Implementar o Algoritmo de Multiplicação Padrão e o Algoritmo de Karatsuba em Java

Algorithm 1 Multiplicação Padrão.

1: **procedure** MULT(a, b, n)
2: **if** $n > 1$ **then**
3: $A1 \leftarrow \left[\frac{a}{2^{n/2}}\right]$
4: $A0 \leftarrow a \mod 2^{n/2}$
5: $B1 \leftarrow \left[\frac{b}{2^{n/2}}\right]$
6: $B0 \leftarrow b \mod 2^{n/2}$
7: $A1B1 \leftarrow MULT(A1, B1, n/2)$
8: $A1B0 \leftarrow MULT(A1, B0, n/2)$
9: $A0B1 \leftarrow MULT(A0, B1, n/2)$
10: $A0B0 \leftarrow MULT(A0, B0, n/2)$
11: **return** $2^n * A1B1 + 2^{n/2}(A1B0 + A0B1) + A0B0$
12: **else**
13: **return** $a * b$

Algorithm 2 Multiplicação Karatsuba.

1: **procedure** MULT(a, b, n)
2: **if** $n > 1$ **then**
3: $A1 \leftarrow \left[\frac{a}{2^{n/2}}\right]$
4: $A0 \leftarrow a \mod 2^{n/2}$
5: $B1 \leftarrow \left[\frac{b}{2^{n/2}}\right]$
6: $B0 \leftarrow b \mod 2^{n/2}$
7: $A1B1 \leftarrow MULT(A1, B1, n/2)$
8: $A0B0 \leftarrow MULT(A0, B0, n/2)$
9: $M3 \leftarrow MULT(A1 + A0, B1 + B0, n/2)$
10: **return** $2^n * A1B1 + 2^{n/2}(M3 - A1B1 - A0B0) + A0B0$
11: **else**
12: **return** $a * b$

1. Algoritmos para realizar a multiplicação de dois número qualquer com uma quantidade arbitrária de dígitos.
2. Os algoritmos devem ser desenvolvidos para funcionar para as bases Binária e Decimal.
3. A implementação deverá ser recursiva.

4. A soma e a subtração bit a bit deverão ser implementadas para o funcionamento dos algoritmos.

5. Os exemplos abaixo devem funcionar para os dois algoritmos e devem constar em uma classe com o método main:

   a. 8 x 5

   b. 62938427 x 32984729

   c. 3209094 x 246

   d. 23487 x 294582745

   e. 2345667789968675555342234234245 x 450294459842450836750959083096

   f. 1 x 0

   g. 11101101 x 11110100

   h. 1111111 x 1100

   i. 1100 x 1111110

   j. 11010101110101011101010111010 1 x 11010101010111100000001111110 1