DAC Example - Multiplication of n-bit (n-digit) integers

Input: $A = a_1 a_2 \cdots a_n$ $\quad B = b_1 b_2 \cdots b_n$ $\quad (a_i, b_i$ - bits/digits, $1 \le i \le n)$

Long multiplication method for computing $A \cdot B$ takes $O(n^2)$ time.

Can DAC yield a better solution?

Take 1: Split A into $\boxed{A_1 | A_2}$ ($n/2$ bit numbers)

and B into $\boxed{B_1 | B_2}$

Compute A_1B_1, A_1B_2, A_2B_1, A_2B_2 recursively.

Output $A_1B_1 * Base^n + (A_1B_2 + A_2B_1) * Base^{n/2} + A_2B_2$

($Base^n$ and $Base^{n/2}$ are shift operations)

RT: $T(n) = 4T(n/2) + n$ ← (2 shifts, 3 add operations)

Using MM, $T(n) = O(n^2)$

Take 2: Use the identity:

$$A_1B_2 + A_2B_1 = (A_1+A_2)(B_1+B_2) - A_1B_1 - A_2B_2$$

Compute A_1B_1, A_2B_2, $(A_1+A_2)(B_1+B_2)$ recursively.

Output: $A_1B_1\,Base^n + \left[(A_1+A_2)(B_1+B_2) - A_1B_1 - A_2B_2\right] * Base^{n/2} + A_2B_2$

RT: $T(n) = 3T(n/2) + n$ ← (2 shifts, 6 add operations, 2 subtract operations)

Using MM, $T(n) = O(n^{\log 3}) = O(n^{1.585})$

Multiplication of n-bit (n-digit) integers : Example

3849 × 7106

Long Multiplication method:

```
  3849 x 7106
 -----------
   23094   <- 3849x6
   0000    <- 3849x0
  3849     <- 3849x1
 26943     <- 3849x7
 -----------
 27350994
```

DAC Method #1

$\boxed{38}\boxed{49} \times \boxed{71}\boxed{06}$

$38 \times 71 = 2698$

$38 \times 06 = 228$

$49 \times 71 = 3479$

$49 \times 06 = 294$

```
 26980000
    22800
   347900
      294
 ---------
 27350994
```

DAC Method #2

$38 \times 71 = 2698$

$49 \times 06 = 294$

$(38+49) * (71+06) = 6699$

$38 \times 06 + 49 \times 71 = 6699 - 2698 - 294$

$= 3707$

```
 26980000
   370700
      294
 ---------
 27350994
```