සියලු ම හිමිකම් ඇවිරිණි / All Rights Reserved]

Bandaranayake - ictfromabc Ravindu Bandaranayake - ictfromabc Ravindu

Information and Communication Technology
තොරතුරු හා සන්නිවේදන තාක්ෂණය

2026 QUIZ NO 254 MARKING

2025/08/21
ictfromabc
Ravindu Bandaranayake

1. What is a type of memory, in which data can only be written once and cannot be erased?
දත්ත එක් වරක් පමණක් ලිවිය හැකි සහ මකා දැමිය නොහැකි ආකාරයේ මතකයක් වන්නේ පහත ඒවායින් කුමක්ද?

1) <u>PROM</u> 2) EPROM 3) EEPROM 4) RAM 5) None of the above ඉහත කිසිවක් නොවේ

Answer: 1)

Explanation / පැහැදිලි කිරීම:

The correct answer is PROM. PROM is a type of memory that is programmed once by a manufacturer or user, typically during the manufacturing process.
නිවැරදි පිළිතුර PROM වේ. PROM යනු නිෂ්පාදකයෙකු හෝ පරිශීලකයෙකු විසින් සාමාන්‍යයෙන් නිෂ්පාදන ක්‍රියාවලියේදී එක් වරක් වැඩසටහන්ගත කරන ලද මතක වර්ගයකි.

Once programmed, the data stored in PROM cannot be altered or erased. PROM retains its programmed data even when the power is turned off.
වැඩසටහන්ගත කළ පසු, PROM හි ගබඩා කර ඇති දත්ත වෙනස් කිරීමට හෝ මැකීමට නොහැකිය. PROM බලය අක්‍රිය වූ විට පවා එහි වැඩසටහන් දත්ත රඳවා තබා ගනී.

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

2. Which of the following statements is true?
පහත දැක්වෙන කුමන වගන්තිය සත්‍ය වේ ද?

1) Charles Babbage introduced the logarithm theory.
චාල්ස් බැබේජ් විසින් ලඝුගණක සිද්ධාන්ත හඳුන්වා දෙන ලදී.
2) William Oughtred improved the adding machine and built the stepper so that it could also perform multiplication and division.
විලියම් අවුට්‍රඩ් විසින් ආකලන යන්ත්‍රය වැඩි දියුණු කර ගුණ කිරීම සහ බෙදීම ද සිදුකල හැකි පරිදි පියවර ගණකය සැදුවේ ය.
3) <u>Herman Hollerith took the first American census using punch cards.</u>
හර්මන් හොලෙරිත් විසින් සිදුරුපත් ආධාරයෙන් ප්‍රථම ඇමරිකානු ජන සංගණනය සිදුකළේය.
4) Joseph Jacquard was the world's first programmer.
ජෝශප් ජැක්වාඩ් ලොව ප්‍රථම ක්‍රමලේඛන ශිල්පියා විය.
5) Gottfried Wilhelm Leibniz invented the adding machine.
ගොට්ෆ්‍රිඩ් විල්හෙල්ම් ලිබ්නිස් විසින් ආකලන යන්ත්‍රය නිර්මාණය කරන ලදී.

Answer: 3)

Explanation / පැහැදිලි කිරීම:

- Logarithm theory was introduced to the world by John Napier.
  ලඝු ගණක සිද්ධාන්ත ලොවට හඳුන්වා දෙන ලද්දේ ජෝන් නේපියර් විසිනි.
- Gottfried Wilhelm Leibniz built the step calculator to improve the adding machine and perform multiplication and division.
  ආකලන යන්ත්‍රය වැඩිදියුණු කර ගුණ කිරීම සහ බෙදීම සිදු කළ හැකි පරිදි පියවර ගණකය සාදන ලද්දේ ගොට්ෆ්‍රිඩ් විල්හෙල්ම් ලීබ්නිස් විසිනි.
- Herman Hollerith conducted the first American census using punch cards.
  සිදුරුපත් ආධාරයෙන් ප්‍රථම ඇමරිකානු ජන සංගණනය සිදුකරන ලද්දේ හර්මන් හොලෙරිත් විසිනි.
- The world's first programmer was Ada Augusta.
  ලොව ප්‍රථම ක්‍රමලේඛන ශිල්පියා වුයේ ඒඩා ඔගස්ටාය.
- The adding machine was invented by Blaise Pascal, a French scientist.
  ආකලන යන්ත්‍රය නිර්මාණය කළේ ප්‍රංශ ජාතික විද්‍යාඥයෙකු වූ බ්ලේස් පැස්කල් විසිනි.

Therefore, the correct answer is 3).
එබැවින් නිවැරදි පිළිතුර 3) වේ.

3. If the hexadecimal number 1A3F is converted to binary, how many bits are required to represent it?
1A3F යන ෂඩ්දශමය සංඛ්‍යාව ද්විමය සංඛ්‍යාවක් බවට පරිවර්තනය කළහොත්, එය නිරූපණය කිරීමට බිටු කියක් අවශ්‍ය වේද?

1) 12 bits 2) 14 bits 3) <u>13 bits</u> 4) 15 bits 5) 16 bits

Answer: 3)

Explanation / පැහැදිලි කිරීම:

Converting Hexadecimal to Binary			
$1A3F_{16}$			
1	A	3	F
0001	1010	0011	1111
0001101000111111_2			

According to this table, 16 bits are required to represent $1A3F_{16}$ as a binary number. But since the first 3 bits are '0', they have no effect on the corresponding binary value. So the answer is 13 bits.
මෙම වගුවට අනුව $1A3F_{16}$ ද්විමය සංඛ්‍යාවක් මගින් දැක්වීමට බිටු 16ක් අවශ්‍ය වේ. නමුත් මුල් බිටු 3ම '0' බැවින් ඒවාගෙන් අදාළ ද්විමය අගයට බලපෑමක් නොමැත. එම නිසා පිළිතුර වන්නේ බිටු 13ක් වේ.

Bits 12, 14, 15 are not suitable for this at all.
කොහෙත්ම මේ සඳහා බිටු 12,14,15 නොගැළපේ.

Therefore, the correct answer is 3).
එබැවින් නිවැරදි පිළිතුර 3) වේ.

4. What is the bitwise AND of 11100011_2 and x ?
Here x is the bitwise XOR of 11110000_2 and 10101010_2
11100011_2 සහ x හි බිටු අනුසාරිත AND මෙහෙයුම සිදුකිරීමෙන් ලැබෙන පිළිතුර කුමක්ද?
මෙහි x යනු 11110000_2 සහ 10101010_2 හි බිටු අනුසාරිත XOR මෙහෙයුමෙහි පිළිතුර වේ.

1) 11100010_2 2) 10111101_2 3) <u>01000010_2</u> 4) 01000000_2 5) 11100011_2

Answer: 3)

Explanation / පැහැදිලි කිරීම:

It has given that x is the bitwise XOR of 11110000_2 and 10101010_2. So, let's find x.
x යනු 11110000_2 සහ 10101010_2 හි බිටු අනුසාරිත XOR බව ලබා දී ඇත. එබැවින් x සොයමු.

1	1	1	1	0	0	0	0
1	0	1	0	1	0	1	0
0	1	0	1	1	0	1	0

Truth table for the XOR gate
XOR ද්වාරය සඳහා සත්‍යතා වගුව

A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

So, x is 01011010_2. Now let's find the bitwise AND of 11100011_2 and x.
x යනු 01011010_2. දැන් 11100011_2 සහ x හි බිටු අනුසාරීත AND සොයමු.

1	1	1	0	0	0	1	1
0	1	0	1	1	0	1	0
0	1	0	0	0	0	1	0

Truth table for the AND gate
AND ද්වාරය සඳහා සත්‍යතා වගුව

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

Therefore, the answer is 3) 01000010_2.
එබැවින් පිළිතුර 3) 01000010_2 වේ.

5. Which places must be 1 in the below Karnaugh map to obtain the expression $B + A\bar{C}$?
$B + A\bar{C}$ යන ප්‍රකාශනය ලබා ගැනීමට පහත කානෝ සිතියමේ 1 විය යුත්තේ කවර ස්ථානද?

B \ AC	01	11	10	00
0	P	R	T	V
1	Q	S	U	W

1) P, Q, R, S, T
2) Q, S, T, U, W
3) R, S, T, U, V
4) P, R, T, V, W
5) P, Q, T, U, V

Answer: 2)

Explanation / පැහැදිලි කිරීම:

According to the above Karnaugh map $B + A\bar{C}$ means B or A and $\bar{C}$. Accordingly, all the points where B=1 and all the points where $A\bar{C}$ =1 should be set to 1 in the Karnaugh map.
ඉහත කානෝ සිතියමට අනුව ලබා දී ඇති $B + A\bar{C}$ යන්නෙන් අදහස් වන්නේ B හෝ A සහ $\bar{C}$ යන්නයි. ඒ අනුව B=1 වන ස්ථාන සියල්ලමත් $A\bar{C}$ =1 වන සියළුම ස්ථානත් කානෝ සිතියම තුළ 1 කළ යුතු වේ.

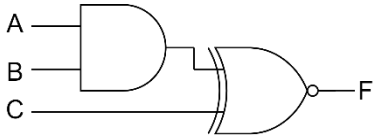
Accordingly, we can derive the following Karnaugh map.
ඒ අනුව, පහත කානෝ සිතියම අපට ව්‍යුත්පන්න කර ගත හැක.

B \ AC	01	11	10	00
0	0	0	1	0
1	1	1	1	1

Accordingly, it can be recognized that the places Q, S, T, U, and W should be 1 on the Karnaugh map.
ඒ අනුව කානෝ සිතියමේ Q, S, T, U, සහ W යන ස්ථාන 1 විය යුතු බව හඳුනා ගත හැක.

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

6. What is the correct K-map equivalent to the following circuit?
පහත පරිපථයට තුල්‍ය වන නිවැරදි කානෝ සිතියම කුමක්ද?



1)

B \ AC	11	01	00	10
1	1	0	1	0
0	0	0	1	1

2)

A \ BC	11	01	00	10
1	1	0	0	0
0	0	1	1	1

3)

A \ BC	00	01	11	10
0	1	0	0	0
1	0	1	1	1

4)

A \ BC	00	01	11	10
0	1	1	0	1
1	0	0	1	0

5)

C \ AB	00	01	11	10
0	0	0	1	0
1	1	1	0	1

Answer: 1)

Explanation / පැහැදිලි කිරීම:

The truth table corresponding to the above circuit is shown below.
ඉහත පරිපථයට අනුරූප වන සත්‍යතා වගුව පහතින් දැක්වේ.

A	B	C	A.B	$\overline{(A.B) \oplus C}$
0	0	0	0	1
0	0	1	0	0
0	1	0	0	1
0	1	1	0	0
1	0	0	0	1
1	0	1	0	0
1	1	0	1	0
1	1	1	1	1

Only Answer 1 draws a Karnaugh map that correctly maps to the truth table.
සත්‍යතා වගුවට නිවැරදිව අනුරූපණය වන ආකාරයට කානෝ සිතියමක් ඇඳ ඇත්තේ 1 පිළිතුර මගින් පමණි.

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

7. What will be the result of the following Python code?
පහත පයිතන් කේතයේ ප්‍රතිඵලය කුමක් වේද?

```
def mystery(x, y):
    x = x + y
    y = y + x
    return x, y
x, y = 10, 20
result = mystery(x, y)
print(x, y, result)
```

1) 10 20 (30, 50) 2) 30 20 (30, 50) 3) 10 20 (30, 30)
4) 30 50 (30, 50) 5) 10 20 (20, 40)

Answer: 1)

This Python code defines a function `mystery` that takes two parameters `x` and `y`. Inside the function, it first updates `x` to be the sum of the original x and y, and then updates `y` to be the sum of the updated y and the new value of `x`.
මෙම පයිතන් කේතය `x` සහ `y` පරාමිති දෙකක් ගන්නා ශ්‍රිතය `mystery` නිර්වචනය කරයි. ශ්‍රිතය තුළ, එය මුලින්ම `x` යනු මුල් `x` සහ `y` හි එකතුව ලෙස යාවත්කාලීන කරයි, පසුව `y` යාවත්කාලීන කරන ලද `y` හි එකතුව සහ `x` හි නව අගය ලෙස යාවත්කාලීන කරයි.

After executing the function, it returns the new values of x and y. When the function is called with `x = 10` and `y = 20`, inside the function, `x` becomes 10 + 20 = 30, and y becomes 20 + 30 = 50.
ශ්‍රිතය ක්‍රියාත්මක කිරීමෙන් පසුව, එය `x` සහ `y` හි නව අගයන් ලබා දෙයි. ශ්‍රිතය `x = 10` සහ `y = 20` සමඟ හැඳින්වූ විට, ශ්‍රිතය තුළ, `x` 10 + 20 = 30 බවටත්, `y` 20 + 30 = 50 බවටත් පත් වේ.

However, these changes are local to the function and do not affect the variables outside it. Thus, when printing x and y after the function call, they remain as their original values, 10 and 20, respectively.
කෙසේ වෙතත්, මෙම වෙනස්කම් ශ්‍රිතයට local වන අතර ඉන් පිටත විචල්‍යයන්ට බලපාන්නේ නැත. මේ අනුව, ශ්‍රිත ඇමතුමෙන් පසුව `x` සහ `y` මුද්‍රණය කරන විට, ඒවා පිළිවෙලින් 10 සහ 20 ලෙස ඒවායේ මුල් අගයන් ලෙස පවතී.

The `result` variable captures the returned tuple `(30, 50).` Therefore, the output of the code will be `10 20 (30, 50).`
ප්‍රතිඵල විචල්‍යය මඟින් ආපසු ලබා දුන් ටියුපල් `(30, 50)` ග්‍රහණය කරයි. එබැවින්, කේතයේ ප්‍රතිදානය `10 20 (30, 50)` වනු ඇත.

Therefore, the correct answer is 1) `10 20 (30, 50)`
එබැවින්, නිවැරදි පිළිතුර 1) `10 20 (30, 50)` වේ.

8. What will be the result of the following python code?
පහත පයිතන් කේතයේ ප්‍රතිඵලය කුමක් වේවිද?

```
def find(a, b):
    return (a**2 + b**2) / (a + b)

result = find(3, 4)
print(round(result, 2))
```

1) 3.50 2) 3.57 3) 7.00 4) 12.50 5) 25.00

Answer: 2)

Explanation / පැහැදිලි කිරීම:

The given Python code defines a function `find` that takes two arguments, `a` and `b`, and computes the value of the expression `(a**2 + b**2) / (a + b)`.
ලබා දී ඇති පයිතන් කේතය `a` සහ `b` යන තර්ක දෙකක් ගන්නා ශ්‍රිතයක් සොයා ගැනීම නිර්වචනය කරයි, සහ `(a**2 + b**2) / (a + b)` ප්‍රකාශනයේ අගය ගණනය කරයි.

For the input values 3 and 4, the function calculates $(3^2 + 4^2) / (3 + 4)$, which simplifies to $(9 + 16) / 7$ or $(25 / 7)$. This results in approximately 3.5714285714285716.
ආදාන අගයන් 3 සහ 4 සඳහා, ශ්‍රිතය $(3^2 + 4^2) / (3 + 4)$ ගණනය කරයි, එය $(9 + 16) / 7$ හෝ $(25 / 7)$ වෙත සරල කරයි. මෙහි ප්‍රතිඵලය ආසන්න වශයෙන් 3.5714285714285716 වේ.

When this result is rounded to two decimal places using `round(result,2),` the output is `3.57`. Therefore, the final printed result of the code will be `3.57`.
මෙම ප්‍රතිඵලය `round(result,2)`භාවිතයෙන් දශම ස්ථාන දෙකකට වට කළ විට ප්‍රතිදානය `3.57` වේ. එබැවින්, කේතයේ අවසන් මුද්‍රිත ප්‍රතිඵලය `3.57` වනු ඇත.

Therefore, the correct answer is 2) `3.57`
එබැවින්, නිවැරදි පිළිතුර 2) `3.57` වේ.

9. What is the output of the following python program?
පහත පයිතන් වැඩසටහනේ ප්‍රතිදානය කුමක්ද?

```
n = 4
fact = 1
i = 2
while i <= n:
 fact = fact*i
 i = i+1
print (fact)
```

1) `6` 2) `120` 3) `24` 4) `2` 5) `1`

Answer: 3)

Explanation / පැහැදිලි කිරීම:

This Python program calculates the factorial of a number, in this case, `4`. It initializes `fact` to `1` and starts a while loop with `i` set to `2`.
මෙම පයිතන් ක්‍රමලේඛය මගින් සංඛ්‍යාවක සාධක අගය ගණනය කරයි, මෙම අවස්ථාවෙහිදී, 4. එය `fact` 1 ට ආරම්භ කරන අතර `i`, 2 ට සකසන ලද අතරමග ලූපයක් ආරම්භ කරයි.

As long as `i` is less than or equal to `n`, the loop multiplies `fact` by `i` and increments `i` by 1.
`i`, `n` ට වඩා අඩු හෝ සමාන වන තාක්, ලූපය `fact`, `i` මගින් ගුණ කරන අතර `i`, `1` න් වැඩි කරයි.

The loop iterates from `i = 2` to `i = 4`, so the calculations are as follows: `fact` starts at `1`, then is multiplied by `2` (resulting in `2`), then by `3` (resulting in `6`), and finally by `4` (resulting in `24`).
ලූපය `i = 2` සිට `i = 4` දක්වා පුනරාවර්තනය වේ, එබැවින් ගණනය කිරීම් පහත පරිදි වේ: `fact` `1` න් ආරම්භ වේ, පසුව `2` න් ගුණ කරනු ලැබේ (ප්‍රතිඵලය `2`), ඉන්පසු `3` (ප්‍රතිඵලය `6`) සහ අවසානයේ `4` (ප්‍රතිඵලය `24`).

After the loop completes, the program prints `24`, which is the factorial of `4`.
ලූපය සම්පූර්ණ වූ පසු, වැඩසටහන `24` මුද්‍රණය කරයි, එය `4` හි සාධක වේ.

Therefore, the correct answer is 3) `24`.
එබැවින්, නිවැරදි පිළිතුර 3) `24` වේ.

10. What will be the output of the below Python code?
පහත පයිතන් කේතයේ ප්‍රතිදානය කුමක් වේවිද?

```
my_dict = {'a': [1, 2], 'b': [3, 4]}
my_dict['a'].append(3)
my_dict['b'] = my_dict['a'] + my_dict['b']
print(my_dict)
```

1) `{'a': [1, 2, 3], 'b': [3, 4, 3]}`
2) `{'a': [1, 2, 3], 'b': [1, 2, 3, 3]}`
3) `{'a': [1, 2, 3], 'b': [1, 2, 3, 4]}`
4) `{'a': [1, 2, 3], 'b': [1, 2, 3, 4, 3]}`
5) `{'a': [1, 2, 3], 'b': [1, 2, 3, 3, 4]}`

Answer: 5)

Let's analyze the Python code given.
ලබා දී ඇති පයිතන් කේතය විශ්ලේෂණය කරමු.

```
my_dict = {'a': [1, 2], 'b': [3, 4]}
```

Initializes a dictionary named my_dict with two keys a and b .
a සහ b ලෙස යතුරු අගයන් දෙකකින් යුත් my_dict නමින් dictionary එකක් ආරම්භ කරයි.

```
my_dict['a'].append(3)
```

Here, the append() method is used to add the value 3 to the list associated with key 'a'.
මෙහිදී, 'a' යතුර සමඟ සම්බන්ධිත ලැයිස්තුවට 3 අගය එකතු කිරීමට append() ක්‍රමය භාවිතා කරයි.

- After this operation, the list for key 'a' changes from [1, 2] to [1, 2, 3].
  මම මෙහෙයුමෙන් පසුව, 'a' යතුර සඳහා ලැයිස්තුව [1, 2] සිට [1, 2, 3] දක්වා වෙනස් වේ.

- Now, my_dict looks like this: {'a': [1, 2, 3], 'b': [3, 4]}.
  දැන්, my_dict මේ ආකාරයට පෙනේ: {'a': [1, 2, 3], 'b': [3, 4]}.

```
my_dict['b'] = my_dict['a'] + my_dict['b']
```

This line creates a new list by concatenating the updated list for key 'a' ([1, 2, 3]) with the list for key 'b' ([3, 4]).
මෙම පේළිය 'a' යතුර ([1, 2, 3]) සඳහා යාවත්කාලීන කළ ලැයිස්තුව 'b' යතුර ([3, 4]) සඳහා ලැයිස්තුව සමඟ ඒකාබද්ධ කිරීමෙන් නව ලැයිස්තුවක් නිර්මාණය කරයි.

The + operator concatenates the two lists, resulting in [1, 2, 3] + [3, 4] which equals [1, 2, 3, 3, 4].
+ ක්‍රියාකරු ලැයිස්තු දෙක ඒකාබද්ධ කරයි, එහි ප්‍රතිඵලයක් ලෙස [1, 2, 3] + [3, 4] එය [1, 2, 3, 3, 4] සමාන වේ.

The key 'b' is then updated to this new list.
පසුව 'b' යතුර මෙම නව ලැයිස්තුවට යාවත්කාලීන ව ඇතුළත්. වේ.

```
print(my_dict)
```

Prints the entire dictionary after the updates.
යාවත්කාලීන කිරීමෙන් පසු සම්පූර්ණ dictionary එක මුද්‍රණය කරයි.

Therefore, the correct answer is 5).
එබැවින් නිවැරදි පිළිතුර 5) වේ.