/\*\*

 \* @param {number} n

 \* @return {number}

 \*/

var nthUglyNumber = function(n) {

    const ugly = [1]; // 丑数序列，第一个丑数是1

    let p2 = 0, p3 = 0, p5 = 0; // 分别指向下一个要乘以2、3、5的丑数位置

    for (let i = 1; i < n; i++) {

        // 计算下一个可能的丑数

        const next2 = ugly[p2] \* 2;

        const next3 = ugly[p3] \* 3;

        const next5 = ugly[p5] \* 5;

        // 选择最小的作为下一个丑数

        const nextUgly = Math.min(next2, next3, next5);

        ugly.push(nextUgly);

        // 移动指针，避免重复计算

        if (nextUgly === next2) p2++;

        if (nextUgly === next3) p3++;

        if (nextUgly === next5) p5++;

    }

    return ugly[n - 1];

};

