සියලු ම හිමිකම් ඇවිරිණි / All Rights Reserved]

Bandaranayake - ictfromabc Ravindu Bandaranayake - ictfromabc Ravindu

Information and Communication Technology
තොරතුරු හා සන්නිවේදන තාක්ෂණය

2026 QUIZ NO 299 MARKING

2025/10/05
ictfromabc
Ravindu Bandaranayake

1. In 1944, IBM along with Howard Aiken of Harvard University created the Automatic Sequence Controlled Calculator. This made it possible to solve complex math problems without an operator. What was the name of this machine?
1944 හාවර්ඩ් විශ්වවිද්‍යාලයේ හෝවර්ඩ් අයිකන් සමග IBM සමාගම එක්ව ස්වයංක්‍රිය අනුක්‍රමන පාලිත ගණක යන්ත්‍රය නිර්මාණය කරන ලදි. මෙමගින් ක්‍රියා කරවන්නෙක් නොමැතිව සංකීර්ණ ගණිත ගැටලු විසඳිය හැකි විය. මෙම යන්ත්‍රය හැඳින්වූ නම කුමක්ද?

1) ABC 2) MARK 1 3) EDVAC 4) IBM 704 5) EDSAC

Answer: 2)

Explanation / පැහැදිලි කිරීම:

In 1944, IBM collaborated with Howard Aiken of Harvard University to create the Automatic Sequence Controlled Calculator, which was more commonly known as the Harvard Mark I or simply Mark I.
1944 දී, IBM හාවඩ් විශ්ව විද්‍යාලයේ හොවාර්ඩ් අයිකන් සමඟ සහයෝගයෙන් ස්වයංක්‍රිය අනුක්‍රමික පාලිත කැල්කියුලේටරය ASCC නිර්මාණය කරන ලදී, එය සාමාන්‍යයෙන් හාවඩ් මාර්ක් I හෝ සරලව මාර්ක් I ලෙස හැඳින්වේ.

The Harvard Mark I was one of the earliest electromechanical computers and was designed to solve complex mathematical problems automatically, without requiring constant human intervention.
Harvard Mark I යනු පැරණිතම විද්‍යුත් යාන්ත්‍රික පරිගණක වලින් එකක් වන අතර නිරන්තර මිනිස් මැදිහත්වීමක් අවශ්‍ය නොවී සංකීර්ණ ගණිතමය ගැටළු ස්වයංක්‍රියව විසඳීමට නිර්මාණය කර ඇත.

Other Options / වෙනත් විකල්ප:

1) ABC
The Atanasoff-Berry Computer, developed in the late 1930s and early 1940s, was an earlier digital computer but not related to the ASCC or Mark I.
1930 අගභාගයේ සහ 1940 මුල් භාගයේ සංවර්ධනය කරන ලද Atanasoff-Berry පරිගණකය, කලින් ඩිජිටල් පරිගණකයක් වූ නමුත් ASCC හෝ Mark I සමඟ සම්බන්ධ නොවීය.

3) EDVAC
The Electronic Discrete Variable Automatic Computer was an early binary-stored program computer, but it was developed later, in the late 1940s.
ඉලෙක්ට්‍රොනික් විවික්ත විචල්‍ය ස්වයංක්‍රිය පරිගණකය මුල් ද්විමය ගබඩා කරන ලද ක්‍රමලේඛ පරිගණකයක් වූ නමුත් එය පසුව 1940 අගභාගයේදී සංවර්ධනය කරන ලදී.

4) IBM 704
This was one of IBM's early computers, introduced in 1954, but it came much later than the Mark I.
මෙය 1954 දී හඳුන්වා දුන් IBM හි මුල් පරිගණක වලින් එකකි, නමුත් එය මාර්ක් I ට වඩා බොහෝ පසුව පැමිණියේය.

5) EDSAC
The Electronic Delay Storage Automatic Calculator was one of the first stored-program computers, developed in the late 1940s at the University of Cambridge.
ඉලෙක්ට්‍රොනික ප්‍රමාද ගබඩා ස්වයංක්‍රිය කැල්කියුලේටරය යනු 1940 අගභාගයේදී කේම්බ්‍රිජ් විශ්ව විද්‍යාලයේ සංවර්ධනය කරන ලද පළමු ගබඩා කරන ලද වැඩසටහන් පරිගණකවලින් එකකි.

So, Mark I is the correct answer.
එබැවින්, මාර්ක් I නිවැරදි පිළිතුරයි.

2. Consider the following statements.
පහත ප්‍රකාශ සලකන්න.

- Capacity limits to 1.44 MB. ධාරිතාවය 1.44 MB කට සීමා වේ.
- The most popular storage device in the early 90s. 90 දශකයේ ජනප්‍රියම ගබඩා උපාංගයයි.
- The cost is high. වියදම අධික වේ.

Which storage device matches the above statements?
ඉහත ප්‍රකාශ සඳහා ගැළපෙන ගබඩා උපාංගය වන්නේ කුමක්ද?

1) Magnetic tapes චුම්බක පටි 2) Floppy disk නම්‍ය තැටි 3) SD cards මතක කාඩ්පත් 4) Punch cards සිදුරුපත් 5) None of the above. ඉහත කිසිවක් නොවේ.

Answer: 2)

The storage device that matches the given statements is 2) Floppy disk.
ලබා දී ඇති ප්‍රකාශයන්ට ගැළපෙන ගබඩා උපාංගය වන්නේ 2) නම්‍ය තැටිය වේ.

Capacity limits to 1.44 MB. / ධාරිතාවය 1.44 MB කට සීමා වේ.
The standard capacity for a 3.5-inch floppy disk is 1.44 MB.
අඟල් 3.5 නම්‍ය තැටියක් සඳහා සම්මත ධාරිතාව 1.44 MB වේ.

The most popular storage device in the early 90s. / 90 දශකයේ ජනප්‍රියම ගබඩා උපාංගයයි.
Floppy disks were widely used during the early 90s for data storage and transfer.
90 දශකයේ මුල් භාගයේදී දත්ත ගබඩා කිරීම සහ මාරු කිරීම සඳහා නම්‍ය තැටි බහුලව භාවිතා විය.

The cost is high. / වියදම අධික වේ.
Compared to the storage capacity, floppy disks were relatively expensive at the time.
ගබඩා ධාරිතාව හා සසඳන විට, නම්‍ය තැටි එකල සාපේක්ෂව මිල අධික විය.

So, the answer is 2).
එබැවින් පිළිතුර 2) වේ.

3. Which of the following is not a valid Binary Coded Decimal (BCD) value?
පහත සඳහන් පිළිතුරු වලින් වලංගු BCD අගයක් නොවන්නේ කුමක්ද?

1) 10001000_{BCD} 2) 10110001_{BCD} 3) 10010111_{BCD} 4) 00010101_{BCD} 5) 00110011_{BCD}

Answer: 2)

10001000_{BCD}:
This is a valid BCD value. Each digit (8 and 8) is represented by its respective 4-bit binary code.
මෙය වලංගු BCD අගයකි. සෑම ඉලක්කමක්ම (8 සහ 8) එහි අදාළ 4-bit ද්විමය කේතය මගින් නිරූපණය කෙරේ.

10110001_{BCD}:
Here, 1011 equals 11 in decimal. 11 is not allowed in BCD (since valid range is 0–9). So, this is not a valid BCD value.
මෙහිදී, 1011 දශමයෙන් 11 ට සමාන වේ. BCD හි 11 ට අවසර නැත (වලංගු පරාසය 0–9 බැවින්). එබැවින්, මෙය වලංගු BCD අගයක් නොවේ.

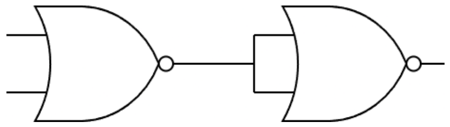
10010111_{BCD}:
This is a valid BCD value. Each digit (9 and 7) is represented by its respective 4-bit binary code.
මෙය වලංගු BCD අගයකි. සෑම ඉලක්කමක්ම (9 සහ 7) එහි අදාළ 4-bit ද්විමය කේතය මගින් නිරූපණය කෙරේ.

00010101_{BCD}:
This is a valid BCD value. Each digit (1 and 5) is represented by its respective 4-bit binary code.
මෙය වලංගු BCD අගයකි. සෑම ඉලක්කමක්ම (1 සහ 5) එහි අදාළ 4-bit ද්විමය කේතය මගින් නිරූපණය කෙරේ.

00110011$_{BCD}$:
This is a valid BCD value. Each digit (3 and 3) is represented by its respective 4-bit binary code.
මෙය වලංගු BCD අගයකි. සෑම ඉලක්කමක්ම (3 සහ 3) එහි අදාළ 4-bit ද්විමය කේතය මගින් නිරූපණය කෙරේ.

So, the answer is 2).
එබැවින් පිළිතුර 2) වේ.

4. What is the logic gate equivalent to the given logic circuit which made using 2 NOR gates?
NOR ද්වාර 2ක් භාවිතයෙන් සාදන ලද ලබා දී ඇති තාර්කික පරිපථයට සමාන තාර්කික ද්වාරය කුමක්ද?



1) XOR 2) AND 3) NOT <u>4) OR</u> 5) XNOR

Answer: 4)

Let's consider the 2 inputs A and B are given to the above logic circuit and F is the output.
ඉහත තාර්කික පරිපථයට A සහ B ලෙස ආදාන දෙකක් සහ F යනුවෙන් ප්‍රතිදානයක් ඇතැයි සලකන්න.

When A and B inputs goes through the 1st NOR gate, we can obtain the expression as $\overline{A+B}$.
A සහ B ආදාන 1 වන NOR ද්වාරය හරහා ගමන් කළ විට, $\overline{A+B}$ යන ප්‍රකාශනය ලබා ගත හැක.

When two $\overline{A+B}$ expressions given to the 2nd NOR gate as 2 inputs, we can obtain the expression to F as $\overline{(\overline{A+B})(\overline{A+B})}$. We can simplify the obtain expression as below.
2 වන NOR ද්වාරයට ආදාන 2ක් ලෙස $\overline{A+B}$ ප්‍රකාශන දෙකක් ලබා දුන් විට, F සඳහා $\overline{(\overline{A+B})(\overline{A+B})}$ ප්‍රකාශය ලබා ගත හැකි වේ. ලබා ගත් ප්‍රකාශනය අපට පහත පරිදි සුළු කරගත හැකිය.

$\overline{(\overline{A+B})(\overline{A+B})}$
$\overline{(\overline{A+B})} + \overline{(\overline{A+B})}$ De Morgan's Law / ද මුවාර් ප්‍රමේයය
$(A+B) + (A+B)$ Double complement Law / ද්විත්ව ප්‍රතිලෝම න්‍යාය
$(A+B)$ Idempotent Law / තදේවභාවි න්‍යාය

Therefore, the expression obtained by solving the above circuit is equivalent to the expression that obtains by the OR gate. Thus, above circuit is equivalent to OR gate.
එබැවින් ඉහත පරිපථය විසඳීමෙන් ලැබෙන ප්‍රකාශනය OR ද්වාරය මගින් ලබා ගන්නා ප්‍රකාශනයට සමාන වේ. මේ අනුව, ඉහත පරිපථය OR ද්වාරයට සමාන වේ.

5. What is the Sum Of Product (SOP) expression that can be obtained by the below Karnaugh map?
පහත කානෝ සිතියම මගින් ලබා ගත හැකි ගුණිතවල එකතු (SOP) ප්‍රකාශනය කුමක්ද?

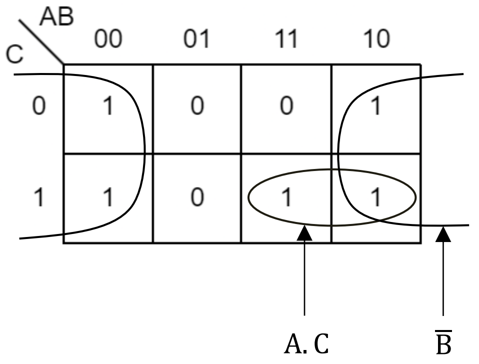
C \ AB	00	01	11	10
0	1	0	0	1
1	1	0	1	1

1) <u>$\overline{B} + CA$</u>
2) $\overline{C} + BA$
3) $C + \overline{B}A$
4) $B + C\overline{A}$
5) $C + \overline{BA}$

Answer: 1)

In order to find the sum of product expression (SOP), first we have to consider and group the points where 1 exists in the Karnaugh map.
ගුණිතයන්ගේ එකතු ප්‍රකාශනය (SOP) සෙවීම සඳහා පළමුව කානෝ සිතියමේ 1 පවතින ස්ථාන සලකා කාණ්ඩගත කළ යුතුය.

Then write expressions related to those categories.
ඉන්පසුව එම කාණ්ඩ වලට අදාල ප්‍රකාශන ලිවිය යුතුය.



Then the SOP expression can be written as follows.
එවිට SOP ප්‍රකාශනය පහත ආකාරයට ලිවිය හැකිය.

$SOP = A.C + \overline{B}$

6. What is the output of the following python code?
පහත පයිතන් කේතයේ ප්‍රතිදානය කුමක්ද?

```
x = 0
a = 5
b = 5
if a > 0:
    if b < 0:
        x = x + 5
    elif a > 5:
        x = x + 4
    else:
        x = x + 3
else:
    x = x + 2
print(x)
```

1) 0 2) 2 <u>3) 3</u> 4) 4 5) 5

Answer: 3)

Let's consider the code. කේතය සලකා බලමු.

- Value Assignment / අගයන් ආදේශ කිරිම

  x = 0
  0 is assigned to x.
  x වෙත 0 පවරා ඇත.

  a = 5
  5 is assigned to a
  a වෙත 5 පවරා ඇත.

  b = 5
  5 is assigned to b.
  b වෙත 5 පවරා ඇත.

- The outer if condition checks
  පිටත if කොන්දේසිය පරීක්ෂා කරයි.

`if a > 0:`
Since a = 5, which is greater than 0, this condition is true.
0 ට වඩා 5 වැඩි බැවින්, මෙම කොන්දේසිය සත්‍ය වේ.

Let's move inside the outer if block.
එවිට පිටත if block කොටස තුළට ගමන් කරයි.

- Inside the first if block, the code checks the inner conditions
  පළමු if block කොටස තුළ, කේතය අභ්‍යන්තර කොන්දේසින් පරීක්ෂා කරයි.

`if b < 0:`
Since b = 5, which is not less than 0, this condition is false, so we skip this block.
b = 5, 0 වඩා නොඅඩු නිසා, මෙම කොන්දේසිය අසත්‍ය වේ, එබැවින් මෙම කොටස මග හරියි.

- The elif condition is then checked.
  පසුව elif කොන්දේසිය පරීක්ෂා කරනු ලැබේ.

`elif a > 5:`
Since a = 5, which is not greater than 5, this condition is also false, so this block is skipped as well.
a = 5, 5 ට වඩා වැඩි නොවන elif නිසා, මෙම කොන්දේසිය ද අසත්‍ය බැවින්, මෙම කොටසද මඟ හරියි.

- Finally, the else block is executed because neither the if nor the elif conditions were true.
  අවසාන වශයෙන්, if හෝ elif කොන්දේසි සත්‍ය නොවූ නිසා else block කොටසද ක්‍රියාත්මක වේ.

```
else:
    x = x + 3
```
x is updated to x + 3, so x = 0 + 3 = 3.
x + 3 ලෙස යාවත්කාලීන වේ, එබැවින් x = 0 + 3 = 3.

- This prints the value of x, which is now 3.
  මෙය x හි අගය මුද්‍රණය කරයි, එය දැන් 3 වේ.

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

```
def func(a, b=[]):
    b.append(a)
    return b
print(func(1))
print(func(2, []))
print(func(3))
```

1) `[1] [2] [3]` 2) `[1, 3] [2] [3]` 3) <u>`[1] [2] [1, 3]`</u>
4) `[1] [2, 3] [1, 3]` 5) `Error`

Answer: 3)

`func(1)`
b defaults to an empty list [ ]. 1 is appended to b. The function returns `[1]`**.** So the Output is **`[1]`**
හිස් ලැයිස්තුවකට`[]`, b පෙරනිමි වේ.1 bට අනුයුක්ත කර ඇත. ශ්‍රිතය `[1]` වෙත නැවත පැමිණේ. එබැවින් ප්‍රතිදානය `[1]` වේ

`func(2, [])`
b is explicitly provided as an empty list [ ]. 2 is appended to this new list. The function returns **`[2]`**.So the Output is `[2]`
b හිස් ලැයිස්තුවක් ලෙස පැහැදිලිව සපයා ඇත `[]`.මෙම නව ලැයිස්තුවට 2 එකතු කර ඇත. ශ්‍රිතය `[2]` වෙත නැවත පැමිණේ. එබැවින් ප්‍රතිදානය `[2]` වේ.

`func(3)`

b defaults to the list that was modified in the first call, which is `[1]` (default mutable argument persists across calls). 3 is appended to this list. So, the function returns `[1, 3]` So the Output is **`[1, 3]`**
පළමු ඇමතුමේදී වෙනස් කරන ලද ලැයිස්තුවට b පෙරනිමි වේ, එනම් `[1]` (පෙරනිමි mutable තර්කය ඇමතුම් හරහා පවතී). මෙම ලැයිස්තුවට 3 එක් කර ඇත. එබැවින් ශ්‍රිතය `[1, 3]` දක්වා නැවත පැමිණේ, එබැවින් ප්‍රතිදානය `[1, 3]` වේ.

Therefore the correct answer is 3) `[1] [2] [1, 3]`
එබැවින් නිවැරදි පිළිතුර 3) `[1] [2] [1, 3]` වේ.

8. What is the output of the following Python code?
පහත පයිතන් කේතයෙහි ප්‍රතිදානය කුමක්ද?

```
a=(4*3//5--2)
b=a
print(b)
```

1) `4.0`   <u>2) `4`</u>   3) `-2`   4) `-4.0`   5) `Syntax error`

Answer: 2)

Explanation / පැහැදිලි කිරීම:

Let's break down the expression in the code step by step,
කේතයේ ප්‍රකාශනය පියවරෙන් පියවර බිඳ දමමු,

- **Multiplication / ගුණ කිරීම (4 * 3)**

  This evaluates to 12.
  මෙය 12 දක්වා තක්සේරු කරයි.

- **Integer division / නිඛිල බෙදීම (12 // 5)**

  This evaluates to 2 because 12 // 5 gives the quotient when dividing 12 by 5 (without the remainder).
  12 5 න් බෙදීමේදී (ඉතුරු කොටස නොමැතිව) 12 // 5 ප්‍රමාණය ලබා දෙන බැවින් මෙය 2 ලෙස ඇගයීමට ලක් කරයි.

- **Handling - -2**

  The -- in Python is interpreted as a double negative. --2 is equivalent to +2.
  පයිතන් හි -- ද්විත්ව සෘණ ලෙස අර්ථ දැක්වේ. --2 යනු 2 ට සමාන වේ.

  So, the expression becomes 2 + 2.
  එබැවින්, ප්‍රකාශනය 2 + 2 බවට පත් වේ.

- **Final result / අවසාන ප්‍රතිඵලය**

  2 + 2 evaluates to 4.
  2 + 2 අගය 4 වෙත.

  The value of `a` is 4, and since b = a, b is also 4.
  a හි අගය 4 වන අතර, b = a නිසා b ද 4 වේ.

  When print(b) is executed, the output is 4.
  print(b) ක්‍රියාත්මක කළ විට, ප්‍රතිදානය 4 වේ.

9. Which of the following data type is immutable in Python?
පහත සඳහන් කුමන දත්ත වර්ගය පයිතන් හිදී වෙනස් කළ නොහැකිද?

1) List 2) Dictionary 3) Tuple 4) Set 5) All of the above

Answer: 3)

1) **List**
A list is a mutable data type, meaning that its contents can be changed after it is created. You can add, remove, or modify elements in a list.
list යනු වෙනස් කළ හැකි දත්ත වර්ගයකි, එයින් අදහස් වන්නේ එය නිර්මාණය කිරීමෙන් පසුව එහි අන්තර්ගතය වෙනස් කළ හැකි බවයි. ඔබට ලැයිස්තුවක ඇති මූලද්‍රව්‍ය එක් කිරීමට, ඉවත් කිරීමට හෝ වෙනස් කිරීමට හැකිය.
ex:
```
my_list = [1, 2, 3]
my_list.append(4)  # List is now [1, 2, 3, 4]
```

2) **Dictionary**
A dictionary is also mutable. You can add, remove, or change key-value pairs.
dictionary mutable වේ. ඔබට key-value යුගල එකතු කිරීමට, ඉවත් කිරීමට හෝ වෙනස් කිරීමට හැකිය.
ex:
```
my_dict = {'a': 1, 'b': 2}
my_dict['c'] = 3  # Dictionary is now {'a': 1, 'b': 2, 'c': 3}
```

3) **Tuple**
A tuple is immutable, meaning once it is created, its contents cannot be changed. You cannot add, remove, or modify elements in a tuple.
Tuple එකක් වෙනස් (immutable) කළ නොහැකි ය, එනම් එය නිර්මාණය කළ පසු එහි අන්තර්ගතය වෙනස් කළ නොහැක. ඔබට Tuple එකක elements එකතු කිරීමට, ඉවත් කිරීමට හෝ වෙනස් කිරීමට නොහැක.
Ex:
```
my_tuple = (1, 2, 3)
# Trying to modify my_tuple[0] = 4 would result in an error
```

4) **Set**
A set is mutable. You can add and remove elements from a set.
Set වෙනස් කළ හැකි (mutable) වේ. ඔබට Set elements එකතු කිරීමට සහ ඉවත් කිරීමට හැකිය.
eg:
```
my_set = {1, 2, 3}
my_set.add(4)  # Set is now {1, 2, 3, 4}
```

5) **All of the above**
This option is incorrect because not all the listed data types are immutable.
ලැයිස්තුගත කර ඇති සියලුම දත්ත වර්ග වෙනස් කළ නොහැකි බැවින් මෙම විකල්පය වැරදිය.

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

10. Which of the following Python variable names are valid?
පහත දක්වා ඇති පයිතන් විචල්‍ය නාම අතරින් වලංගු වන්නේ කුමක්ද?

1) `I-C-T` 2) `I C T` 3) `I_C_T_` 4) `I+C+T` 5) `I#C#T`

Answer: 3)

Explanation / පැහැදිලි කිරීම:

In Python, variable names must follow certain rules.
Python හි, විචල්‍ය නම් යම් නීති රීති අනුගමනය කළ යුතුය.

Variable names can contain letters (a-z, A-Z), digits (0-9), and underscores (_).
විචල්‍ය නම්වල අකුරු (a-z, A-Z) ඉලක්කම් (0-9) සහ යටි ඉරි (_) අඩංගු විය හැක.

Variable names cannot start with a digit.
විචල්‍ය නම් ඉලක්කමකින් ආරම්භ කළ නොහැක.

Spaces, special characters like hyphens (-), plus signs (+), and hashtags (#) are not allowed.
හිස්වැසුම්, hyphens (-), plus signs (+) සහ hashtags (#) වැනි විශේෂ අක්ෂරවලට ඉඩ නොදේ.

Let's analyze each option,
එක් එක් විකල්පය විශ්ලේෂණය කරමු,

1) **I-C-T**
   Invalid. The hyphen (-) is not allowed in Python variable names.
   වලංගු නැත. Python විචල්‍ය නම් වල hyphen (-) ඉඩ නොදේ.

2) **I C T**
   Invalid. Spaces are not allowed in Python variable names.
   වලංගු නැත. Python විචල්‍ය නම් වල මැදින් ඉඩ නොදේ.

3) **_I_C_T_**
   Valid. It contains only letters and underscores, which are allowed.
   වලංගුයි. එහි අඩංගු වන්නේ අවසර දී ඇති අකුරු සහ යටි ඉරි පමණි.

4) **I+C+T**
   Invalid. The plus sign (+) is not allowed in Python variable names.
   වලංගු නැත. Python විචල්‍ය නාමවල ප්ලස් ලකුණ (+) ඉඩ නොදේ.

5) **I#C#T**
   Invalid. The hashtag (#) is not allowed in Python variable names.
   වලංගු නැත. පයිතන් විචල්‍ය නාමවල හැෂ් ටැගය (#) ඉඩ නොදේ.

Thus, option 3 (_I_C_T_) is the only valid Python variable name.
මේ අනුව, විකල්ප 3 (_I_C_T_) එකම වලංගු Python විචල්‍ය නාමය වේ.