සියලු ම හිමිකම් ඇවිරිණි /All Rights Reserved]

Ravindu Bandaranayake - ictfromabc

Information and Communication Technology
තොරතුරු හා සන්නිවේදන තාක්ෂණය

2026 QUIZ NO 383 MARKING

2025/12/28
ictfromabc
Ravindu Bandaranayake

1. What was the first developed full size stored program computer?
ලොව ප්‍රථම පූර්ණ ප්‍රමාණයේ ආචිත ක්‍රමලේඛන පරිගණකය කුමක්ද?

1) ENIAC 2) EDSAC 3) EDVAC 4) UNIVAC 5) MARK 1

Answer : 2)

Let's go through the options provided as answers.
පිළිතුරු ලෙස සපයා ඇති විකල්ප හරහා යමු.

- **ENIAC (Electronic Numerical Integrator and Computer)**
  Developed in the 1940s, it was one of the first electronic general-purpose computers but did not use the stored-program concept; it was programmed by plugging and unplugging cables.
  1940 දී සංවර්ධනය කරන ලද, එය පළමු ඉලෙක්ට්‍රොනික පොදු කාර්ය පරිගණක වලින් එකක් වූ නමුත් ගබඩා කරන ලද වැඩසටහන් සංකල්පය භාවිතා නොකළ අතර එය ක්‍රමලේඛනය කර ඇත්තේ කේබල් සන්ධි කිරීම සහ විසන්ධි කිරීම මගිනි.

- **EDSAC (Electronic Delay Storage Automatic Calculator)**
  EDSAC was the first full size stored program computer developed by Maurice Wilkes in 1947. It was built according to the von Neumann machine principles.
  EDSAC යනු 1947 දී Maurice Wilkes විසින් සංවර්ධනය කරන ලද ප්‍රථම පූර්ණ ප්‍රමාණයේ ගබඩා කරන ලද වැඩසටහන් පරිගණකයයි. එය වොන් නියුමන් යන්ත්‍ර මූලධර්මවලට අනුව ගොඩනගා ඇත.

- **EDVAC (Electronic Discrete Variable Automatic Computer)**
  EDVAC is considered as the first digital computer that could store program was built in 1948.
  EDVAC, 1948 දී නිපදවන ලද වැඩසටහන් ගබඩා කළ හැකි පළමු අංකිත පරිගණකය ලෙස සැලකේ

- **UNIVAC (Universal Automatic Computer)**
  This was an electrical computer containing thousands of vacuum tubes that utilized punch cards. Used switches to input data and punch cards to output and store data.
  මෙය සිදුරු පත් භාවිතා කරන රික්තක නල දහස් ගණනක් අඩංගු විදුලි පරිගණකයකි. දත්ත ආදාන කිරීමට ස්විච සහ දත්ත ප්‍රතිදානය කිරීමට සහ ගබඩා කිරීමට සිදුරු පත් භාවිතා කරයි

- **MARK 1 (IBM Automatic Sequence Controlled Calculator)**
  Developed during World War II, it was an electromechanical computer and not a full-size stored-program computer.
  දෙවන ලෝක සංග්‍රාම සමයේදී සංවර්ධනය කරන ලද විද්‍යුත් යාන්ත්‍රික පරිගණකයක් මිස එය සම්පූර්ණ ප්‍රමාණයේ ගබඩා කරන ලද වැඩසටහන් පරිගණකයක් නොවේ.

So, the first developed full-size stored-program computer is generally credited to EDSAC.
මේ අනුව, ප්‍රථම පූර්ණ ප්‍රමාණයේ ගබඩා කරන ලද වැඩසටහන් පරිගණකය EDSAC වේ.

Therefore the answer is, 2).
එබැවින් පිළිතුර වන්නේ, 2) වේ.

2. What is the binary representation of decimal value 10.75 ?
දශමය අගය 10.75 හි ද්විමය නිරූපණය කුමක්ද?

1) 1011.11_2 2) 1010.01_2 3) $\underline{1010.11_2}$ 4) 1011.01_2 5) 1001.11_2

Answer : 3)

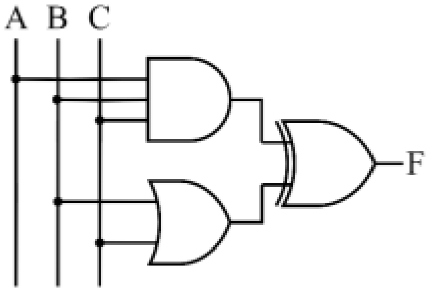
First Let's convert 10 into binary. මුලින්ම අපි 10 ද්විමය බවට පරිවර්තනය කරමු.							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	0	0	0	1	0	1	0
$10 = 1010_2$							

Now Let's convert 0.75 into binary දැන් අපි 0.75 ද්විමය බවට පරිවර්තනය කරමු	
.75	0.75 ×2 = 1.5
1.5	0.5 × 2 = 1.0
1.0	
$0.75 = 0.11_2$	

So the Binary representation of the Decimal value 10.75 is 1010.11_2
එබැවින් දශම අගය 10.75 හි ද්විමය නිරූපණය 1010.11_2 වේ.

Therefore the correct answer is 3
එබැවින් නිවැරදි පිළිතුර 3 වේ

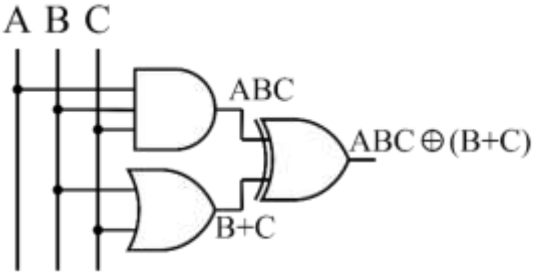
3. What is the answer when the Boolean expression for output F in the above logic circuit is simplified?
ඉහත තාර්කික පරිපථයේ F ප්‍රතිදානය සඳහා ලැබෙන බූලීය ප්‍රකාශනය සුළු කළ විට ලැබෙන පිළිතුර කුමක්ද?



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

Answer: 3)

First, let's obtained an expression from the above logical circuit.
පළමුව, ඉහත තාර්කික පරිපථයෙන් ප්‍රකාශනයක් ලබා ගනිමු.



Expression : $ABC \oplus B + C$	
: $\overline{ABC}.(B + C) + ABC.(\overline{B + C})$	By $A \oplus B = \overline{A}B + A\overline{B}$
: $\overline{ABC}.(B + C) + ABC.\overline{B}\overline{C}$	De morgan's Law ඩි මෝගන් න්‍යාය
: $(\overline{A} + \overline{B} + \overline{C}).(B + C) + ABC.\overline{B}\overline{C}$	De morgan's Law ඩි මෝගන් න්‍යාය
: $(\overline{A} + \overline{B} + \overline{C}).(B + C) + 0$	Inverse Law ප්‍රතිලෝම න්‍යාය
: $(\overline{A} + \overline{B} + \overline{C}).(B + C)$	Identity Law සර්වසාම්‍ය න්‍යාය
: $(\overline{A}B + \overline{B}B + \overline{C}B + \overline{A}C + \overline{B}C + \overline{C}C)$	Distributive Law විඝටන න්‍යාය
: $(\overline{A}B + 0 + \overline{C}B + \overline{A}C + \overline{B}C + \overline{C}C)$	Inverse Law ප්‍රතිලෝම න්‍යාය
: $(\overline{A}B + \overline{C}B + \overline{A}C + \overline{B}C + \overline{C}C)$	Identity Law සර්වසාම්‍ය න්‍යාය
: $(\overline{A}B + \overline{C}B + \overline{A}C + \overline{B}C + 0)$	Inverse Law ප්‍රතිලෝම න්‍යාය
: $(\overline{A}B + \overline{C}B + \overline{A}C + \overline{B}C)$	Identity Law සර්වසාම්‍ය න්‍යාය

Correct answer is **3)**.
නිවැරදි පිළිතුර **3)** වේ.

4. Which of the followings can be considered as the candidate key regarding the student relation?
ශිෂ්‍ය සම්බන්ධතාව සම්බන්ධයෙන් අපේක්ෂක යතුර ලෙස සැලකිය හැක්කේ පහත සඳහන් කවරක් ද?

`Student(Stu_ID,Stu_Name,Gender,Address,DOB,Class)`

1) `Stu_ID` 2) `Stu_Name` 3) `Gender` 4) `Address` 5) `DOB`

Answer: 1)

Explanation / පැහැදිලි කිරීම:

A candidate key is a minimal set of attributes that uniquely identifies a tuple (row) in a relation.
අපේක්ෂක යතුරක් යනු සම්බන්ධතාවයක පේළියක් අනන්‍ය ලෙස හඳුනා ගන්නා අවම ගුණාංග සමූහයකි.

1) **Stu_ID**
This is typically assigned uniquely to each student, ensuring that no two students have the same ID. This makes it suitable as a candidate key.
මෙය සාමාන්‍යයෙන් සෑම ශිෂ්‍යයෙකුටම අනන්‍ය ලෙස පවරනු ලැබේ, සිසුන් දෙදෙනෙකුට එකම හැඳුනුම්පත නොමැති බව සහතික කරයි. මෙය අපේක්ෂක යතුරක් ලෙස සුදුසු වේ.

2) **Stu_Name**
Student names can be duplicated (ex: multiple students with the same name).
ශිෂ්‍ය නම් අනුපිටපත් විය හැක (උදා: එකම නම සහිත සිසුන් කිහිප දෙනෙකු).

3) **Gender**
Gender values can be repeated (ex: multiple female or male students).
ස්ත්‍රී පුරුෂ භාවය නැවත නැවතත් අනුපිටපත් විය හැක (උදා: කාන්තා හෝ පිරිමි සිසුන් කිහිපයක්).

4) **Address**
Addresses can also be shared by multiple students (ex: roommates, siblings).
එකම ලිපිනය සිසුන් කිහිප දෙනෙකුට ද බෙදා ගත හැකිය (උදා: නේවාසික මිතුරන්, සහෝදර සහෝදරියන්).

5) **DOB**
Dates of birth can be duplicated (ex: students born on the same day).
උපන් දිනයන් අනුපිටපත් විය හැක (උදා: එකම දිනයේ උපන් සිසුන්).

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

5. What is the correct SQL query to retrieve the details of all columns and rows from a table named Students?
Students නම් වගුවකින් සියලුම තිරු සහ පේළි වල විස්තර ලබා ගැනීමට අදාළ නිවැරදි SQL විමසුම කුමක්ද?

1) `SELECT * FROM Students;`
2) `GET ALL FROM Students;`
3) `FETCH ALL FROM Students;`
4) `SHOW TABLE Students;`
5) `SELECT ALL FROM Students;`

Answer: 1)

Explanation of each option / එක් එක් විකල්පය පැහැදිලි කිරීම:

1) **`SELECT * FROM Students;`**
The `SELECT` ***** statement in SQL is used to retrieve all columns and rows from a specified table. * is a wildcard character that represents all columns in the table.

SQL හි `SELECT *` ප්‍රකාශය නිශ්චිත වගුවකින් සියලුම තීරු සහ පේළි ලබා ගැනීමට භාවිතා කරයි. * යනු වගුවේ සියලුම තීරු නියෝජනය කරන wildcard අක්ෂරයකි.

`FROM Students` specifies the table Students, from which the data should be retrieved.
`FROM Students` දත්ත ලබා ගත යුතු වගුව Students නියම කරයි.

2) **`GET ALL FROM Students;`**
This is incorrect because `GET` is not a valid SQL command for retrieving data. The correct command is `SELECT`.
මෙය වැරදියි, මන්ද `GET` දත්ත ලබා ගැනීම සඳහා වලංගු SQL විධානයක් නොවන බැවිනි. නිවැරදි විධානය `SELECT` වේ.

3) **`FETCH ALL FROM Students;`**
This is incorrect because `FETCH` is not used for querying a table directly. It is typically used in conjunction with cursors for fetching rows one at a time or in blocks.
`FETCH` වගුවක් සෘජුවම විමසුම සඳහා භාවිතා නොකරන බැවින් මෙය වැරදියි. එය සාමාන්‍යයෙන් පේළි එකින් එක ලබා ගැනීම සඳහා කර්සර සමඟ ඒකාබද්ධව භාවිතා වේ.

4) **`SHOW TABLE Students;`**
This is incorrect because `SHOW TABLE` is not valid syntax for querying data. The `SHOW TABLES` command in some databases (like MySQL) is used to list all tables in a database, but it does not retrieve rows or columns.
`SHOW TABLE` දත්ත විමසුම සඳහා වලංගු වාක්‍ය ඛණ්ඩයක් නොවන බැවින් මෙය වැරදියි. සමහර දත්ත සමුදායන්හි (MySQL වැනි) `SHOW TABLES` විධානය දත්ත සමුදායක සියලුම වගු ලැයිස්තුගත කිරීමට භාවිතා කරයි, නමුත් එය පේළි හෝ තීරු ලබා නොගනී.

5) **`SELECT ALL FROM Students;`**
This is incorrect because `ALL` is not valid syntax in this context. The correct wildcard for selecting all columns is *.
මෙම සන්දර්භය තුළ `ALL` වලංගු වාක්‍ය ඛණ්ඩයක් නොවන බැවින් මෙය වැරදියි. සියලුම තීරු තේරීම සඳහා නිවැරදි wildcard එක * වේ.

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

6. What is the correct SQL code to create a database as Student?
Student ලෙස දත්ත සමුදායක් නිර්මාණය කිරීමට අදාළ නිවැරදි SQL කේතය වන්නේ කුමක්ද?

1) `CREATE Student;`
2) `CREATE SCHEMA Student,`
3) `CREATE TABLE Student`
4) <u>`CREATE DATABASE Student;`</u>
5) `INSERT INTO Student;`

Answer: 4)

The SQL statement to create a new database is CREATE DATABASE, followed by the name of the database. In this case, Student is the database name.
නව දත්ත සමුදායක් නිර්මාණය කිරීම සඳහා වන SQL ප්‍රකාශය CREATE DATABASE වන අතර, පසුව දත්ත සමුදායේ නම ඇත. මෙම අවස්ථාවේදී, Student යනු දත්ත සමුදායේ නමයි.

The correct answer is 4) CREATE DATABASE Student;
නිවැරදි පිළිතුර නම්, 4) CREATE DATABASE Student;

Why other options are incorrect / වෙනත් විකල්ප වැරදි ඇයි,

Option / විකල්ප 1)
Invalid as it does not specify what is being created.
නිර්මාණය කරන්නේ කුමක්ද යන්න නිශ්චිතව දක්වා නොමැති බැවින් වලංගු නොවේ.

Option / විකල්ප 2)
Incorrect syntax due to the comma.
කොමාව නිසා වැරදි වාක්‍ය ඛණ්ඩයකි.

Option / විකල්ප 3)
Used to create a table, not a database.
දත්ත සමුදායක් නොව වගුවක් නිර්මාණය කිරීමට භාවිතා කරයි.

Option / විකල්ප 5)
Used to insert data into an existing table, not to create a database.
දත්ත සමුදායක් නිර්මාණය කිරීමට නොව පවතින වගුවකට දත්ත ඇතුළු කිරීමට භාවිතා කරයි.

7. What is a computer language that does not use an interpreter?
අර්ථවින්‍යාසක පරිවර්තකයක් භාවිතා නොකරන පරිගණක භාෂාවක් වන්නේ කුමක්ද?

1) Python 2) Ruby 3) Pascal 4) JavaScript 5) PHP

Answer: 3)

Let's consider each option.
එක් එක් විකල්පයන් සලකා බලමු.

Python
This is an interpreted language, meaning it uses an interpreter to execute the code line by line.
මෙය අර්ථ වින්‍යාසක භාෂාවකි, එනම් එය පේළියෙන් පේළිය ක්‍රියාත්මක කිරීමට අර්ථ වින්‍යාසකයක් (interpreter) භාවිතා කරයි.

Ruby
Like Python, Ruby is also an interpreted language.
පයිතන් මෙන්ම Ruby ද අර්ථ වින්‍යාසක (interpreter) භාෂාවකි.

Pascal
This is primarily a compiled language, which means that it is translated into machine code by a compiler before execution, not using an interpreter.
මෙය මූලික වශයෙන් සම්පාදනය (compiled) කරන ලද භාෂාවකි, එයින් අදහස් වන්නේ එය අර්ථ වින්‍යාසකයක් (interpreter) භාවිතා නොකර ක්‍රියාත්මක කිරීමට පෙර සම්පාදකයක් (compiler) මගින් යන්ත්‍ර කේතයට පරිවර්තනය කරන බවයි.

JavaScript
Although it can be executed in an interpreted manner in browsers, it is often compiled just-in-time (JIT) for performance.
එය අතිරික්සුවල අර්ථ වින්‍යාසකය (interpreter) ආකාරයෙන් ක්‍රියාත්මක කළ හැකි වුවද, එය බොහෝ විට කාර්ය සාධනය සඳහා නියමිත වේලාවට (just-in-time – (JIT)) සම්පාදනය කෙරේ.

PHP
PHP is an interpreted language, executing scripts line by line at runtime.
PHP යනු අර්ථ වින්‍යාසය (interpreter) කරන ලද භාෂාවක් වන අතර, ධාවන වේලාවේදී ස්ක්‍රිප්ට් එකේ පේළියෙන් පේළිය ක්‍රියාත්මක කරයි.

In summary, among the options listed, Pascal stands out as the language that does not use an interpreter in its standard implementation.
සාරාංශයක් ලෙස, ලැයිස්තුගත කර ඇති විකල්ප අතර, **Pascal** එහි සම්මත ක්‍රියාත්මක කිරීමේදී අර්ථ වින්‍යාසකය (interpreter) භාවිතා නොකරන භාෂාව ලෙස කැපී පෙනේ.

Correct answer is **3)**.
නිවැරදි පිළිතුර **3)** වේ.

8. What will be the output of the following code?
පහත කේතයේ ප්‍රතිදානය කුමක් වේද?

```
a, b = 10, 20
if a + b == 30 and a > b:
    print("Condition 1")
elif a < b or b - a == 10:
    print("Condition 2")
else:
    print("Condition 3")
```

1) Condition 1 2) Condition 2 3) Condition 3 4) No output 5) Error

Answer : 2)

Explanation / පැහැදිලි කිරීම

In the given code, the variables a and b are assigned the values 10 and 20, respectively. The first condition checks if a + b == 30 and a > b. While a + b == 30 is true since 10 + 20 equals 30, the condition a > b is false because 10 is not greater than 20. Since both conditions are connected with and, the entire first condition evaluates to false, and "Condition 1" is not printed.
ලබා දී ඇති කේතයෙහි, a සහ b යන විචල්‍යයන්ට පිළිවෙලින් 10 සහ 20 අගයන් පවරනු ලැබේ. පළමු කොන්දේසිය a + b == 30 සහ a > b දැයි පරීක්ෂා කරයි. 10 + 20 30ට සමාන බැවින් a + b == 30 සත්‍ය වන අතර, 10 20 ට වඩා වැඩි නොවන නිසා a > b කොන්දේසිය අසත්‍ය වේ. කොන්දේසි දෙකම and සමඟ සම්බන්ධ වී ඇති බැවින්, සම්පූර්ණ පළමු කොන්දේසිය අසත්‍ය ලෙස තක්සේරු කරයි, සහ "Condition 1" මුද්‍රණය නොකෙරේ.

The second condition, written under elif, checks if a < b or b - a == 10. The expression a < b evaluates to true because 10 is less than 20, and since the or operator requires only one part to be true, this condition becomes true. As a result, "Condition 2" is printed, and the else block does not execute. Therefore, the final output of the code is "Condition 2", making option 2 the correct answer.
elif යටතේ ලියා ඇති දෙවන කොන්දේසිය, a < b හෝ b - a == 10 දැයි පරීක්ෂා කරයි. 10 20ට වඩා අඩු නිසා a < b ප්‍රකාශය සත්‍ය ලෙස ඇගයීමට ලක් කරයි, සහ or ක්‍රියාකරුට සත්‍ය වීමට එක් කොටසක් පමණක් අවශ්‍ය බැවින්, මෙම කොන්දේසිය සත්‍ය වේ. එහි ප්‍රතිඵලයක් ලෙස, "Condition 2", මුද්‍රණය කර ඇති අතර, else කොටස ක්‍රියාත්මක නොවේ. එබැවින්, කේතයේ අවසාන ප්‍රතිදානය "Condition 2" වේ, 2 විකල්පය නිවැරදි පිළිතුර බවට පත් කරයි.

Therefore the Correct answer is 2)
එබැවින් නිවැරදි පිළිතුර 2) වේ

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

```
print(type(42))
```

1) <class 'int'>
2) <class 'str'>
3) <class 'float'>
4) <class 'bool'>
5) <class 'complex'>

Answer : 1)

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

1) <class 'int'> (Correct)
The number 42 is an integer, and the type() function identifies it as an object of type int. Therefore, the correct output will be <class 'int'>.

අංක 42 යනු පූර්ණ සංඛ්‍යාවක් වන අතර type() ශ්‍රිතය එය int වර්ගයේ වස්තුවක් ලෙස හඳුනා ගනී. එබැවින් නිවැරදි ප්‍රතිදානය <class 'int'>.

2) **<class 'str'>**
The number 42 is not a string (str), so this type would not be returned. A string would be something like "42", which would yield <class 'str'>.
අංක 42 string (str) එකක් නොවේ, එබැවින් මෙම වර්ගය ආපසු ලබා නොදෙනු ඇත. string එකක් 42 වැනි දෙයක් වනු ඇත, එය <class 'str'>.ලබා දෙනු ඇත.

3) **<class 'float'>**
42 is an integer, not a floating-point number. A floating-point number would be written as 42.0, and only then would the type returned be <class 'float'>.
42 යනු පූර්ණ සංඛ්‍යාවක් මිස floating-point සංඛ්‍යාවක් නොවේ. floating-point අංකයක් 42.0 ලෙස ලියා ඇති අතර, ආපසු එන වර්ගය <class 'float'> වනු ඇත.

4) **<class 'bool'>**
A boolean value in Python is either True or False. 42 is'nt a boolean value, so this option is incorrect.
Python හි boolean True සත්‍ය හෝ False වේ. 42 යනු boolean අගයක් නොවේ, එබැවින් මෙම විකල්පය වැරදියි.

5) **<class 'complex'>**
A complex number in Python would look like 42 + 0j. Since 42 is a simple integer and not a complex number, this type is incorrect.
Python හි සංකීර්ණ සංඛ්‍යාවක් ලෙස 42 + 0j පෙනෙනු ඇත. 42 යනු සරල පූර්ණ සංඛ්‍යාවක් මිස සංකීර්ණ සංඛ්‍යාවක් නොවන බැවින්, මෙම වර්ගය වැරදිය.

So, the correct answer is 1).
එබැවින්, නිවැරදි පිළිතුර 1) වේ.

10. Which of the following correctly describes the behavior of the <br> tag?

 උසුලනයේ හැසිරීම නිවැරදිව විස්තර කරන්නේ පහත සඳහන් කුමක්ද?

1) It is a semantic tag that structures paragraphs in HTML documents.
එය HTML ලේඛනවල ඡේද ව්‍යුහගත කරන අර්ථකථන උසුලනයකි.

2) It creates a line break without adding any semantic meaning.
එය කිසිදු අර්ථකථන අර්ථයක් එකතු නොකර පේළි බිඳීමක් නිර්මාණය කරයි.

3) It inserts a visible horizontal rule in the document.
එය ලේඛනය තුළ දෘශ්‍යමාන තිරස් රීතියක් ඇතුළත් කරයි.

4) It is replaced by <p> in modern HTML versions.
එය නවීන HTML අනුවාදවල <p> මගින් ප්‍රතිස්ථාපනය වේ.

5) It adds spacing between two HTML elements.
එය HTML මූලාංග දෙකක් අතර පරතරයක් එක් කරයි.

Answer : 2)

The <br> tag is used to create a line break in an HTML document without adding any semantic meaning to the content. It simply moves the text following the tag to a new line, providing a visual break without affecting the structure or meaning of the document. This makes <br> particularly useful for formatting content like addresses, poems, or other text requiring line breaks within a single paragraph. As it doesn't add any additional functionality or meaning, option 2 correctly describes the behavior of the <br> tag.

 උසුලනය HTML ලේඛනයක අන්තර්ගතයට කිසිදු අර්ථකථන අර්ථයක් එක් නොකර රේඛා බිඳීමක් නිර්මාණය කිරීමට භාවිතා කරයි. එය ලේඛනයේ ව්‍යුහයට හෝ අර්ථයට බලපෑමක් නොකර දෘශ්‍ය විරාමයක් සපයමින් ලේඛනයට පසු පෙළ නව රේඛාවකට ගෙන යයි. මෙය
 විශේෂයෙන් ලිපින, කවි හෝ තනි ඡේදයක් තුළ රේඛා බිඳීම් අවශ්‍ය වෙනත් පෙළ වැනි අන්තර්ගත හැඩතල ගැන්වීම සඳහා ප්‍රයෝජනවත් කරයි. එය අමතර ක්‍රියාකාරීත්වයක් හෝ අර්ථයක් එක් නොකරන බැවින්, විකල්ප 2
 උසුලනයේ හැසිරීම නිවැරදිව විස්තර කරයි.