

සියලු ම හිමිකම් ඇවිරිණි /All Rights Reserved]

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

Information and Communication Technology

තොරතුරු හා සන්නිවේදන තාක්ෂණය

2026 QUIZ NO 456 MARKING

2026/03/11
ictfromabc
Ravindu Bandaranayake

1. Which of the following processors is a single-core processor?

තනි හරයක් සහිත සකසනයක් වන්නේ පහත සකසන වලින් කුමන සකසනය ද?

- 1) Intel core 2 Duo E6750
- 2) Intel core i9-14900KS
- 3) AMD Athlon X2 6400+
- 4) Intel 4004
- 5) Apple M3

Answer : 4)

Intel Core 2 Duo E6750

This is a dual-core processor. Part of Intel's Core 2 Duo series. Primarily used in everyday computing tasks, office applications, and basic gaming.

මෙම සකසනය ද්විත්ව හර වේ. එය Intel හි Core 2 Duo මාලාවට අයත් වේ. ප්‍රධාන වශයෙන් දිනපතා භාවිතා කරන පරිගණක, කාර්යාල පරිගණක සහ මූලික ක්‍රීඩා කටයුතු සඳහා භාවිතා වේ.

Intel Core i9-14900KS

This is a multi-core processor (typically many cores, ranging from 8 to 24 or more). A high-end processor in Intel's Core i9 series. Suitable for gaming, video editing, 3D rendering, and professional data processing.

මෙම සකසනය බහු හර වේ. මෙය Intel හි Core i9 මාලාවට අයත් ප්‍රධාන වශයෙන් වේගවත් බහු හර සකසනයකි. විශේෂයෙන්ම ක්‍රීඩා, විඩියෝ සංස්කරණ, 3D රෙන්ඩරින් සහ වෘත්තීයමය දත්ත සැකසුම් සඳහා භාවිතා වේ.

AMD Athlon X2 6400+

This is a dual-core processor. A part of AMD's Athlon 64 X2 series. Used for everyday computing and basic office tasks.

මෙම සකසනය ද්විත්ව හර වේ. AMD හි Athlon 64 X2 මාලාවේ ද්විත්ව හර සකසනයයි. දිනපතා භාවිතාවන් සහ මූලික කාර්යාල කටයුතු සඳහා භාවිතා වේ.

Intel 4004

This is a single-core processor. Originally released in 1971 as the first commercially available microprocessor. They were used in calculators and other simple electronic devices, marking a significant milestone in computing history.

මෙම සකසනය තනි හර වේ. මෙය Intel විසින් 1971 දී නිකුත් කරන ලද වාණිජමය වශයෙන් ලබා ගත හැකි පළමු ක්ෂුද්‍ර සකසනයයි. 1970 දශකයේ කැල්කියුලේටර් සඳහා භාවිත වූ අතර සාර්ව වශයෙන් ඉතිහාසයේ වැදගත්ම සන්ධිස්ථානයක් ලෙස සැලකිය හැකිය.

Apple M3

This is a multi-core processor, the latest in Apple's M-series processors. Found in the latest MacBooks, iMacs, and iPads, designed for a wide range of tasks from general computing to intensive creative work.

Apple M-series හි නවතම බහු හර සකසනයකි. නවතම MacBook, iMac සහ iPad ආදී උපාංගවල පරිපාලන කාර්ය, නිර්මාණාත්මක වැඩ සහ වේගවත් කාර්ය සාධනය සඳහා භාවිතා වේ.

Therefore, Intel 4004 is the correct answer.

එබැවින් Intel 4004 නිවැරදි පිළිතුරයි.

2. $10101011 \text{ AND } 11011100 = A$, $A \text{ OR } 11100111$ What is the result of the arithmetic operation?
 $10101011 \text{ AND } 11011100 = A$, $A \text{ OR } 11100111$ අංක ගණිත මෙහෙයුමෙහි ප්‍රතිඵලය කුමක් ද ?

- 1) 10001101 2) 11100011 3) 11101111 4) 10111100 5) 11001100

Answer : 3)

$10101011_2 \text{ AND } 11011100_2$

$$\begin{array}{r} 1 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 1 \\ 1 \ 1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \\ \hline 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \end{array}$$

$A \text{ OR } 11100111_2$ ($A = 10001000_2$)

$$\begin{array}{r} 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \\ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \\ \hline 1 \ 1 \ 1 \ 0 \ 1 \ 1 \ 1 \ 1 \end{array}$$

This answer equates to A.

මෙම ලැබුණු පිළිතුර A ට සමාන කරයි.

The correct answer is, 11101111_2

එමනිසා නිවැරදි පිළිතුර වන්නේ, 11101111_2

3. Among these, the number of bits possible for Sign, Exponent, and Mantissa in IEEE 754 - Double Precision is,

මේ අතරින් IEEE 754 - Double Precision හි Sign, Exponent සහ Mantissa සඳහා තිබිය හැකි බිටු ගණන වන්නේ,

- 1) 1, 11, 32 2) 1, 7, 24 3) 1, 8, 23 4) 1, 11, 52 5) 1, 12, 33

Answer : 4)

The correct answer is **4) 1, 11, 52**. Why each option can or cannot be correct for the number of bits allocated to the Sign, Exponent, and Mantissa in IEEE 754 double precision format,

නිවැරදි පිළිතුර **4) 1, 11, 52**. IEEE 754 ද්විත්ව නිරවද්‍යතා ආකෘතියෙන් Sign, Exponent සහ Mantissa සඳහා වෙන් කර ඇති බිටු ගණන සඳහා එක් එක් විකල්පය නිවැරදි විය හැක්කේ හෝ නොවිය හැක්කේ,

- IEEE 754 Double Precision Format

IEEE 754 ද්විත්ව නිරවද්‍යතා ආකෘතිය

IEEE 754 double precision format, also known as binary64, represents floating-point numbers using three components: a sign bit, an exponent, and a mantissa (also known as the significand). The total number of bits used is 64

IEEE 754 ද්විත්ව නිරවද්‍යතා ආකෘතිය, binary64 ලෙසද හැඳින්වේ, සංරචක තුනක් භාවිතා කරමින් floating-point සංඛ්‍යා නියෝජනය කරයි: sign bit, exponent සහ a mantissa (significand) ලෙසද හැඳින්වේ. භාවිතා කරන ලද මුළු බිටු ගණන 64 කි.

1) 1, 11, 32

In this option, the sign bit is correctly assigned 1 bit. However, the exponent and mantissa bits do not add up correctly to 64 bits. The exponent is given 11 bits (which is correct), but the mantissa is given only 32 bits, resulting in a total of 44 bits when combined with the sign bit, which is incorrect.

මෙම විකල්පය තුළ, සලකුණු බිටු නිවැරදිව 1 බිටු ලබා දී ඇත. කෙසේ වෙතත්, exponent සහ mantissa බිටු බිටු 64 දක්වා නිවැරදිව එකතු නොවේ. දර්ශකය බිටු 11 ක් ලබා දී ඇත (එය නිවැරදියි), නමුත් mantissa ලබා දී ඇත්තේ බිටු 32ක් පමණි, එහි ප්‍රතිඵලය ලෙස ලකුණු බිටු සමඟ එකතු වූ විට සම්පූර්ණ බිටු 44ක් ලැබේ, එය වැරදිය.

2) 1, 7, 24

In this option, the sign bit is correctly assigned 1 bit. However, the exponent is given 7 bits, and the mantissa 24 bits, totaling only 32 bits, which is the format for single precision, not double precision.

මෙම විකල්පය තුළ, සලකුණු බිටු නිවැරදිව 1 බිටු ලබා දී ඇත. කෙසේ වෙතත්, exponent ට බිටු 7ක් ලබා දී ඇති අතර, mantissa ට බිටු 24ක් ලබා දී ඇති අතර, එය බිටු 32ක් පමණි, එය ද්විත්ව නිරවද්‍යතාවයක් නොව තනි නිරවද්‍යතාව සඳහා වන ආකෘතියයි.

3) 1, 8, 23

This option correctly assigns 1 bit for the sign. However, 8 bits for the exponent and 23 bits for the mantissa represent the IEEE 754 single precision format, not double precision, totaling 32 bits.

මෙම විකල්පය ලකුණු සඳහා බිටු 1ක් නිවැරදිව පවරයි. කෙසේ වෙතත්, exponent සඳහා බිටු 8ක් සහ mantissa සඳහා බිටු 23ක් නියෝජනය කරන්නේ IEEE 754 තනි නිරවද්‍යතා ආකෘතිය මිස ද්විත්ව නිරවද්‍යතාවයක් නොව, මුළු බිටු 32කි.

4) 1, 11, 52

This option correctly assigns 1 bit for the sign, 11 bits for the exponent, and 52 bits for the mantissa. This adds up to 64 bits in total, which is the correct format for IEEE 754 double precision representation. The 11 bits for the exponent allow for a range of values, and the 52 bits for the mantissa provide the precision needed for double precision numbers.

මෙම විකල්පය නිවැරදිව ලකුණු සඳහා බිටු 1 ක්, exponent සඳහා බිටු 11 ක් සහ mantissa සඳහා බිටු 52 ක් නියම කරයි. මෙය IEEE 754 ද්විත්ව නිරවද්‍යතා නිරූපණය සඳහා නිවැරදි ආකෘතිය වන මුළු බිටු 64ක් දක්වා එකතු කරයි. exponent සඳහා වන බිටු 11 අගයන් පරාසයකට ඉඩ සලසයි, සහ mantissa සඳහා බිටු 52 ද්විත්ව නිරවද්‍යතා සංඛ්‍යා සඳහා අවශ්‍ය නිරවද්‍යතාවය සපයයි.

5) 1, 12, 33

In this option, the sign bit is correctly assigned 1 bit. However, 12 bits for the exponent and 33 bits for the mantissa add up to 46 bits, which is not the correct total for double precision (64 bits).

මෙම විකල්පය තුළ, සලකුණු බිට් නිවැරදිව 1 බිට් ලබා දී ඇත. කෙසේ වෙතත්, exponent සඳහා බිටු 12 ක් සහ mantissa සඳහා බිටු 33 ක් බිටු 46 ක් දක්වා එකතු වේ, එය ද්විත්ව නිරවද්‍යතාවය බිටු 64 සඳහා නිවැරදි එකතුව නොවේ.

So, the correct configuration for the IEEE 754 double precision format is 1 bit for the sign, 11 bits for the exponent, and 52 bits for the mantissa, which makes option 4 the correct answer.

එබැවින්, IEEE 754 ද්විත්ව නිරවද්‍යතා ආකෘතිය සඳහා නිවැරදි වින්‍යාසය ලකුණු සඳහා බිට් 1 ක්, exponent සඳහා බිටු 11 ක් සහ mantissa සඳහා බිටු 52 ක් වන අතර එමඟින් විකල්පය 4 නිවැරදි පිළිතුර වේ.

4. Which of the following is false about NTFS compared to the FAT file system?

FAT ගොනු පද්ධතිය හා සසඳන විට NTFS සම්බන්ධයෙන් පහත සඳහන් දේවලින් අසත්‍ය වන්නේ කුමක්ද?

- 1) NTFS supports automatic recovery from some disk-related errors
NTFS සමහර තැටි සම්බන්ධ දෝෂ වලින් ස්වයංක්‍රීයව ප්‍රකෘති තත්වයට පත් කිරීම සඳහා සහය දක්වයි
- 2) NTFS supports the Unicode encoding system
NTFS යුනිකේතන පද්ධතියට සහය දක්වයි
- 3) NTFS improves support for larger hard disks
NTFS විශාල දෘඪ තැටි සඳහා සහය වැඩි දියුණු කරයි
- 4) NTFS stores the root directory at a fixed location
NTFS මූල නාමාවලිය ස්ථාවර ස්ථානයක ගබඩා කරයි
- 5) NTFS provides better security through permissions and encryption
NTFS අනුමැතියක් සහ ගුප්ත කේතනය හරහා වඩා හොඳ ආරක්ෂාවක් සපයයි

Answer: 4)

Explanation of each option / එක් එක් විකල්පය පැහැදිලි කිරීම:

Option / විකල්පය 1)

True. NTFS uses a journaling file system that can automatically recover from some disk errors without user intervention.

නිවැරදියි. NTFS මගින් පරිශීලක මැදිහත්වීමකින් තොරව සමහර තැටි සම්බන්ධ දෝෂ ස්වයංක්‍රීයව යථා තත්ත්වයට පත් කළ හැකි journaling ගොනු පද්ධතියක් භාවිතා කරයි.

Option / විකල්පය 2)

True. NTFS supports Unicode, allowing the use of international characters in file and folder names.

නිවැරදියි. NTFS යුනිකේතනය සඳහා සහය දක්වයි, ගොනු සහ ගොනු බහාලුම් නාමවලි ජාත්‍යන්තර අක්ෂර භාවිතා කිරීමට ඉඩ සලසයි.

Option / විකල්පය 3)

True. NTFS is designed to handle large volumes and file sizes far beyond the limits of FAT (especially FAT32).

නිවැරදියි. NTFS නිර්මාණය කර ඇත්තේ FAT ගොනු පද්ධතිවල (විශේෂයෙන් FAT32) සීමාවන් ඉක්මවා විශාල පරිමාවන් සහ ගොනු ප්‍රමාණයන් හැසිරවීමටයි.

Option / විකල්පය 4)

False. This is a feature of FAT file systems, not NTFS. In FAT, the root directory is stored at a fixed location on the disk. In NTFS, the root directory is stored as a regular metadata file, and its location is not fixed. This provides flexibility and robustness.

වැරදියි. මෙය NTFS නොව FAT ගොනු පද්ධතිවල ලක්ෂණයකි. FAT ගොනු පද්ධතියෙහි, මූල නාමාවලිය තැටියේ ස්ථාවර ස්ථානයක ගබඩා කර ඇත. NTFS හි, මූල නාමාවලිය සාමාන්‍ය පාර-දත්ත ගොනුවක් ලෙස ගබඩා කර ඇති අතර එහි පිහිටීම ස්ථාවර නොවේ. මෙය නම්‍යශීලී බව සහ ශක්තිමත් බව සපයයි.

Option / විකල්පය 5)

True. NTFS supports file and folder-level permissions (via ACLs) and also supports features like EFS (Encrypting File System), offering much better security than FAT.

නිවැරදියි. NTFS, ගොනු සහ ගොනු බහාලුම් මට්ටමේ අනුමැතියන්(ACL හරහා) සඳහා සහය දක්වන අතර FAT ගොනු පද්ධතියට වඩා බෙහෙවින් හොඳ ආරක්ෂාවක් ලබා දෙන EFS(Encrypting File System) වැනි විශේෂාංග සඳහාද සහය දක්වයි.

Therefore, the answer is 4).

එබැවින්, පිළිතුර 4) වේ.

5. Student(RegNo, Name, Address, Gender)
Section(S_Id, Name, RegNo)

What is the correct SQL code to insert the foreign key RegNo into the Section table above?

ඉහත Section වගුවට RegNo නම් ආගන්තුක යතුර ඇතුළත් කිරීමට අදාළ නිවැරදි SQL කේතය කුමක්ද?

- | | |
|---|---|
| 1) INSERT TABLE Section
ADD FOREIGN KEY RegNo; | 2) ALTER TABLE Section
ADD FOREIGN KEY (RegNo)
REFERENCES Student(RegNo); |
| 3) ALTER TABLE Section
ADD FOREIGN KEY RegNo; | 4) ALTER TABLE Student
ADD FOREIGN KEY (S_Id); |
| 5) INSERT INTO Section
ADD PRIMARY KEY (RegNo)
REFERENCES Student(RegNo); | |

Answer: 2)

The correct SQL code to insert the foreign key RegNo into the Section table is,

RegNo යන ආගන්තුක යතුර Section වගුවට ඇතුළු කිරීමට නිවැරදි SQL කේතය වන්නේ,

```
ALTER TABLE Section
ADD FOREIGN KEY (RegNo)
REFERENCES Student(RegNo);
```

Explanation / පැහැදිලි කිරීම:

This SQL command does the following

මෙම SQL විධානය පහත සඳහන් දේ කරයි

ALTER TABLE Section

Modifies the Section table. / Section වගුව වෙනස් කරයි.

ADD FOREIGN KEY (RegNo)

Adds a foreign key constraint to the RegNo column.

RegNo තීරුවට ආගන්තුක යතුරු සීමාවක් එක් කරයි.

REFERENCES Student(RegNo)

Links this column to the RegNo column in the Student table (which is assumed to be a primary key).

මෙම තීරුව ශිෂ්‍ය වගුවේ RegNo තීරුවට සම්බන්ධ කරයි (එය ප්‍රාථමික යතුරක් ලෙස උපකල්පනය කෙරේ).

This enforces referential integrity, ensuring that any RegNo value in the Section table must already exist in the Student table.

මෙය යොමු අඛණ්ඩතාව බලාත්මක කරයි, Section වගුවේ ඇති ඕනෑම RegNo අගයක් දැනටමත් ශිෂ්‍ය වගුවේ තිබිය යුතු බව සහතික කරයි.

Why the other options are incorrect / අනෙකුත් විකල්ප වැරදි වන්නේ ඇයි?

1) INSERT TABLE Section ADD FOREIGN KEY RegNo;

Invalid SQL syntax. INSERT is for inserting data, not modifying table structure.

අවලංගු SQL වාක්‍ය ඛණ්ඩය. INSERT යනු දත්ත ඇතුළු කිරීම සඳහා වන අතර වගු ව්‍යුහය වෙනස් කිරීම නොවේ.

3) ALTER TABLE Section ADD FOREIGN KEY RegNo;

Syntax is incomplete, it does not specify the referenced table and column.

වාක්‍ය ඛණ්ඩය අසම්පූර්ණයි, එය යොමු කළ වගුව සහ තීරුව සඳහන් නොකරයි.

4) ALTER TABLE Student ADD FOREIGN KEY (S_Id) REFERENCES Student(RegNo);

Wrong table (Student instead of Section) and wrong logic (foreign key from S_Id to RegNo doesn't make sense).

වැරදි වගුව (Section වෙනුවට Student) සහ වැරදි තර්කනය (S_Id සිට RegNo දක්වා ආගන්තුක යතුර තේරුමක් නැත).

5) INSERT INTO Section ADD PRIMARY KEY (RegNo) REFERENCES Student(RegNo);

INSERT INTO is used to add data, not constraints.

INSERT INTO භාවිතා කරන්නේ දත්ත එකතු කිරීමට මිස සීමා කිරීම් නොවේ.

A primary key cannot reference another table, only foreign keys do that.

ප්‍රාථමික යතුරකට වෙනත් වගුවක් යොමු කළ නොහැක, එය කරන්නේ ආගන්තුක යතුරු පමණි.

6. Consider the following Python Code

පහත පයිතන් කේතය සලකා බලන්න.

```
x=20
L=[]
if x<=0:
    print("")
else:
    for i in range(1,x):
        y=i*i
        if y>=x:
            break
        L.append(y)
    print(L)
```

Instead of displaying a list in the above code, we want to display the * markers as a square according to the size of each item. What Python code should be used instead of print(L) for this?

ඉහත කේතයේ ලැයිස්තුවක් ප්‍රදර්ශනය කිරීම වෙනුවට එම එක් එක් අයිතමයේ ප්‍රමාණය පරිදි * සලකුණු සමවතුරුසුයක් ලෙස ප්‍රදර්ශනය කිරීමට අවශ්‍ය වේ. මේ සඳහා print(L) වෙනුවට යෙදිය යුතු පයිතන් කේත කොටස කුමක්ද?

1) print(L* '*')

2) for n in L:
 print(n* '*')

3) for n in range(L):
 print(n* '*')
 print('next')

4) for n in L:
 L=int(n**0.5)
 for d in range(L):
 print(L* '*')
 print("next")

5) None of the above / ඉහත කිසිවක් නොවේ.

Answer: 4)

Explanation / පැහැදිලි කිරීම:

Let's study the given code line by line.

ලබා දී ඇති කේතයේ පේළියෙන් පේළිය අධ්‍යයනය කරමු.

x=int(input())

A positive integer is taken as input for x.

x සඳහා ධන නිඛිලයක් ආදානය ලෙස ලබාගනී.

L=[]

if x<=0:

print("")

If the value of x is 0 or less, an empty string is output.

x හි අගය 0 හෝ ඊට වඩා කුඩානම් නිස් ස්ට්‍රින්ග් එකක් ප්‍රතිදානය කරනු ලබයි.

else:

All subsequent lines of code will be executed only if the value of x is greater than 0.

x හි අගය 0 ට වඩා වැඩිනම් පමණක් මින් පසු පවතින සියළුම කේත පේළි ක්‍රියාත්මක වේ.

for i in range(1,x):

The loop iterates from 1 to x (input number) in a for loop. In each iteration, the corresponding value is substituted for i.

ලූපය 1 සිට x (ආදානය කළ සංඛ්‍යාව) දක්වා for ලූපයක් තුළදී පුනරාවර්තනය වේ. එහිදී සෑම පුනරාවර්තනය වාරයකදීම i සඳහා අදාළ අගය ආදේශ වෙයි.

y=i*i

Then the value of i is multiplied by i. That is, the exponent of i is obtained. It is then substituted for y that value.

ඉන්පසු i හි අගය i ගෙන් ගුණ කරනු ලබයි. එනම් i හි වර්ගය ලබා ගනියි. ඉන්පසු එම අගය y සඳහා ආදේශ කරනු ලබයි.

if y>=x:

break

Then it checks if the value of y is greater than or equal to x and if so, the break statement is executed and the loop ends.

ඉන්පසු y හි අගය x ට වඩා විශාල හෝ සමානද යන්න පරීක්ෂා කරන අතර එසේ වේනම් break විධානය ක්‍රියාත්මක කර ලූපය අවසන් කරයි.

L.append(y)

Otherwise, the value of y is appended to the list L.

එසේ නොවේනම් L ලැයිස්තුව සඳහා y හි අගය append කරයි.

print(L)

Accordingly, by studying this function, it can be recognized that the purpose of this code is to calculate the square numbers from 1 to a given number.

ඒ අනුව මෙම ක්‍රියාකාරීත්වය අධ්‍යයනයෙන් මෙම කේතයේ අරමුණ ලෙස දෙන ලද සංඛ්‍යාවක් දක්වා 1 සිට පවතින වර්ග සංඛ්‍යා ගණනය කිරීම බව හඳුනා ගත හැක.

Assume x=20.

x=20 ලෙස උපකල්පනය කරන්න.

This above code outputs as [1, 4, 9, 16]. The requirement here is to display * points squarely to the size of each item in that list.

මෙම ඉහත කේතයේ [1, 4, 9, 16] ලෙස ප්‍රතිදානය කරයි. මෙහි අවශ්‍යතාවය වන්නේ එම ලැයිස්තුවේ එක් එක් අයිතමයේ ප්‍රමාණයට සමවතුරුකාරව * ලකුණු පෙන්වීමයි.

The output contains square numbers from 0 to 20. There the square root of the square number is obtained and then the number of rows equal to the size of the square root number and the number of * points to the size of the square root number in that one row is output. That is, if the square number of 16 is obtained, its square root will be 4. Then 4 rows with 4 * points per row should be output.

එම ප්‍රතිදානය තුළ පවතින්නේ 0 සිට 20 දක්වා පවතින වර්ග සංඛ්‍යා වේ. එහිදී වර්ග සංඛ්‍යා වල වර්ගමූලය ලබා ගෙන ඉන්පසු එම වර්ගමූල සංඛ්‍යාවේ ප්‍රමාණයට සමාන පේළි සංඛ්‍යාවක් හා එම එක් පේළියක වර්ගමූල සංඛ්‍යාවේ ප්‍රමාණයට * ලකුණු ප්‍රතිදානය කිරීමයි. එනම් 16 යන වර්ග සංඛ්‍යාව ලබා ගත නොහේ එහි වර්ගමූලය 4 වෙයි. ඉන්පසු පේළියකට * ලකුණු 4 බැගින් පේළි 4ක් ප්‍රතිදානය කළ යුතු වේ.

```
print( L* '*' )
```

This is definitely incorrect. The reason is that multiplying a list by a non-integer value generates an error. මෙය අනිවාර්යෙන්ම වැරදි වේ. ඊට හේතුව වන්නේ ලැයිස්තුවක් නිඛිල නොවන අගයකින් ගුණ කළ විට දෝෂයක් ජනනය කිරීමයි.

```
for n in L:
    print( n* '*' )
```

Here * is output to the size of each value in the list L, but not output as a square. Therefore this is also incorrect.

මෙහි L ලැයිස්තුවේ එක් එක් අගයේ ප්‍රමාණයට * ප්‍රතිදානය කළද සමචතුරස්‍රයක් ආකාරයට ප්‍රතිදානය නොකරයි. එම නිසා මෙයද වැරදි වේ.

```
for n in range(L):
    print( n* '*' )
    print( 'next' )
```

This also generates an error because a list cannot be given as a parameter value to the range function. So, this is incorrect.

range ශ්‍රිතයේ පරාමිති අගයක් ලෙස ලැයිස්තුවක් ලබා දිය නොහැකි බැවින් මෙමගින් ද දෝෂයක් ජනනය කරනු ලබයි. එම නිසා මෙය වැරදි වේ.

```
for n in L:
    l=int( n**0.5 )
    for d in range(l):
        print( l* '*' )
    print("next")
```

Here, in a for loop, all the elements in the list L are substituted for n at each stage of repetition and repeated. Then take the square root of n to the square of 0.5 and substitute it for the variable l. A for loop then executes a number of iterations l, multiplying "*" by l at each stage and outputting it. Then when that loop is finished it outputs "next" and moves on to the next iteration of the main loop. So this is the correct answer.

මෙහිදී for ලූපයක් තුළදී L ලැයිස්තුවේ ඇති සියලුම මූලාංග එක් එක් අදියරේදී n සඳහා ආදේශ වී පුනරාවර්තනය වේ. ඉන් පසු න් හි 0.5 වර්ගය එනම් සංඛ්‍යාවේ වර්ග මූලය ලබා ගෙන එය l විචල්‍ය සඳහා ආදේශ වෙයි. ඉන් පසු for ලූපයක් l පුනරාවර්තන ප්‍රමාණයක් ක්‍රියාත්මක වන අතර ඒ එක් එක් අදියරේදී "*" l ගෙන් වැඩි කර එය ප්‍රතිදානය කරයි. ඉන්පසු එම ලූපය අවසන් වූ පසු "next" ලෙස ප්‍රතිදානය කර ප්‍රධාන ලූපයෙහි මීලඟ පුනරාවර්තනය වෙත ගමන් කරයි. එම නිසා මෙය නිවැරදි පිළිතුර වේ.

Therefore, the correct answer is 4). / එබැවින්, නිවැරදි පිළිතුර 4) වේ.

7. `x=[20,25,13,62,47,3]`
`list(reversed(sorted(x)))`

What is the output of the above Python code?

ඉහත පයිතන් කේත කොටසේ ප්‍රතිදානය කුමක්ද?

- 1) `[3, 13, 20, 25, 47, 62]`
- 2) `[3, 47, 62, 13, 25, 20]`
- 3) `[62, 47, 25, 20, 13, 3]`
- 4) `[62, 13, 25, 20, 3, 47]`
- 5) syntax error

Answer : 3)

Let's consider the process of this python code step by step.

මෙම පයිතන් කේතයේ ක්‍රියාවලිය පියවරෙන් පියවර සලකා බලමු.

`x=[20,25,13,62,47,3]`

This is the initial list x containing six integer elements.

මෙය නිඛිල මූලාංග හයක් අඩංගු x ආරම්භක ලැයිස්තුව වේ.

`sorted(x)`

The `sorted(x)` function returns a new list containing all elements of x in ascending order. So, `sorted(x)` would produce:

`sorted(x)` ශ්‍රිතය ආරෝහණ අනුපිළිවෙලට x හි සියලුම මූලාංග අඩංගු නව ලැයිස්තුවක් ලබා දෙයි.

එබැවින්, `sorted(x)` මගින්:

`[3, 13, 20, 25, 47, 62]`

`reversed(sorted(x))`

The `reversed()` function takes an iterable and returns an iterator that yields elements in reverse order.

Applying `reversed()` to `sorted(x)` would reverse the sorted list, resulting in:

`reversed()` ශ්‍රිතය පුනරාවර්තනය කළ හැකි අතර ප්‍රතිලෝම අනුපිළිවෙලින් මූලාංග ලබා දෙයි.

`sorted(x)` වෙත `reversed()` යෙදීමෙන් අනුපිළිවෙලට ගත් ලැයිස්තුව ආපසු හැරවිය හැකි අතර, ප්‍රතිඵලයක් ලෙස:

`<list_reverseiterator object at 0x000002939F1E4910>`

`list(reversed(sorted(x)))`

Finally, the `list()` function converts the iterator returned by `reversed()` into a list. This results in:

අවසාන වශයෙන්, `list()` ශ්‍රිතය මගින් `reversed()` මගින් ලැබූ පුනරාවර්තනය ලැයිස්තුවක් බවට පරිවර්තනය කරයි. මෙහි ප්‍රතිඵලය වන්නේ:

`[62, 47, 25, 20, 13, 3]`

So, the correct answer number is 3).

එබැවින්, නිවැරදි පිළිතුරු අංකය 3) වේ.

```
8. name= 'Piumi'
   age=16
   print('My name is' +name)
   print('I am' +age+ 'years old')
```

What would be the output of above python code segment?

ඉහත පයිතන් කේත කොටසේ ප්‍රතිදානය කුමක් විය හැකි ද?

- 1) My name isPiumi
I am 16 years old
- 2) My name isPiumi and error message
- 3) My name isPiumi I am 16 years old. I am 16 years old
- 4) None of the above
- 5) Display error message only

Answer : 2)

```
print('My name is' + name)
```

Here, 'My name is' is a string literal and name is a variable containing the string 'Piumi'. The + operator concatenates these two strings together. Therefore, the output of this line will be මෙහි 'My name is' යනු string වචනාර්ථයක් වන අතර name යනු 'Piumi' string එක අඩංගු විචල්‍යයකි. + ක්‍රියාකරු මෙම string දෙක එකට සම්බන්ධ කරයි. එබැවින්, මෙම ප්‍රකාශයේ ප්‍රතිදානය වනුයේ

My name is Piumi

```
print('I am' + age + 'years old')
```

In this statement, 'I am' and 'years old' are string literals. age is a variable containing the integer 16. The issue here is that Python does not allow concatenation of a string and an integer directly without conversion. Therefore, this line will raise a TypeError because it tries to concatenate a string ('I am') with an integer (age).

මෙම ප්‍රකාශයේ, 'I am' සහ 'years old' යන්න වචනාර්ථයන් වේ. වයස යනු පූර්ණ සංඛ්‍යාව 16 අඩංගු විචල්‍යයකි. මෙහි ඇති ප්‍රශ්නය නම් පයිතන් string එකක් සහ පූර්ණ සංඛ්‍යාවක් පරිවර්තනයකින් තොරව සෘජුවම එකතුවීම කිරීමට ඉඩ නොදීමයි. එමනිසා, මෙම 'I am' string එකක් පූර්ණ සංඛ්‍යාවක් (age). සමඟ එකතුවීම කිරීමට උත්සාහ කරන නිසා TypeError මතු කරයි.

Since the first print statement will execute successfully and the error occurs in the second print statement, the correct answer is 2) My name isPiumi and error message.

පළමු print ප්‍රකාශය සාර්ථකව ක්‍රියාත්මක වන බැවින් සහ print මුද්‍රණ ප්‍රකාශයේ දෝෂය සිදු වන බැවින්, නිවැරදි පිළිතුර වන්නේ 2) My name isPiumi and error message.

9. Which attribute of the <option> element specifies the value that will be sent to the server when the form is submitted?

<option> මූලාංගයේ කුමන ගුණාංගය, පෝරමය ඉදිරිපත් කරන විට සේවාදායකය වෙත යවනු ලබන අගය සඳහන් කරයිද?

- 1) label 2) value 3) text 4) id 5) content

Answer : 2)

Let's consider each statement in the answers.

පිළිතුරු වල ඇති එක් එක් ප්‍රකාශ සලකා බලමු.

1) label

This statement is false. The label attribute is used to define the visible text of the <option> element, not the value sent to the server.

මෙම ප්‍රකාශය අසත්‍ය වේ. ලේබල් ගුණාංගය <option> මූලද්‍රව්‍යයේ දෘශ්‍ය පෙළ නිර්වචනය කිරීමට භාවිතා කරයි, සේවාදායකය වෙත යවන ලද අගය නොවේ.

2) value

This statement is true. The value attribute of the <option> element specifies the value that will be sent to the server when the form is submitted. For example, in <option value="ict1">, the value ict1 is sent to the server.

මෙම ප්‍රකාශය සත්‍ය වේ. <option> මූලද්‍රව්‍යයේ අගය ගුණාංගය පෝරමය ඉදිරිපත් කරන විට සේවාදායකය වෙත යවනු ලබන අගය සඳහන් කරයි. උදාහරණයක් ලෙස, <option value="ict1"> හි, ict1 අගය සේවාදායකය වෙත යවනු ලැබේ.

3) text

This statement is false. There is no text attribute for <option>. The visible text inside the <option> tags is displayed to the user but does not affect the data sent to the server.

මෙම ප්‍රකාශය අසත්‍ය වේ. <option> සඳහා පාඩ ගුණාංගයක් නොමැත. <option> උසුලන තුළ ඇති දෘශ්‍ය පෙළ පරිශීලකයාට දර්ශනය වන නමුත් සේවාදායකය වෙත යවන ලද දත්තවලට බලපාන්නේ නැත.

4) id

This statement is false. The id attribute is used to uniquely identify an element within the HTML document and is not related to the form submission values.

මෙම ප්‍රකාශය අසත්‍ය වේ. HTML ලේඛනය තුළ ඇති මූලද්‍රව්‍යයක් අනන්‍ය ලෙස හඳුනා ගැනීමට id ගුණාංගය භාවිතා කරන අතර එය පෝරමය ඉදිරිපත් කිරීමේ අගයන් හා සම්බන්ධ නොවේ.

5) content

This statement is false. There is no content attribute for <option>. The content inside the <option> tags represents the text displayed to the user.

මෙම ප්‍රකාශය අසත්‍ය වේ. <option> සඳහා අන්තර්ගත ගුණාංගයක් නොමැත. <option> උසුලන තුළ ඇති අන්තර්ගතය පරිශීලකයාට දර්ශනය වන පෙළ නියෝජනය කරයි.

10. Consider the following part of webpage.

පහත දැක්වෙන වෙබ් පිටුවේ කොටස සලකා බලන්න.

Browser >

Apple Music Trending Playlists

Here is the top 5 songs now on trending.

- I. Fortnight - Taylor Swift
- II. Espresso - Sabrina Carpenter
- III. Popular - The Weeknd

HTML code relevant to above website is also given below.

ඉහත වෙබ් අඩවියට අදාළ HTML කේතය ද පහත දක්වා ඇත.

```
<html>
<head><link rel="stylesheet" href="style.css"/><title>Browser</title></head>
<body>
  <h1>Apple Music Trending Playlists</h1>
  <p id="ul">
    Here is the top 5 songs now on trending.
    <ol type="I">
      <li>Fortnight - Taylor Swift</li>
      <li>Espresso - Sabrina Carpenter</li>
      <li>Popular - The Weeknd</li>
    </ol></p>
</body>
</html>
```

(Consider the text “Here is the top 5 songs now on trending” is in blue colour)

(“Here is the top 5 songs now on trending” යන වාක්‍යය නිල් පැහැයින් ඇති බව සලකන්න)

Which of the following CSS Code is suitable for obtained above output.

ඉහත ප්‍රතිදානය ලබා ගැනීම සඳහා සුදුසු CSS කේතය වන්නේ

- 1) `h1{align:center;color:white;background-color:black;}
#ul{color:blue;}`
- 2) `h1{text-align:center;color:white;background-color:black;}
#ul{color:black;}`
- 3) `h1{text-align:center;color:white;background-color:black;}
#ul{color:blue;}`
- 4) `h1{text-align:center;color:white;bg-color:black;}
#ul{color:blue;}`
- 5) `h1{align:center;color:white;bg-color:black;}
#ul{color:blue;}`

Answer : 3)

Let's consider the above website.

ඉහත සඳහන් වෙබ් අඩවිය සලකා බලමු.



Let's consider each tag one by one. / එක් එක් උසුලන එකින් එක සලකා බලමු.

1) **<h1>** and **</h1>**

Defines a heading, displaying "Apple Music Trending Playlists".

"Apple Music Trending Playlists" සංදර්ශනය කරමින්, මාතෘකාවක් නිර්වචනය කරයි.

2) **<p>** and **</p>**

Defines a paragraph, containing the text "Here is the top 5 songs now on trending".

"Here is the top 5 songs now on trending" යන පාඨය අඩංගු ජේදයක් නිර්වචනය කරයි.

3) **<p id="u1">**

Defines a paragraph, containing the text "Here is the top 5 songs now on trending". Assigns an ID named "u1" to the paragraph element, which can be used for styling purposes.

"Here is the top 5 songs now on trending" යන පාඨය අඩංගු ජේදයක් නිර්වචනය කරයි. විලාසිතා කිරීමේ අරමුණු සඳහා භාවිතා කළ හැකි ජේද මූලාංගයට "u1" නමින් ID එකක් පවරයි.

h1{text-align:center; color:white; background-color:black;}

By using the above code line, we can obtain the above "Apple Music Trending Playlists" text with white font color with black color text background. Also, the text is aligned to the center of the content.

ඉහත කේත රේඛාව භාවිතා කිරීමෙන්, අපට කළු වර්ණ පාඨ පසුබිමක් සහිත සුදු වර්ණ අකුරු සහිත ඉහත "Apple Music Trending Playlists" පාඨය ලබා ගත හැක. එසේම, පාඨය, අන්තර්ගතයේ කේන්ද්‍රයට පෙළගස්වා ඇත.

#u1{color:blue;}

By using the above code line, we can obtain the above "Here is the top 5 songs now on trending." text with font color blue.

ඉහත කේත රේඛාව භාවිතා කිරීමෙන් අපට ඉහත "Here is the top 5 songs now on trending." යන අකුරු වර්ණය නිල් පැහැති පාඨය ලබා ගත හැක.

Option / විකල්ප 1)

This is invalid code line, due to align:center. The correct property for centering text is text-align:center;.

align:center නිසා මෙය වලංගු නොවන කේත රේඛාවකි. පෙළ කේන්ද්‍රගත කිරීම සඳහා නිවැරදි ගුණය වන්නේ text-align:center;.

Option / විකල්ප 3)

This is not suitable code line, due to "Here is the top 5 songs now on trending." text is in blue color, not in black color.

මෙය සුදුසු කේත රේඛාවක් නොවේ, "Here is the top 5 songs now on trending." පාඨය ඇත්තේ නිල් පාටින් මිස කළු පාටින් නොවේ.

Option / විකල්ප 4)

This is not suitable code line, due to the correct syntax for setting the background color is background-color, not bg-color.

මෙය සුදුසු කේත රේඛාවක් නොවේ, පසුබිම් වර්ණය සැකසීම සඳහා නිවැරදි කේත ඛණ්ඩය background-color වන අතර bg-color නොවේ.

Option / විකල්ප 5)

This is not suitable code line, due to the correct property for centering text is text-align, not align. Also, the correct property for setting the background color is background-color, not bg-color.

මෙය සුදුසු කේත රේඛාවක් නොවේ, පාඨ කේන්ද්‍රගත කිරීම සඳහා ඇති නිවැරදි ගුණාංගය නිසා text-align වන අතර align නොවේ. එසේම, පසුබිම් වර්ණය සැකසීම සඳහා නිවැරදි මූලාංගය වන්නේ background-color වන අතර bg-color නොවේ.