Heap:

Write a program to find the nth super hideous number.
Super hideous numbers are positive numbers whose all prime factors are in the given prime list.

Input:

n=12, primes=[2,7,13,19]

Output:
32

Explanation:[1,2,4,7,8,13,14,16,19,26,28,32] is the sequence of the first 12 super hideous numbers as given primes[2,7,13,19]

Solution:

```
import heapq

class Solution(object):
    def nthSuperHideousNumber(self, n, primes):

        hideous = [1]
        def gen(prime):
            for ugly in hideous:
                yield ugly * prime
        merged = heapq.merge(*map(gen, primes))
        while len(hideous) < n:
            ugly = next(merged)
            if ugly != hideous[-1]:
                hideous.append(ugly)
        return hideous[-1]

n=int(input())
primes=[int(x) for x in input().split()]
r=Solution().nthSuperHideousNumber(n, primes)
print(r)
```