**Code Review Practice**

**Task1.**

A sequence of 10 integers is received for processing. It is known that the entered numbers do not exceed 106 in absolute value. You need to write a program that displays the number of non-negative numbers in a sequence and their product. If there are no non-negative numbers, display "NO" on the screen. The programmer was in a hurry and wrote the program incorrectly. Find all the errors in this program (there are exactly 4). It is known that each error affects only one line and can be fixed without changing other lines. Note 1. The number x does not exceed 106 in absolute value if − 106 ≤ x ≤106 . Note 2: If necessary, you can add the necessary lines of code.

count = 0

p = 0

for i in range(1, 10):

x = int(input())

if x > 0:

p = p \* x

count = count + 1

if count > 0:

print(x)

print(p)

else:

print('NO')

**Task2.**

A sequence of 10 integers is received for processing. It is known that the entered numbers do not exceed 106 in absolute value. You need to write a program that displays the sum of all negative numbers in a sequence and the maximum negative number in the sequence. If there are no negative numbers, display "NO" on the screen. The programmer was in a hurry and wrote the program incorrectly. Find all the errors in this program (there are exactly 5 of them). It is known that each error affects only one line and can be fixed without changing other lines. Note 1. The number x does not exceed 106 in absolute value if − 106 ≤ x ≤ 106. Note 2: If necessary, you can add the necessary lines of code.

mx = 0

s = 0

for i in range(11):

x = int(input())

if x < 0:

s = x

if x > mx:

mx = x

print(s)

print(mx)

**Task3.**

A sequence of 7 integers is received for processing. It is known that the entered numbers do not exceed 106 in absolute value. You need to write a program that calculates and prints the sum of all even numbers in a sequence, or 0 if there are no even numbers in the sequence. The programmer was in a hurry and wrote the program incorrectly. Find all the errors in this program (there are exactly 4). It is known that each error affects only one line and can be fixed without changing other lines. Note 1. The number x does not exceed 106 in absolute value if − 106 ≤ x ≤ 106. Note 2: If necessary, you can add the necessary lines of code.

s = 1

for i in range(1, 7):

n = input()

if i % 2 == 0:

s = s + n

print(s)

**Task4.**

A natural number is being processed. You need to write a program that displays the maximum digit of a number that is a multiple of 3. If the number does not contain digits that are multiples of 3, you need to display "NO" on the screen. The programmer was in a hurry and wrote the program incorrectly. Find all the errors in this program (there are exactly 5 of them). It is known that each error affects only one line and can be fixed without changing other lines. Note 1. The number 0 is divisible by any natural number. Note 2: If necessary, you can add the necessary lines of code.

n = int(input())

max\_digit = n % 10

while n > 0:

digit = n % 10

if digit % 3 == 0:

if digit < max\_digit:

digit = max\_digit

n = n % 10

if max\_digit == 0:

print('NO')

else:

print(max\_digit)

**Task5.**

A natural number is being processed. You need to write a program that displays its first (highest) digit on the screen. The programmer was in a hurry and wrote the program incorrectly. Find all the errors in this program (there are exactly 2 of them). It is known that each error affects only one line and can be fixed without changing other lines.

n = int(input())

while n > 0:

n %= 10

print(n)

**Task6.**

A natural number is being processed. You need to write a program that displays the product of the digits of the entered number. The programmer was in a hurry and wrote the program incorrectly. Find all the errors in this program (there are exactly 3 of them). It is known that each error affects only one line and can be fixed without changing other lines.

n = input()

product = n % 10

while n >= 10:

digit = n % 10

product = product \* digit

n //= 10

print(product)