Integer Multiplication

Input :- Two n-digit nonnegative integer, x and y

Output :- The product $x \cdot y$

Ex :-

$$
\begin{array}{r}
5678 \\
1234 \\
\hline
22712 \\
17034 \\
11356 \\
5678 \\
\hline
7006652
\end{array}
$$

Basic operations :-

computing a partial product Requires

n multiplication (one per digit)

at most n addition (per digit)

Total per digit $\leq 2n$

There are n digit so

at most $2n^2$ operation required

Divide & Conquer

1- Divide the input into smaller subproblems

2. Conquer the subproblem recursively

3- Combine the solution for the subproblems into a solution for the original problem.

Karatsuba Multiplication

a = 56, b = 78

c = 12, d = 34

1- Compute $a*c = 56*12$ (672)

2- Compute $b \cdot d = 78*34$ (2652

3- compute $(a+b)\cdot(c+d) = 134 \cdot 46 = 6164$

4. Subtract the result of step1 and step2 from step 3

~~6164 - 672 - 2656 = 2840~~

$6164 - 672 - 2652 = 2840$

Step 5 Compute $10^4 \cdot \overset{(a \cdot c)}{672} + 10^2 \, \overset{\uparrow}{2840} + \overset{b \cdot d}{2652}$

$= 70066552$

Recursive algorithm

A number x with even number of n digit

Can be expressed as

$X = \cancel{10^{n/2} \cdot (a+b)}$

$X = 10^{n/2} a + b$

like $\overset{a}{2}\overset{b}{3} = 2 * 10^1 + 3 \quad (n = 2)$

Another number $\overset{c}{1}\overset{d}{4} = 1 * 10^1 + 4$

Let

$(23 \times 14) = (2 * 10^1 + 3)(1 * 10^1 + 4)$

$= \underset{(a*c)}{(2*1)} 10^2 + (3 \times 1 + 2 * 4) 10^1 + \underset{(b*d)}{(3*4)} 10^0$

$= \cancel{(a*c) 10^2 + 10^1 (a \cdot d + b \cdot c) + b \cdot d}$

$= (a * c) 10^2 + (a \cdot d + b \cdot c) 10^1 + bd$

final one look like

$$(X * y) = 10^{n}(a\cdot c) + 10^{n/2}(a\cdot d + b\cdot d) + b\cdot d$$

Rec. Algo. Int. Mul.

Input two n-digit positive integer x and y

output the product x*y

Assumption n is a power of 2

If $n = 1$ then

compute $x * y$ in one step and return the result.

else

$a, b :=$ first & second halves of x

$c, d :=$ " " " " " y

recursively compute

$ac := a\cdot c$

$ad := a\cdot d$

$bc := b\cdot c$

$bd := b\cdot d$

compute $10^{n}\cdot ac + 10^{n/2}(ad + bc) + bd$

It requires 4 recursive call.

How many digit multiplication does this algorithm takes?

$$M(n) = 4\, M(n/2) \quad \text{for } n > 1$$
$$M(1) = 1$$

Let $n = 2^k$

$$M(2^k) = 4 \cdot M(2^{k-1})$$
$$= 4 \cdot [4\, M(2^{k-2})] \quad \text{(backward substitution)}$$
$$= 4^2 \cdot M(2^{k-2})$$
$$= 4^i\, M(2^{k-i})$$
$$= 4^k\, M(2^{k-k})$$
$$= 4^k \quad (k = \log_2 n)$$
$$= 4^{\log_2 n}$$
$$= n^{\log_2 4} \quad (a^{\log_b c} = c^{\log_b a})$$
$$= \boxed{n^2}$$

Improvement

we have to compute

$$10^n \cdot ac + 10^{n/2}(ad + bc) + bd$$

Recall step 4 of Karatsuba algorithm

normal addition

$$\boxed{(a+b)\cdot(c+d) - a*c - b*d}$$

already calculated

~~So (ad + bc) = a+~~

$$\cancel{ac} + ad + bc + \cancel{bd} - \cancel{ac} - \cancel{bd}$$

$$(ad + bc)$$

Final one Karatsuba

I.

Input :- Two n digit positive integer x and y

output :- product $x \cdot y$

Assumption : n is a power of 2

If $n = 1$ then

compute $x * y$ in one step and return the result

else

$a, b :=$ first and second halves of x

$c, d :=$ " " " " " y

Compute $p := a + b$

$q := c + d$

recursively compute

$(a \cdot c)$

$(b \cdot d)$

$(p \cdot q)$

compute $(ad + bc) := p.q - ac - bd$

compute $10^n \cdot ac + 10^{n/2} \cdot (ad + bc) + bd$

8

$M(n) = 3M(n/2) \quad n > 1$

$M(1) = 1$

Let $n = 2^k$ and use backward substitution

$M(2^k) = 3M(2^{k-1})$

$= 3[3M(2^{k-2})]$

$= 3^2 M(2^{k-2})$

$\vdots$

$= 3^i M(2^{k-i})$

$\vdots$

$= 3^k M(2^{k-k})$

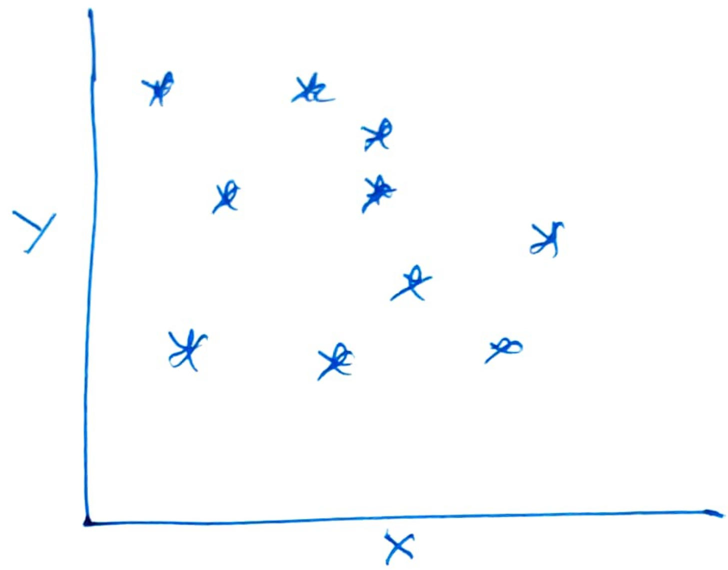
$= 3^k$

$= 3^{\log_2 n}$

$= n^{\log_2 3}$

$\approx n^{1.585}$

Closest pair problem



distance

$$d(P_i, P_j) = \sqrt{(x_i - x_j)^2 + (y_1 - y_j)^2}$$

Brute Force Closest pair

Input :- A list P of $n\ (n \geq 2)$ points

output :- Indices index1 and index2 of closest pair

Let $dmin \leftarrow \infty$

for $i \leftarrow 1$ to $n-1$ do

  for $j \leftarrow i+1$ to n do

    $d \leftarrow sqrt((x_i - x_j)^2 + (y_i - y_j)^2$

    if $d < dmin$

      $dmin \leftarrow d$; $index1 \leftarrow i$; $index2 \leftarrow j$

return index1, index2

Basic operation - squaring a number and square root.

we can ignore square root

$$C(n) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} 2$$

$$= 2\sum_{i=1}^{n-1} (n-i)$$

by $\sum_{i=l}^{u} 1 = u - l + 1$

$$\sum_{j=i+1}^{n} = n-(i+1)+1$$

$$= n-i$$

$$= 2\left[(n-1)+(n-2)+ \cdots + 1\right] \; (\cancel{\text{...}}$$

How to solve it?