
RAM	8 – 16 GB (PC), 4 – 12 GB (phone)	ප්‍රධාන මතකය, නශ්‍ය
CD / DVD	700 MB / 4.7 GB	Optical, දැන් කලාතුරකින්
USB flash drive / memory card	32 GB – 1 TB	රැගෙන යා හැකි flash ගබඩාව
SSD / NVMe	256 GB – 2 TB	වේගවත් flash ද්විතීයික ගබඩාව
Hard disk (HDD)	1 – 20 TB	චුම්බක, GB එකකට ලාභයි
Cloud ගබඩාව	15 GB free – many TB	Online, internet අවශ්‍යයි

ධාරිතාවේ පිළිවෙළ (කුඩාම → විශාලතම, සාමාන්‍ය): Cache < ROM < CD < DVD < RAM < USB flash drive < SSD < hard disk. වේගය බොහෝ දුරට ප්‍රතිවිරුද්ධයි: cache වේගවත්ම.

2026 නව **2026 යාවත්කාලීනය:** floppy disks ඉතිහාසයට එකතුයි. නව laptops බොහොමයක් **NVMe SSD** භාවිතා කරයි, phones තුළ flash (GB 128 – TB 1), සිසුන් වැඩ **cloud ගබඩාවේ** (Google Drive, OneDrive) සුරකියි.

ප්‍රධාන පද

Byte – bit 8 ක කාණ්ඩය; එක් අක්ෂරයක් ගබඩා කරයි.

Nibble – Byte භාගයක් = bit 4.

Kilobyte (KB) – Bytes 1024 (2^{10}).

Megabyte (MB) – KB 1024 = bytes 1,048,576.

Gigabyte (GB) – MB 1024; RAM සහ phone ගබඩාව සඳහා.

Terabyte (TB) – GB 1024; hard disks සහ විශාල SSD සඳහා.

ගබඩා ධාරිතාව – උපාංගයකට දැරිය හැකි දත්ත ප්‍රමාණය.

Cache මතකය – CPU තුළ ඉතා වේගවත්, කුඩා මතකය (MB කිහිපයක්).

SSD – Solid State Drive – චලන කොටස් නැති වේගවත් flash ගබඩාව.

Memory card – Phones සහ cameras වල භාවිත කුඩා flash card (SD/microSD).

Cloud ගබඩාව – Internet servers මත ගොනු සුරැකීම, උදා: Google Drive.

Kibibyte (KiB) – හරියටම bytes 1024 සඳහා ප්‍රමිත නම; kB = bytes 1000.

ආදර්ශ ප්‍රශ්න සහ පිළිතුරු

1. Byte එකක bits කීයක්?

> **පිළිතුර:** bit 8.

2. MB 3 KB වලට හරවන්න.

> **පිළිතුර:** 3 × 1024 = KB 3072.

3. MB 4096 GB වලට හරවන්න.

> **පිළිතුර:** 4096 ÷ 1024 = GB 4.

4. Cache, hard disk, RAM සහ DVD කුඩාම සිට විශාලතම ධාරිතාවට සකසන්න.

> **පිළිතුර:** Cache < DVD (4.7 GB) < RAM (8–16 GB) < hard disk.

5. TB 1 disk එකක් PC එකේ GB 1024 ට අඩුවෙන් පෙන්වන්නේ ඇයි?

> **පිළිතුර:** නිෂ්පාදකයින් TB 1 = bytes 1,000,000,000,000 ලෙස ගණනය කරයි; PC එක 1024 න් බෙදන නිසා GB 931ක් පමණ පෙන්වයි.

6. Cloud ගබඩාවේ වාසි දෙකක් දෙන්න.

> **පිළිතුර:** ඕනෑම තැනක ඕනෑම උපාංගයකින් ගොනු ලබාගත හැක; ස්වයංක්‍රීය backup; පහසු බෙදාගැනීම.

බහුවරණ ප්‍රශ්න

1. Byte එකක් සෑදෙන bits ගණන?

A. 8
B. 1024
C. 4
D. 16

2. KB 1 සමාන වන්නේ:

A. Bits 1024
B. Bytes 1024
C. Bytes 8
D. Bits 1000

3. MB 2 ක KB කීයක්?

A. 512
B. 2000
C. 2048
D. 1024

4. GB 16 ක MB කීයක්?

A. 16000
B. 8192
C. 16384
D. 1600

5. GB 2048 සමාන වන්නේ:

A. TB 0.5
B. TB 20
C. MB 2
D. TB 2

6. Bytes 4 ක bits කීයක්?

A. 4096
B. 12
C. 32
D. 4

7. කුඩාම සිට විශාලතම දක්වා නිවැරදි පිළිවෙළ?

A. KB, bit, byte, GB, MB, TB
B. Byte, bit, KB, MB, TB, GB
C. Bit, KB, byte, MB, GB, TB
D. Bit, byte, KB, MB, GB, TB

8. සාමාන්‍යයෙන් කුඩාම ධාරිතාව ඇති මතකය?

A. Cache
B. RAM
C. SSD
D. Hard disk

9. 2026 සාමාන්‍ය සිසු laptop එකක RAM ප්‍රමාණය:

A. MB 8 – 16
B. GB 8 – 16
C. KB 640
D. TB 8 – 16

10. GB 64 memory card එකක MB 4 ඡායාරූප කීයක් ගැළපේද?

A. 16,384
B. 256
C. 1,600
D. 16

11. මේවායින් විශාලතම ධාරිතාව ඇත්තේ?

A. DVD
B. CD
C. RAM
D. Hard disk (TB 4)

12. CD එකක ගබඩා කළ හැක්කේ:

A. TB 4.7
B. KB 7
C. MB 700
D. GB 700

13. භාවිතයට internet අවශ්‍ය ගබඩාව?

A. Cloud ගබඩාව
B. Cache
C. SSD
D. USB flash drive

14. ජාත්‍යන්තර ප්‍රමිතියේ kB 1 (කුඩා k) යනු:

A. Bytes 1024
B. Bytes 1000
C. Bytes 8
D. Bits 1000

15. MB 1 ක bytes කීයක්?

A. 1,024
B. 1,048,576
C. 1,000
D. 1,073,741,824

ව්‍යුහගත ප්‍රශ්න

1. හිස්තැන් පුරවන්න: (i) byte 1 = bits __ (ii) KB 1 = bytes __ (iii) GB 1 = MB __ (iv) TB 1 = GB __ (ලකුණු 4) *(ලකුණු 4)*

> (i) 8 (ii) 1024 (iii) 1024 (iv) 1024

2. හරවන්න: (i) KB 5 bytes වලට (ii) MB 8192 GB වලට (iii) bytes 3 bits වලට. ක්‍රියාවලිය පෙන්වන්න. (ලකුණු 3) *(ලකුණු 3)*

> (i) 5 × 1024 = bytes 5120 (ii) 8192 ÷ 1024 = GB 8 (iii) 3 × 8 = bits 24

3. සිසුවෙකුට GB 32 USB drive එකක් ඇත. සෑම වීඩියෝවක්ම MB 512. (i) GB 32 MB වලට හරවන්න. (ලකුණු 1) (ii) වීඩියෝ කීයක් ගබඩා කළ හැකිද? (ලකුණු 2) *(ලකුණු 3)*

> (i) 32 × 1024 = MB 32,768 (ii) 32,768 ÷ 512 = වීඩියෝ 64

4. සාමාන්‍ය ධාරිතාව අනුව ආරෝහණ පිළිවෙළට සකසන්න: RAM, cache, SSD, DVD, ROM. (ලකුණු 3) *(ලකුණු 3)*

> Cache (MB) < ROM (MB) < DVD (GB 4.7) < RAM (GB 8–16) < SSD (GB 256–TB 2).

5. පෙළ ගොනුවක අක්ෂර 2000ක් ඇත. සෑම අක්ෂරයක්ම byte 1. (i) ප්‍රමාණය bytes වලින්? (ලකුණු 1) (ii) එය KB 2 ට වඩා වැඩිද අඩුද? පැහැදිලි කරන්න. (ලකුණු 2) *(ලකුණු 3)*

> (i) bytes 2000 (ii) අඩුයි – KB 2 = 2 × 1024 = bytes 2048; 2000 < 2048.

6. (i) Cloud ගබඩාවේ එක් වාසියක් සහ එක් අවාසියක් දෙන්න. (ලකුණු 2) (ii) නව laptops වල hard disk වෙනුවට SSD වඩා කැමති ඇයි? (ලකුණු 2) *(ලකුණු 4)*

(i) වාසිය: ඕනෑම තැනක සිට / backup / බෙදාගැනීම. අවාසිය: internet අවශ්‍යයි / පෞද්ගලිකත්ව අවදානම / විශාල ඉඩට මාසික ගාස්තු. (ii) වේගවත් ආරම්භය සහ loading, චලන කොටස් නැති නිසා ශක්තිමත්, සැහැල්ලු, අඩු බලය.

සාරාංශය

- Byte 1 = bit 8; nibble 1 = bit 4.
- KB 1 = bytes 1024; MB 1 = KB 1024; GB 1 = MB 1024; TB 1 = GB 1024.
- විශාල ඒකකයට → 1024 න් බෙදන්න; කුඩා ඒකකයට → 1024 න් ගුණ කරන්න.
- Bytes ↔ bits: 8 න් ගුණ/බෙදීම.
- ධාරිතාව: cache < ROM < CD < DVD < RAM < USB < SSD < HDD (සාමාන්‍ය).
- 2026: RAM GB 8–16, SSD GB 256–TB 2, cache MB කිහිපයක්.
- Cloud ගබඩාව ගොනු online තබයි; internet අවශ්‍යයි.
- kB = bytes 1000, KiB = bytes 1024 (අමතර ප්‍රමිතිය).

MCQ පිළිතුරු – 3.4

1. **A** – Byte 1 = bit 8.
2. **B** – O/L: KB 1 = bytes 1024.
3. **C** – 2 × 1024 = 2048.
4. **C** – 16 × 1024 = 16384.
5. **D** – 2048 ÷ 1024 = 2.
6. **C** – 4 × 8 = 32.
7. **D** – සෑම පියවරක්ම × 1024 (byte = bit 8).
8. **A** – Cache MB කිහිපයක් පමණි.
9. **B** – නූතන RAM GB වලින් මනිනවා.
10. **A** – 64 × 1024 ÷ 4 = 16,384.
11. **D** – HDD TB ගණනක් දක්වා.
12. **C** – CD ≈ MB 700; DVD ≈ GB 4.7.
13. **A** – Cloud ගොනු internet servers මත.
14. **B** – kB = bytes 1000; KiB = bytes 1024.
15. **B** – 1024 × 1024.

© 2026 schoolict.net (Ruwan Mangala Suraweera). සියලු හිමිකම් ඇවිරිණි.

G.C.E. O/L ICT · 10 ශ්‍රේණිය · පාඩම 3

3.5 කේතන පද්ධති – BCD, EBCDIC, ASCII සහ Unicode

ඔබ **A** අකුර type කළ විට පරිගණකය 'A' රූපයක් ගබඩා නොකරයි – binary **කේත අංකයක්** ගබඩා කරයි. **කේතන පද්ධතියක්** යනු අක්ෂර binary කේතවලට සම්බන්ධ කරන එකඟ වූ වගුවකි. **BCD, EBCDIC, ASCII සහ Unicode**, ඒවා භාවිත වන තැන් සහ ඒවායේ සීමා මේ සටහනෙන් ඉගෙන ගමු.

1. කේතන පද්ධති අවශ්‍ය ඇයි?

පරිගණක තේරුම් ගන්නේ 0 සහ 1 පමණයි. අකුරු, ඉලක්කම්, සංකේත ගබඩා කිරීමට සෑම අක්ෂරයකටම අනන්‍ය අංකයක් දෙනවා. සියලු පරිගණක **එකම කේතය** භාවිතා කළොත් එක් උපාංගයක type කළ පණිවිඩයක් තවත් උපාංගයක නිවැරදිව කියවිය හැක.

2. BCD – Binary Coded Decimal

BCD **සෑම දශම ඉලක්කමක්ම වෙන වෙනම bit 4 කින්** කේත කරයි. ඉලක්කම් එකින් එක පෙන්වන digital ඔරලෝසු, calculators සහ ඉලෙක්ට්‍රොනික තිරවල භාවිතා වේ.

දශම	BCD	සාමාන්‍ය binary
7	0111	111
59	0101 1001	111011
2026	0010 0000 0010 0110	11111101010

💡 **සටහන:** 59 හි BCD (0101 1001) 59 හි binary (111011) නොවේ. BCD 0000–1001 පමණක් භාවිතා කරයි; 1010–1111 භාවිත නොවන නිසා ඉඩ නාස්තියි, **අකුරු කේත කළ නොහැක**.

3. EBCDIC

EBCDIC (Extended Binary Coded Decimal Interchange Code) **IBM** සමාගම mainframe පරිගණක සඳහා සෑදූ **bit 8** කේතයකි. අක්ෂර 256ක් කේත කළ හැක. උදා: 'A' = $C1_{16}$ = 11000001_2.

- සීමාව: ප්‍රධාන වශයෙන් IBM mainframes වල පමණි – ASCII සමඟ නොගැළපේ.
- අකුරු එක අඛණ්ඩ පෙළක නොමැති නිසා පිළිවෙළ කිරීම අපහසුයි.

4. ASCII

ASCII (American Standard Code for Information Interchange) ඉංග්‍රීසි පෙළ සඳහා බහුලවම භාවිත කේතයයි. **සම්මත ASCII bit 7 → 2^7 = අක්ෂර 128** (කේත 0–127). **Extended ASCII bit 8 → අක්ෂර 256**; é, £ වැනි සංකේත එකතු කරයි.

අක්ෂරය	දශම	Binary (bit 7)	Hex
A	65	1000001	41
B	66	1000010	42
Z	90	1011010	5A
a	97	1100001	61

0	48	0110000	30
9	57	0111001	39
space	32	0100000	20

💡 **විභාග කෙටි මං:** 'A' = 65, 'a' = 97 (කුඩා අකුර = ලොකු අකුර + 32), '0' = 48. ඒ නිසා 'C' = 67, 'c' = 99, '5' = 53.

උදාහරණය: CAT වචනය ASCII වලින් = 67 65 84 = 1000011 1000001 1010100.

- සීමාව: අක්ෂර 128 (හෝ 256) පමණි – සිංහල, දෙමළ, චීන හෝ emoji සඳහා ඉඩ නැත.

5. Unicode

Unicode සෑම භාෂාවකම සෑම අක්ෂරයකටම සහ සංකේත, emoji වලට අනන්‍ය කේතයක් (code point) දෙයි – අක්ෂර 150,000කට වැඩියි. U+ සහ hex ලෙස ලියයි. පළමු Unicode කේත 128 ASCII ට සමානයි. **UTF-8** (අක්ෂරයකට bytes 1–4) හෝ UTF-16 භාවිතයෙන් ගබඩා කරයි.

අක්ෂරය	Code point	UTF-8 bytes
A	U+0041	1
අ	U+0D85	3
ක	U+0D9A	3
😀	U+1F600	4

2026 නව **Unicode හි සිංහල:** සිංහල අකුරු **U+0D80 – U+0DFF** කොටසේ ඇත (උදා: අ = U+0D85). Unicode නිසා Facebook, WhatsApp සහ මේ වෙබ් අඩවියේ සිංහල type කළ හැකි අතර 😀 (U+1F600) වැනි emoji හැම phone එකකම ක්‍රියා කරයි.

- සීමාව: ASCII ට වඩා ගබඩා ඉඩ වැඩියි (බොහෝ අක්ෂරවලට bytes 2–4).
- පැරණි මෘදුකාංග/උපාංග සියලු අක්ෂර නොපෙන්විය හැක (□ කොටු පෙනේ).

6. සැසඳීම

කේතය	Bits	අක්ෂර	ප්‍රධාන භාවිතය	සීමාව
BCD	4 per digit	10 digits	ඔරලෝසු, calculators	ඉලක්කම් පමණි
EBCDIC	8	256	IBM mainframes	ASCII සමඟ නොගැළපේ
ASCII	7 (8 extended)	128 (256)	ඉංග්‍රීසි පෙළ, keyboards	අනෙක් අක්ෂර මාලා නැත
Unicode	8–32 (UTF-8: 1–4 bytes)	150,000+	සියලු භාෂා, emoji, web	ඉඩ වැඩියෙන්

ප්‍රධාන පද

කේතන පද්ධතිය – සෑම අක්ෂරයකටම binary කේතයක් දෙන එකඟ වූ වගුව.

අක්ෂරය – අකුරක්, ඉලක්කමක්, සංකේතයක් හෝ හිස්තැනක්.

BCD – Binary Coded Decimal – සෑම දශම ඉලක්කමක්ම bit 4 කින්.

EBCDIC – Mainframes සඳහා IBM හි bit 8 අක්ෂර කේතය.

ASCII – අක්ෂර 128 සඳහා bit 7 කේතය; 'A' = 65.	**Extended ASCII** – අක්ෂර 256 ඇති ASCII හි bit 8 අනුවාදය.
Unicode – සියලු භාෂා සහ emoji ආවරණය කරන විශ්ව කේතය.	**Code point** – Unicode අක්ෂරයකට දෙන අංකය, උදා: U+0D85.
UTF-8 – අක්ෂරයකට bytes 1–4 භාවිතයෙන් Unicode ගබඩා කරන ක්‍රමය.	**සිංහල Unicode කොටස** – සිංහල අකුරු සඳහා U+0D80–U+0DFF පරාසය.
Emoji – Unicode හි කේත කළ 😀 වැනි රූප අක්ෂර.	**ගැළපුම** – විවිධ පද්ධතිවලට එකම දත්ත නිවැරදිව කියවීමේ හැකියාව.

ආදර්ශ ප්‍රශ්න සහ පිළිතුරු

1. කේතන පද්ධතියක් යනු කුමක්ද?

> **පිළිතුර:** අක්ෂර සඳහා එකඟ වූ binary කේත කට්ටලයක්; එවිට සියලු පරිගණක පෙළ එකම ලෙස ගබඩා කර කියවයි.

2. 59 BCD වලින් ලියන්න.

> **පිළිතුර:** 5 → 0101, 9 → 1001 → 0101 1001.

3. bit 7 ASCII අක්ෂර කීයක් නිරූපණය කරයිද? ඇයි?

> **පිළිතුර:** 128, මන්ද $2^7 = 128$.

4. 'A' හි ASCII කේතය 65. 'E' සහ 'e' හි කේත?

> **පිළිතුර:** 'E' = 69; 'e' = 101 (32 එකතු කරන්න).

5. Unicode හඳුන්වා දුන්නේ ඇයි?

> **පිළිතුර:** ASCII ට සිංහල, දෙමළ, චීන වැනි භාෂා නිරූපණය කළ නොහැකි විය. Unicode සෑම භාෂාවකම සෑම අක්ෂරයකටම කේතයක් දෙයි.

6. සිංහල සඳහා භාවිත Unicode පරාසය?

> **පිළිතුර:** U+0D80 සිට U+0DFF.

බහුවරණ ප්‍රශ්න

1. සම්මත bit 7 ASCII අක්ෂර කීයක් නිරූපණය කරයිද?

A. 65,536
B. 256
C. 7
D. 128

2. 'A' හි ASCII කේතය 65. 'D' හි කේතය?

A. 100
B. 67
C. 68
D. 69

3. කුඩා 'a' හි ASCII කේතය:

A. 97
B. 1
C. 32
D. 65

4. IBM mainframes සඳහා සැකසූ කේතන පද්ධතිය?

A. Unicode
B. EBCDIC
C. ASCII
D. BCD

5. 47 BCD වලින්?

A. 0100 0111
B. 0100 0110
C. 0111 0100
D. 101111

6. සිංහල අකුරු නිරූපණය කළ හැකි කේතන පද්ධතිය?

A. BCD
B. EBCDIC
C. Unicode
D. bit 7 ASCII

7. Extended ASCII අක්ෂර කීයක් නිරූපණය කරයිද?

A. 512
B. 128
C. 256
D. 1024

8. BCD ප්‍රධාන වශයෙන් භාවිතා වන්නේ:

A. වෙබ් පිටු සිංහල පෙළ
B. Phones වල emoji
C. රූප සම්පීඩනය
D. Digital ඔරලෝසු සහ calculators

9. ASCII හි ප්‍රධාන සීමාවක්:

A. අක්ෂරයකට bit 32
B. A අකුර නිරූපණය කළ නොහැක
C. බොහෝ ඉංග්‍රීසි නොවන අක්ෂර මාලා නිරූපණය කළ නොහැක
D. IBM mainframes වල පමණි

10. සිංහල අ අකුරේ Unicode code point:

A. U+0D85
B. U+1F600
C. U+0D00
D. U+0041

11. '0' ඉලක්කමේ ASCII කේතය:

A. 65
B. 48
C. 0
D. 30

12. 'HI' වචනය ASCII වලින් ගබඩා වන්නේ:

A. 71 72
B. 72 73
C. 104 105
D. 8 9

13. 😀 වැනි emoji ගබඩා කළ හැක්කේ:

A. Unicode
B. EBCDIC
C. BCD
D. bit 7 ASCII

14. UTF-8 හි අ අකුර bytes කීයක් ගනීද?

A. 1
B. 4
C. 2
D. 3

15. bit 7 ASCII කේතය 1000010 නිරූපණය කරන්නේ:

A. 2
B. B
C. b
D. A

ව්‍යුහගත ප්‍රශ්න

1. (i) ASCII හි සම්පූර්ණ නම? (ලකුණු 1)
(ii) සම්මත ASCII bits කීයක් භාවිතා කරයිද, අක්ෂර කීයක් කේත කළ හැකිද? (ලකුණු 2)
(iii) ASCII හි එක් සීමාවක් සඳහන් කරන්න. (ලකුණු 1) *(ලකුණු 4)*

> (i) American Standard Code for Information Interchange (ii) bit 7; 2^7 = අක්ෂර 128 (iii) සිංහල/දෙමළ/චීන හෝ emoji කේත කළ නොහැක.

2. 'A' හි ASCII කේතය 65. (i) 'G' හි ASCII කේතය දශම වලින් ලියන්න. (ලකුණු 1) (ii) එය bit 7 binary වලට හරවන්න. (ලකුණු 2) (iii) 'g' හි කේතය? (ලකුණු 1) *(ලකුණු 4)*

> (i) 71 (ii) 1000111_2 (iii) 103

3. (i) 2026 BCD වලින් ලියන්න. (ලකුණු 2) (ii) 2026 සාමාන්‍ය binary වලින් ලියන්න. (ලකුණු 2) (iii) bits වැඩියෙන් ගන්නේ කුමක්ද? (ලකුණු 1) *(ලකුණු 5)*

> (i) 0010 0000 0010 0110 (ii) 11111101010_2 (iii) BCD bit 16; binary bit 11; ඒ නිසා BCD වැඩියි.

4. ASCII සහ Unicode සසඳන්න: (i) අක්ෂර ගණන (ii) සහාය දක්වන භාෂා (iii) අවශ්‍ය ගබඩා ඉඩ. (ලකුණු 3) *(ලකුණු 3)*

> (i) ASCII 128/256; Unicode 150,000+. (ii) ASCII – ඉංග්‍රීසි පමණි; Unicode – සිංහල, දෙමළ ඇතුළු සියලු භාෂා සහ emoji. (iii) ASCII byte 1; Unicode (UTF-8) bytes 1–4.

5. පණිවිඩයක් bit 7 ASCII වලින් ගබඩා කර ඇත: 1000011 1000001 1010100. (i) සෑම කේතයක්ම දශම වලට හරවන්න. (ලකුණු 2) (ii) වචනය විකේත කරන්න. (ලකුණු 1) *(ලකුණු 3)*

> (i) 67, 65, 84 (ii) CAT

6. සිසුවෙක් WhatsApp හි 😀 emoji සහිත සිංහල පණිවිඩයක් type කරයි. (i) මෙය හැකි කරන කේතන පද්ධතිය? (ලකුණු 1) (ii) සිංහල Unicode පරාසය? (ලකුණු 1) (iii) මේ සඳහා EBCDIC භාවිත නොකරන්නේ ඇයි? (ලකුණු 1) *(ලකුණු 3)*

> (i) Unicode (UTF-8) (ii) U+0D80 – U+0DFF (iii) EBCDIC හි කේත 256 පමණි, IBM mainframes වල පමණි – සිංහල හෝ emoji නැත.

සාරාංශය

- කේතන පද්ධතියක් සෑම අක්ෂරයකටම binary කේතයක් දෙයි.
- BCD සෑම දශම ඉලක්කමක්ම bit 4 කින් (ඔරලෝසු, calculators).
- EBCDIC: bit 8 IBM mainframe කේතය, අක්ෂර 256.
- ASCII: bit 7 = අක්ෂර 128; extended ASCII bit 8 = 256.
- ASCII: 'A' = 65, 'a' = 97, '0' = 48.
- Unicode සියලු භාෂා සහ emoji ආවරණය කරයි; U+hex ලෙස ලියයි.
- සිංහල Unicode කොටස: U+0D80 – U+0DFF; UTF-8 මගින් ගබඩා.
- සීමා: BCD ඉලක්කම් පමණි, EBCDIC IBM පමණි, ASCII ඉංග්‍රීසි පමණි, Unicode ඉඩ වැඩියි.

MCQ පිළිතුරු – 3.5

1. **D** – $2^7 = 128$.
2. **C** – A65 B66 C67 D68.
3. **A** – 'a' = 'A' + 32 = 97.
4. **B** – EBCDIC = IBM, bit 8.
5. **A** – 4 → 0100, 7 → 0111.
6. **C** – සිංහල කොටස U+0D80–U+0DFF.
7. **C** – bit 8 → $2^8 = 256$.
8. **D** – BCD දශම ඉලක්කම් එකින් එක පෙන්වයි.
9. **C** – අක්ෂර 128/256 පමණි.
10. **A** – සිංහල කොටස U+0D80 න් ඇරඹේ; අ = U+0D85.
11. **B** – '0' = 48 (hex හි 30).
12. **B** – H = 72, I = 73.
13. **A** – 😀 = U+1F600.
14. **D** – U+0800 සිට U+FFFF දක්වා bytes 3.

15. **B** – 1000010_2 = 66 = 'B'.

© 2026 schoolict.net (Ruwan Mangala Suraweera). සියලු හිමිකම් ඇවිරිණි.