Logo

STUDENT REPORT

DETAILS

Name

A.Padmini

Roll Number

3BR21CS002

EXPERIMENT

Title

NUMBER AND ITS HUGENESS

Description

You are given an integer **N**. You have the starting **N** natural numbers with you arranged in ascending order. It was found that the hugeness of the **i^{th}** number is the bitwise OR of the **$i\text{-}1^{th}$** number and i number. All numbers whose hugeness is a prime number are considered Huge numbers.
Your task is to find and print the maximum hugeness among the Huge Numbers in the array.

Input format:
The input consists of a single lines:
- The first and only line contains an integer **N**.
The input will be read from the STDIN by the candidate.
Output format:
Print the maximum hugeness among the Huge Numbers in the array. If there is no Huge Number then print -1.
The output will be matched to the candidate's output printed on the STDOUT.
Constraints:

- 1 <= N <= 1000

Example:
Input: 8
Output: 7
Explanation: The hugeness of the numbers will be 1, 2,3, 7, 7, 7, 7 and 15. Among which 15 is the maximum but not a prime number whereas 7 is the largest prime number. So, it is the highest hugeness which is a huge number as well.
Sample input: 3
Sample output: 3

Source Code:

```
import math
def is_prime(n):
    if n<=1:
        return False
    for i in range(2,int(math.sqrt(n))+1):
        if n%i==0:
            return False
    return True
def find_huge(num):
    if num==1:
        return -1
    mx=-1
    for i in range(2,num+1):
        huge=(i-1)|i
        if is_prime(huge):
            mx=max(mx,huge)
    return mx if mx!=-1 else -1
num=int(input())
print(find_huge(num))
```

RESULT

4 / 5 Test Cases Passed | 80 %