efficient power raising

function description : given a base b and natural exponent n it computes b^n.

algorithm idea : instead of 3^{15} requiring 14 products, one computes $3, 3^2, 3^4, 3^8$ successively, each taking one multiplication, and then multiply them all for a total of 6 products.

algorithm (b, n)

    let prod $\leftarrow$ 1, base $\leftarrow b$

    while $n > 0$

        if n odd then prod $\leftarrow$ prod * base

        base $\leftarrow$ base * base, $n \leftarrow \lfloor n/2 \rfloor$

    output prod

algorithm complexity : $O(\log n)$ multiplications.