1 Grade-school multiplication

Suppose we multiply two \( n \)-digit integers \((x_1x_2\ldots x_n)\) and \((y_1y_2\ldots y_n)\) using the grade-school multiplication algorithm. How many pairs of digits \(x_i\) and \(y_j\) get multiplied in this algorithm?

- \(n^3\)
- \(2n - 1\)
- \(n^2\)  \(\text{Correct}\)

What is the smallest exponent \(x\) such that the number of one-digit operations in grade-school multiplication is always at most \(10000 \cdot n^x\)?

\[
2
\]

Correct

2 Divide-and-conquer multiplication

Suppose that we have a divide-and-conquer algorithm \(A\) that multiplies two \( n \)-digit integers by recursively calling itself to perform \(t\) number of \([n/2]\)-digit integer multiplications; when \(n \leq 1\), it performs single-digit multiplication.

If \(t = 4\), what is the smallest exponent \(x\) such that the number of one-digit multiplications is always at most \(10000 \cdot n^x\)?

\[
2
\]

Correct

For what values of \(t\) does the algorithm perform fewer one-digit multiplications than the grade-school multiplication algorithm for inputs that have \(n > 10000\) digits?

- For all values of \(t\)
- \(t = 1, 2\)
- \(t = 1, 2, 3\)
- \(t = 1, 2, 3, 4\)  \(\text{Correct}\)

What is the value of \(t\) for Karatsuba integer multiplication algorithm?

\[
3
\]

Correct