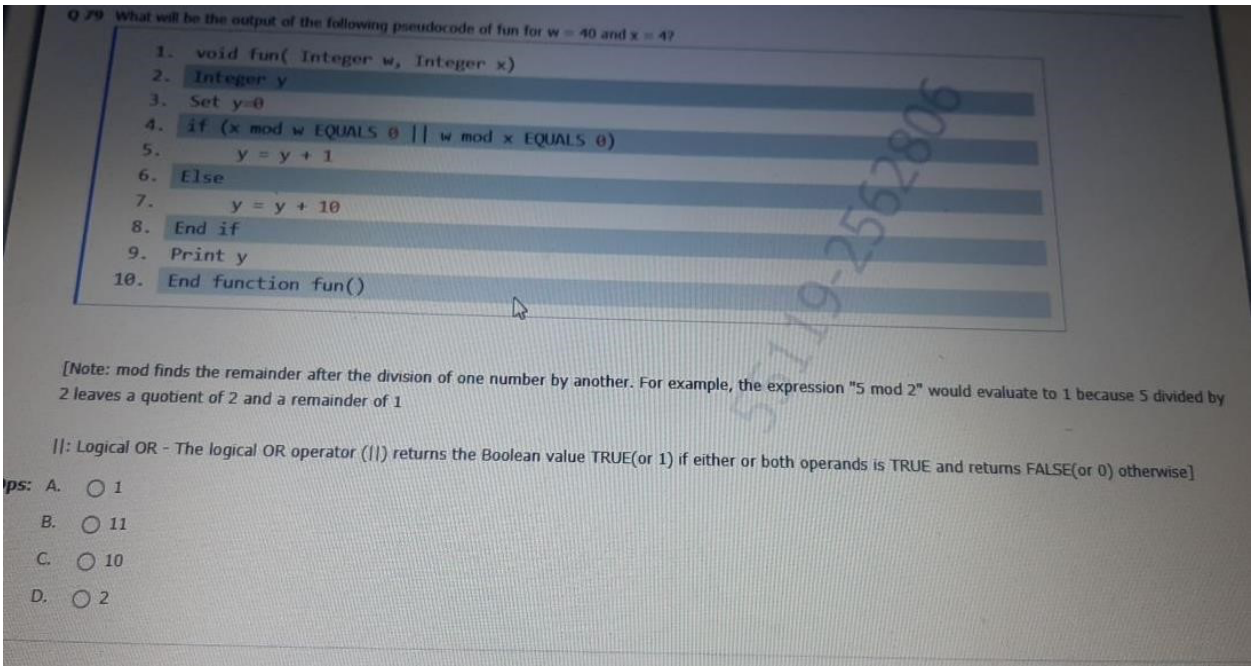
You might not be familiar with the snapshots of the solutions results. It's written in Python and executed using Google Colab. Python codes are very similar to pseudocodes, so the best way to verify answer of pseudocodes is by writing the code using Python :

Q1)



Q 79 What will be the output of the following pseudocode of fun for w = 40 and x = 4?

```
1.  void fun( Integer w, Integer x)
2.  Integer y
3.  Set y=0
4.  if (x mod w EQUALS 0 || w mod x EQUALS 0)
5.      y = y + 1
6.  Else
7.      y = y + 10
8.  End if
9.  Print y
10. End function fun()
```

[Note: mod finds the remainder after the division of one number by another. For example, the expression "5 mod 2" would evaluate to 1 because 5 divided by 2 leaves a quotient of 2 and a remainder of 1

||: Logical OR - The logical OR operator (||) returns the Boolean value TRUE(or 1) if either or both operands is TRUE and returns FALSE(or 0) otherwise]

ps: A. 1

B. 11

C. 10

D. 2

Solution :

```
[ ] def fun(w,x):
      y=0
      if((x%w==0) or (w%x)==0):
        y=y+1
      else:
        y=y+10
      print(y)
    print(fun(40,4))
```

https://github.com/amirkhan1092

Q2)

Q 63 Consider the pseudocode mentioned below. For how many times, the while loop will be executed?

```
Integer a
Set a = 1
while(a < 5)
        a = a + 2
end while
Print a
```

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

Solution :

When while loop is executed 1st time, a is incremented to 3, next time again it enters loop and a is incremented to 5. Next time, while loop is not entered. Hence while loop is executed twice.

Q3)

Q 64 What will be the output of the following pseudocode for a = 8, b = 8?

```
Integer funn(Integer a, Integer b)
    if(a && b && a+b > 0 )
        return a + funn(a - 2, b - 2) + b
    End if
    return a ^ b
End function funn()
```

[Note- &&: Logical AND - The logical AND operator (&&) returns the Boolean value true(or 1) if both operands are true and return false (or 0) otherwise.

^ is the bitwise exclusive OR operator that compares each bit of its first operand to the corresponding bit of its second operand. If one bit is 0 and the other bit is 1, the corresponding result bit is set to 1. Otherwise, the corresponding result bit is set to 0.]

Ops:
- A. ○ 39
- B. ○ 46
- C. ○ 48
- D. ○ 40

Solution :

```python
def fun(a,b):
  if(a and b and (a+b)>0 ):
    return a + fun(a-2,b-2)+b
  return a ^ b
res = fun(8,8)
print(res)
```

```
48
```

Q4)

https://github.com/amirkhan1092

Q 66 What will be the output of the following pseudocode?

```
Integer a, b
Set  a = 3, b = 3
a = b
if(1^1)
    a = 1
Else
    b = 2
End if
Print a + b
```

[Note: ^ is the bitwise exclusive OR operator that compares each bit of its first operand to the corresponding bit of its second operand. If one bit is 0 and the other bit is 1, the corresponding result bit is set to 1. Otherwise, the corresponding result bit is set to 0

If(x) gets executed if the value inside if(), i.e., x is not zero]

s: A. ○ 16

B. ○ 1

C. ○ 5

D. ○ 6

Solution :

```
a=3
b=3
if(1^1):
  a=1
else:
  b=2
print(a+b)
```

Q5)

```
Integer funn(Integer a, Integer b)
    Integer c
    for (each c from 2 to 4 )
        if(a mod 2 < b mod 3)
            a = 4 mod 3
        Else
            if(5 mod 3 > b)
                a = b
            End if
            b = 1
        End if
    End for
    return a + b
End function funn()
```

[Note- mod finds the remainder after the division of one number by another. For example, the expression "5 mod 2" would evaluate to 1 because 5 divided by 2 leaves a quotient of 2 and a remainder of 1.]

s: A. ○ 23

B. ○ -2

C. ○ 6

D. ○ 8

Solution : Option wrongly given as –2. Should be 2 as executed by Python program below.

```python
def fun(a,b):
  for c in range(2,5):
    if((a%2) < (b%3)):
      a=4%3
    if(5%3>b):
      a=b
    b=1
  return(a+b)
print(fun(7,5))
```

2

Q6)

Q 78 What will be the output of the following pseudocode for a = 4, b = 3?

```
Integer funn(Integer a, Integer b)
    if(a > 0)
        return funn(a - 2, a + b) + funn(a - 3, a + b) + funn(a - 4, a + b)
    Else
        a = b
        b = a
        return a + b
    End if
End function funn()
```

ps: A. 116

B. 117

C. 114

D. 125

Solution :

```
[ ] def fun(a,b):
      if(a>0):
        return fun(a-2,a+b) + fun(a-3,a+b) + fun(a-4,a+b)
      else:
        a=b
        b=a
        return(a+b)
    print(fun(4,3))
```

Q7)

https://github.com/amirkhan1092

Q 77 What will be the output of the following pseudocode for a = 6, b = 1?

```
Integer funn(Integer a, Integer b)
    a = a + a
    b = b + b
    return a + b
End function funn()
```

Ops:

A. 23

B. 14

C. 12

D. 16

Solution : a = 6+6 = 12 ; b = 1+1 = 2. Return 12+2 that is 14. Hence ans is 14.

```
def fun(a,b):
  a=a+a
  b=b+b
  return a+b
print(fun(6,1))
```

14

Q8)

Q 76 What will be the output of the following pseudocode?

```
Integer a, b, c
Set  a = 2, b = 4, c = 2
if(b - a)
    b = a ^ b
    a = c
    if(b)
        a = a ^ b
    End if
    b = b - 1
End if
if(c)
    a = b
End if
Print a + b + c
```

[Note- ^ is the bitwise exclusive OR operator that compares each bit of its first operand to the corresponding bit of its second operand. If one bit is 0 and the other bit is 1, the corresponding result bit is set to 1. Otherwise, the corresponding result bit is set to 0.

If(x) gets executed if the value inside if(), i.e., x is not zero]

Ops: A. ○ 13

B. ○ 12

C. ○ 30

D. ○ 9

```
a=2
b=4
c=2
if(b-a):
  b= a ^ b
  a=c
  if(b):
    a = a ^ b
  b=b-1
if(c):
  a=b
print(a+b+c)
```

Solution :

https://github.com/amirkhan1092

Q9)

```
Integer a, b, c
Set  a = 2, b = 3
for (each c from 4 to 6)
    a = a + b
    if(a > 4)
        a = 0
    End if
    if(a + 2)
        b = a + 10
    Else
        Jump out of the loop
    End if
    b = a + 1
End for
Print a + b
```

[Note: If(x) gets executed if th value inside if(), i.e., x is not zero]

s: A. -7

B. 23

C. 7

D. 15

Solution :

```
a=2
b=3
for c in range(4,7):
  a=a+b
  if(a>4):
    a=0
  if(a+2):
    b=a+10
  else:
    break
  b=a+1
print(a+b)
```

Q10)

Q 73 What will be the output of the following pseudocode?

```
Integer m, n
Set m = 9, n = 6
m = m + 1
n = n - 1
m = m + n
if(m>n)
        print m
 else
        print n
end if
```

Ops:

A. ○ 5

B. ○ 10

C. ○ 15

D. ○ 6

Solution :

```
[ ]  m=9
     n=6
     m = m+1
     n = n-1
     m = m+n
     if(m>n):
       print(m)
     else:
       print(n)
```

Q11)

Q 72 What will be the output of the following pseudocode?

```
Integer p, q, r
Set  p = 1, q = 4, r = 2
if (p^q > p&q)
      r = p&q
Else
      r = p ^q
End if
if (q > r && r > p)
      p = q
End if
Print p - q + r - 2
```

[Note- &&: Logical AND - The logical AND operator (&&) returns the Boolean value true(or 1) if both operands are true and return false (or 0) otherwise.

&: bitwise AND - The bitwise AND operator (&) compares each bit of the first operand to the corresponding bit of the second operand. If both bits are 1, the corresponding result bit is set to 1. Otherwise, the corresponding result bit is set to 0.

^ is the bitwise exclusive OR operator that compares each bit of its first operand to the corresponding bit of its second operand. If one bit is 0 and the other bit is 1, the corresponding result bit is set to 1. Otherwise, the corresponding result bit is set to 0.]

A. -5

B. -4

C. -12

D. 3

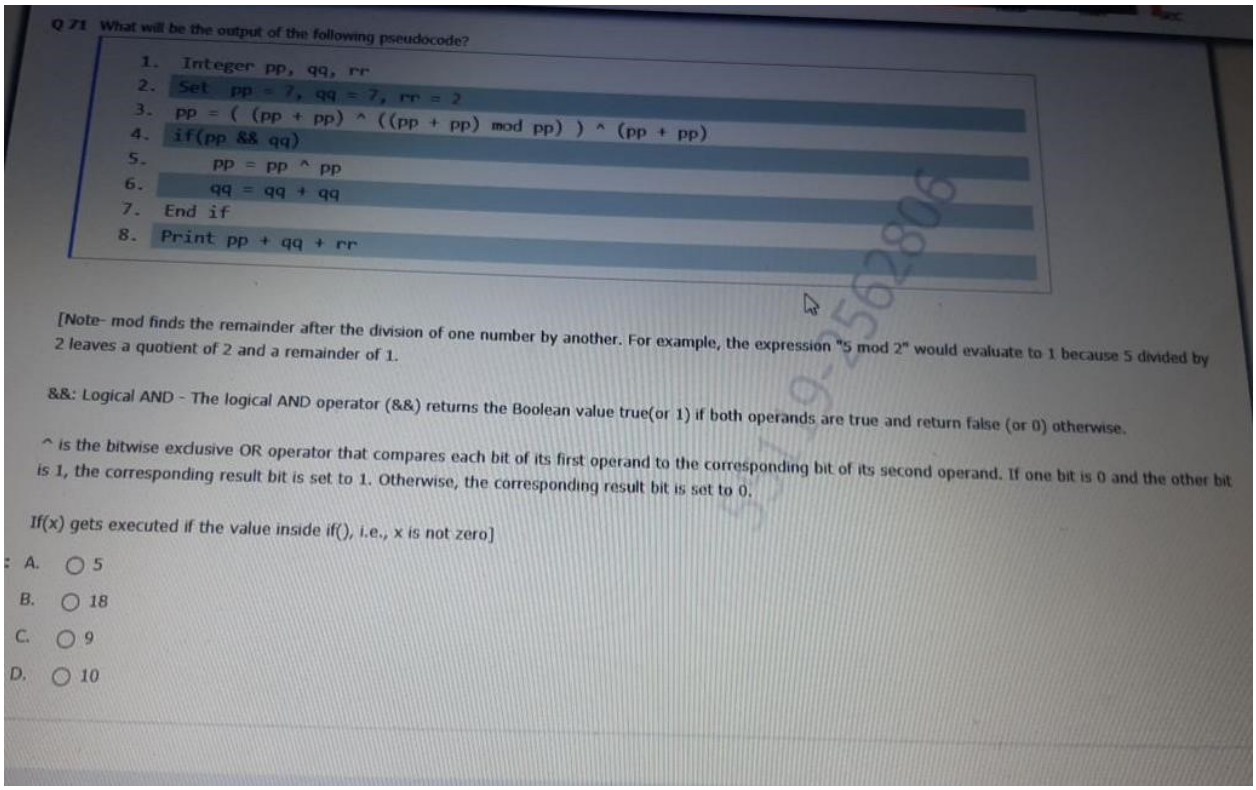
Solution :

https://github.com/amirkhan1092

```
[ ] p=1
    q=4
    r=2
    if((p^q)>p&q):
      r= p&q
    else:
      r=p^q

    if(q>r and r>p):
      p=q
    print(p-q+r-2)
```

Q12)



Q 71 What will be the output of the following pseudocode?

```
Integer pp, qq, rr
Set  pp = 7, qq = 7, rr = 2
pp = ( (pp + pp) ^ ((pp + pp) mod pp) ) ^ (pp + pp)
if(pp && qq)
    pp = pp ^ pp
    qq = qq + qq
End if
Print pp + qq + rr
```

[Note- mod finds the remainder after the division of one number by another. For example, the expression "5 mod 2" would evaluate to 1 because 5 divided by 2 leaves a quotient of 2 and a remainder of 1.

&&: Logical AND - The logical AND operator (&&) returns the Boolean value true(or 1) if both operands are true and return false (or 0) otherwise.

^ is the bitwise exclusive OR operator that compares each bit of its first operand to the corresponding bit of its second operand. If one bit is 0 and the other bit is 1, the corresponding result bit is set to 1. Otherwise, the corresponding result bit is set to 0.

If(x) gets executed if the value inside if(), i.e., x is not zero]

A. 5

B. 18

C. 9

D. 10

Solution :

https://github.com/amirkhan1092

```python
pp=7
qq=7
rr=2
pp = ((pp+pp) ^ ((pp+pp) % pp)) ^ (pp+pp)
if (pp and qq):
  pp = pp ^ pp
  qq = qq + qq
print(pp+qq+rr)
```

```
0
```

Q13) Coding Round Question 1 :

accenture

Hide problem statement

Problem statement

You are required to implement the following function:

```
def InitialsName(name):
```

The function accepts a string 'name' as its argument. The string 'name' is composed of first, middle (optional) and last name separated with a space. Initials of a 'name' are written using the first letter of first name followed by a dot('.') and space, first letter of middle name (if present) followed by a dot('.') and space and retaining the last name as it is. You are required to return the initials of string 'name'.

Note:

- last name in initials are retained
- If 'name' = NULL, return NULL. Incase of python if 'name' is None return None.
- 'name' can have only English alphabets and dot ('.')

Assumptions:

- Length of string 'name' ≤ 100000 characters

Example:

Input:

name: Abc Xyz Pqrst

Output:

A. X. Pqrst

Explanation:

Instru

Now le

https://github.com/amirkhan1092

accenture Welcome 509457

space and retaining the last name as it is. You are required to return the initials of string 'name'.

Note:

- last name in initials are retained
- If 'name' = NULL, return NULL. Incase of python if 'name' is None return None.
- 'name' can have only English alphabets and dot ('.')

Assumptions:

- Length of string 'name' ≤ 100000 characters

Example:

Input:

name: Abc Xyz Pqrst

Output:

A. X. Pqrst

Explanation:

The initials of name "Abc Xyz Pqrst", using the first letter of first name followed by a dot ('.') and space, middle name followed by a dot ('.') and space and retaining the last name is "A. X. Pqrst".

Sample Input

```
name: Mnas Cdefgh
```

Sample Output

```
M. Cdefgh
```

Now

1 2 3 4 5 6 7 8 9

user

Solution :

https://github.com/amirkhan1092

```python
name=input()

def InitialsName(name):
    [redacted]
    for i in range(len(name)):
        if(i == [redacted]):
            [redacted]
        else:
            [redacted] + [redacted] + ". "
    return [redacted]
p = [redacted]
print(p)
```

```
Mnas Cdefgh
M. Cdefgh
```

https://github.com/amirkhan1092