ب.م.م

- محدودیت حافظه: ۵۰ مگابایت

ب.م.م: بزرگترین مقسوم علیه مشترک چند عدد بزرگترین عدد طبیعی است که مقسوم علیه همه ی آن اعداد باشد.

ک.م.م: کوچکترین مضرب مشترک چند عدد کوچکترین عدد طبیعی است که مضرب همه ی آن اعداد باشد.

دو عدد نسبت به هم اول هستند اگر ب.م.مشان ۱ باشد.

ورودی

عدد n در یک خط داده می‌شود.

$$2 \leq n \leq 20$$

خروجی

ک.م.م اعدادی که از n کمتر اند و نسبت به آن اول هستند.

مثال

ورودی نمونه ۱

```
5
```

خروجی نمونه ۱

راه‌حل

ب.م.م دو عدد a و b را با $gcd(a,b)$ و ک.م.م‌شان را با $lcm(a,b)$ نشان می‌دهیم.

تابعی می‌نویسیم برای گرفتن ب.م.م بین دو عدد را بگیرد، برای این کار دو روش را می‌نویسیم:

```
FUNCTION >>> gcd(a , b)
greatest common divisor of numbers 'a' and 'b'
----------------------------------------------
1. for i:1 to a + b do
2.     x <- a + b - i + 1
3.     if a mod x = 0 AND b mod x = 0 then
4.         return x
----------------------------------------------
1. if a > b then
2.     swap(a , b)
3. while a > 0 do
4.     b <- b mod a
5.     swap(a , b)
6. return b
```

روش اول طبق تعریف ب.م.م درست است و از نظر زمانی به اندازه ی $a+b$ واحد زمانی طول می‌کشد. در روش دوم از این نکته استفاده شده است که $gcd(a,b) = gcd(a-b,b)$ بنابراین داریم: $gcd(a,b) = gcd(a \; mod \; b, b)$

حالا می‌خواهیم الگوریتم دوم را تحلیل زمانی کنیم. اتفاقی که هر بار در حلقه می‌افتد این است که a برابر باقیماندهٔ تقسیم b بر a می‌شود و مقدار b برابر a می‌شود، این همان الگوریتم نردبانی یا اقلیدسی است! در واقع هر سری که عددی مثل x برابر باقیمانده‌اش بر y می‌شود، به شرطی که y کمتر از x باشد، مقدار x حداقل نصف می‌شود! چرا که یا $y \leq \frac{x}{2}$ که در اینصورت چون در هر تقسیمی باقیمانده کمتر از مقسوم علیه

است مقدار x حداقل نصف می‌شود. در غیر اینصورت نیز x تبدیل به $x - y$ می‌شود که چون y از نصف x بیشتر فرض شده است، باز هم حکم برقرار است.

پس می‌توان نتیجه گرفت پس از دو مرحله $a + b$ حداقل نصف می‌شود! یعنی الگوریتم حدوداً $log_2(a + b)$ واحد زمانی طول می‌کشد! پس بهتر است از مدل دوم استفاده کنیم. اگر در مورد الگوریتم نردبانی اقلیدسی اطلاعات بیشتری می‌خواهید، به این لینک مراجعه کنید.

از طرفی ک.م.م دو عدد a و b برابر است با $lcm(a, b) = \frac{a \times b}{gcd(a,b)}$

```
FUNCTION >>> lcm(a , b)
least common multiplier of 'a' and 'b'
--------------------------------------
1. return a * b / gcd(a , b)
```

شبه کد مسأله:

```
Output lcm of all the numbers smaller than 'n'
that are co-prime with 'n'
----------------------------------------------
1. answer <- 1
2. for i:1 to n-1
3.     if gcd(i , n) = 1 then
4.         answer <- lcm(answer , i)
5. output answer
```