

84. Karatsuba algorithm for multiplication

PROGRAM:-

```
import time
```

```
def karatsuba(x, y):
    # Base case for recursion
    if x < 10 or y < 10:
        return x * y

    # Calculate the size of the numbers
    n = max(len(str(x)), len(str(y)))
    m = n // 2

    # Split the digit sequences about the middle
    high1, low1 = divmod(x, 10**m)
    high2, low2 = divmod(y, 10**m)

    # Perform 3 multiplications
    z0 = karatsuba(low1, low2)
    z1 = karatsuba((low1 + high1), (low2 + high2))
    z2 = karatsuba(high1, high2)

    # Combine the results
    return (z2 * 10**(2*m)) + ((z1 - z2 - z0) * 10**m) + z0

def find_karatsuba_time(x, y):
    start_time = time.time() # Start time measurement

    result = karatsuba(x, y) # Perform Karatsuba multiplication

    end_time = time.time() # End time measurement
    elapsed_time = end_time - start_time

    return result, elapsed_time

# Example usage
x = 12345678901234567890
y = 98765432109876543210

result, execution_time = find_karatsuba_time(x, y)

print(f"Multiplication result: {result}")
print(f"Execution time: {execution_time:.10f} seconds")
```

OUTPUT:-

```
Multiplication result: 1219326311370217952237463801111263526900  
Execution time: 0.0000848770 seconds  
  
=== Code Execution Successful ===
```

TIME COMPLEXITY:- $O(n^{\log_2 3}) \approx O(n^{1.585})$