

Limited factor number

To determine whether a given positive integer (x) is a limited factor number based on the criteria provided, we can implement an algorithm that checks if the prime factors of (x) consist only of the values 2, 3, and 5.

Here's how you can do it:

Steps:

1. If ($x = 1$), return true (since it has no prime factors).
2. Divide (x) by 2 until it is no longer divisible by 2.
3. Do the same for 3 and then for 5.
4. After removing all factors of 2, 3, and 5, if the resulting number is 1, return true. Otherwise, return false.

Implementation:

Here is a Python function that implements the logic described:

```
def is_limited_factor_number(x):
    # Step 1: Check if x is 1
    if x == 1:
        return True

    # Step 2: Remove factors of 2
    while x % 2 == 0:
        x //= 2

    # Step 3: Remove factors of 3
    while x % 3 == 0:
        x //= 3

    # Step 4: Remove factors of 5
    while x % 5 == 0:
        x //= 5

    # Step 5: Check if the resulting number is 1
    return x == 1

# Example Usage
print(is_limited_factor_number(6)) # Output: True
print(is_limited_factor_number(1)) # Output: True
print(is_limited_factor_number(14)) # Output: False
```

Explanation:

- For input ($x = 6$):
- (6) can be factored into (2×3). It only uses allowed factors, so return true.
- For input ($x = 1$):
- It has no prime factors, thus return true.
- For input ($x = 14$):
- (14) includes (7) which is not among (2, 3, 5). Thus, return false