**Siena College’s 35th Annual High School Programming Contest**

| **Sponsored by** |  |
| --- | --- |

##### **March 31, 2023**

###### Gold Problem #5: Hip and Hippie Numbers

###### 

Background Information: A number is “hip” if it’s not a prime number and its prime decomposition contains no repeated factors which means that it’s square-free. For example, 30, 77, and 561 are hip because their factorizations do not require any prime to be used more than once.

We have 30 = 2 × 3 × 5, 35 = 5 × 7, and 561 = 3 × 11 × 17. Note that you are competing in a hip contest. The first non-hip or square numbers are 4, 8, 9, 12, 16, and 18. Alas, 2023 is not hip because 2023 = 7 × 17 × 17 and 17 × 17 = 289 which is a perfect square.

A number N is a **hippie** number if it’s hip and N - 1 is divisible by one less than each of the prime factors of N. For example, 561 is a hippie number because it’s hip and 560 is divisible by 2, 10, and 16.

Write a program that inputs a range and outputs all of the hippie numbers in this range.

###### Programming Problem:

Input:  Integers A and B with 2 ≤ A ≤ 1,000,000 and A ≤ B ≤ 1,000,000, each on a separate line.

Output: In ascending order, all of the hippie numbers in the given range, inclusive of A and B, one per line. If no hippie numbers are found, print "None".

###### Example : Input: 1000

###### 2000

###### Output:  1105

###### 1729

###### 

###### Example : Input: 4

###### 5

###### Output:  None

###### 

###### 

###### 