```
P-MATRIX-MULTIPLY-RECURSIVE(A, B, C, n)
 1  if n == 1                                  // just one element in each matrix?
 2      c_11 = c_11 + a_11 · b_11
 3      return
 4  let D be a new n × n matrix                // temporary matrix
 5  parallel for i = 1 to n                    // set D = 0
 6      parallel for j = 1 to n
 7          d_ij = 0
 8  partition A, B, C, and D into n/2 × n/2 submatrices
        A_11, A_12, A_21, A_22; B_11, B_12, B_21, B_22; C_11, C_12, C_21, C_22;
        and D_11, D_12, D_21, D_22; respectively
 9  spawn P-MATRIX-MULTIPLY-RECURSIVE(A_11, B_11, C_11, n/2)
10  spawn P-MATRIX-MULTIPLY-RECURSIVE(A_11, B_12, C_12, n/2)
11  spawn P-MATRIX-MULTIPLY-RECURSIVE(A_21, B_11, C_21, n/2)
12  spawn P-MATRIX-MULTIPLY-RECURSIVE(A_21, B_12, C_22, n/2)
13  spawn P-MATRIX-MULTIPLY-RECURSIVE(A_12, B_21, D_11, n/2)
14  spawn P-MATRIX-MULTIPLY-RECURSIVE(A_12, B_22, D_12, n/2)
15  spawn P-MATRIX-MULTIPLY-RECURSIVE(A_22, B_21, D_21, n/2)
16  spawn P-MATRIX-MULTIPLY-RECURSIVE(A_22, B_22, D_22, n/2)
17  sync                                       // wait for spawned submatrix products
18  parallel for i = 1 to n                    // update C = C + D
19      parallel for j = 1 to n
20          c_ij = c_ij + d_ij
```