**Problem Statement:**

Design and implement **large integer multiplication** using the **Divide and Conquer method**. Determine the time required to multiply two large integers.

**Brief About the Problem:**

Multiplication of large integers is an essential problem in computer science, especially in **cryptography, numerical computing, and big-number arithmetic**. Traditional multiplication methods (such as the schoolbook method) become inefficient as the number of digits increases.

To optimize large integer multiplication, **Divide and Conquer methods** like **Karatsuba’s Algorithm** and **Strassen-like methods** can be used to reduce the number of required multiplications.

**Code:**

**#include <iostream>**

**#include <chrono>**

**#include <cmath>**

**using namespace std;**

**using namespace std::chrono;**

**// Karatsuba multiplication (using only int and no strings)**

**int karatsuba(int x, int y) {**

**if (x < 10 || y < 10) {**

**return x \* y;**

**}**

**// Determine the number of digits in the larger number**

**int temp\_x = x, temp\_y = y;**

**int n\_x = 0, n\_y = 0;**

**while (temp\_x > 0) {**

**temp\_x /= 10;**

**n\_x++;**

**}**

**while (temp\_y > 0) {**

**temp\_y /= 10;**

**n\_y++;**

**}**

**int n = max(n\_x, n\_y);**

**n = (n + 1) / 2;**

**int power\_of\_10 = pow(10, n); // Potential Overflow Here!!!**

**// Split the numbers into two halves**

**int a = x / power\_of\_10;**

**int b = x % power\_of\_10;**

**int c = y / power\_of\_10;**

**int d = y % power\_of\_10;**

**// Recursively calculate the three products**

**int ac = karatsuba(a, c);**

**int bd = karatsuba(b, d);**

**int ad\_bc = karatsuba(a + b, c + d) - ac - bd;**

**// Combine the results**

**return ac \* (int)pow(10, 2 \* n) + ad\_bc \* (int)pow(10, n) + bd; // Potential Overflow Here!!!**

**}**

**int main() {**

**int num1, num2;**

**cout << "Enter the first integer: ";**

**cin >> num1;**

**cout << "Enter the second integer: ";**

**cin >> num2;**

**auto start = high\_resolution\_clock::now();**

**int result = karatsuba(num1, num2);**

**auto stop = high\_resolution\_clock::now();**

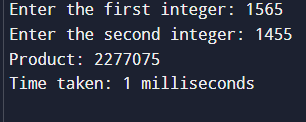
**auto duration = duration\_cast<milliseconds>(stop - start);**

**cout << "Product: " << result << endl;**

**cout << "Time taken: " << duration.count() << " milliseconds" << endl;**

**return 0;**

**}**

****