Ugly numbers are positive numbers whose prime factors only include 2, 3, 5. For example 1, 2, 3, 4, 5, 6, 8, 9, 10, 12 is the sequence of the first 10 ugly numbers. Note that 1 is typically treated as an ugly number.

Using ad hoc:

For this approach we need to run a loop for all positive integers until ugly number count is equal to n, if an integer is ugly than we have to increase ugly number count. To check if a number is ugly, divide the number by 2, 3 and 5, if any of the remainders is 0 then it is an ugly number otherwise not.

This method is not time efficient as it checks for all integers until ugly number count becomes n, but space complexity of this method is O(1)

Using Dynamic Programming:

The ugly number sequence is 1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 15, ... If we look at the sequence, we can find that every number can only be divided by 2, 3 and 5. So we can split the sequence to three groups as below:

1. 1*2, 2*2, 3*2, 4*2, 5*2, ...
2. 1*3, 2*3, 3*3, 4*3, 5*3, ...
3. 1*5, 2*5, 3*5, 4*5, 5*5, ...

We can find every subsequence is the ugly number itself (1, 2, 3, 4, 5, ..) multiply 2, 3, 5.Then we can use similar merge method as merge sort, to get every ugly number from the three subsequence. Every step we choose the smallest one, and move one step after.