SAMSUNG 삼성SDS

SW 아카데미 알고리즘 특강

Contents

오늘의 원리

- 복잡한 알고리즘 문제를 상황에 따른 여러 경우로 나누고, 논리적으로 문제를 풀어보는 과정을 연습한다.
- 유클리드 호제법은 빠른 시간에 두 수의 최대공약수를 구할 수 있는 방법이다.
- 확장 유클리드 호제법을 이용하여 특정 방정식을 만족하는 해를 구할 수 있다.
- 1보다 큰 정수의 양의 약수가 1과 자기 자신뿐일 때 그 수를 소수(Prime number)라 하며 소수를 계산하는 대표적인 방법은 에라토스테네스의 체를 사용하는 방법이다.
- 서로소, 인수분해, 최소공배수와 최대공약수 등의 문제를 소수를 활용하여 계산할 수 있다.

학습목표

- **합동식의 성질을 이해**하고 증명할 수 있다.
- **유클리드 호제법과 확장 유클리드 호제법을 이해**하고 응용하여 문제를 해결할 수 있다.
- **소수 및 소수와 관련된 수의 특징을 이해**하고 관련 문제를 해결할 수 있다.

사전문제

1. 홀수의 제곱을 8로 나눈 나머지는 항상 1임을 증명하시오.

2. 2이상의 정수 N에 대하여, $N^3 - N$ 은 항상 6으로 나눠짐을 증명하시오.

3. 짝수인 소수는 정확히 한 개만 존재함을 증명하시오.

사전문제 접근법

(1) 임의의 홀수를 표현하는 방법에 대해 생각해 본다.

(2) 연속된 N개의 숫자들의 곱이 갖는 특징에 대해 생각해 본다.

(3) 그 명제의 부정(否定)을 참이라고 가정할 때 나타나는 모순을 증명함으로써 원래의 명제가 참인 것을 보여 주는 "귀류법"을 이용해 본다.

사전문제

(1) 모든 홀수 n은 $n = 2k + 1,\ k \in \{0, 1, \dots\}$의 꼴로 나타낼 수 있다.
따라서 $n^2 = (2k+1)^2 = 4k^2 + 4k + 1 = 4k(k+1) + 1$을 만족한다.
또한 k의 값에 관계없이 k와 $k+1$의 곱은 짝수이므로 $n^2 \equiv 8m + 1 \equiv 1 (mod\ 8)$ 가 된다.

(2) $N^3 - N = N(N^2 - 1) = (N-1)N(N+1)$ 로 나타낼 수 있다.
다시 말해, 연속된 세 수의 곱으로 표현되므로 $N^3 - N$은 6의 배수임을 알 수 있다.

(3) 명제를 부정하여 짝수인 소수가 2개 이상, 즉 최소 2개라고 가정해보자.
짝수인 소수 중 하나는 2임을 알고 있으므로 2보다 큰 짝수이면서 소수인 수 P가 존재한다.
하지만 P는 짝수이기 때문에 2의 배수이고, 이는 P가 소수라는 가정에 모순이 된다.
따라서 짝수인 소수는 정확히 하나뿐임을 알 수 있다.

합동식(Congruences)

- **항등식과 합동식**

항등식은 등호(=)로 그 관계를 나타내지만, 합동식은 합동기호(≡)로 관계를 나타낸다.
합동식 $a \equiv b(mod\ p)$은 a를 p로 나눈 나머지가 b를 p로 나눈 나머지와 같다는 뜻이다.

(1) 항등식과 합동식의 공통점

- $a \equiv a\ (mod\ p)$
- $a \equiv b\ (mod\ p) \Rightarrow b \equiv a\ (mod\ p)$
- $a \equiv b\ (mod\ p),\ \ b \equiv c\ (mod\ p) \Rightarrow a \equiv c\ (mod\ p)$
- $a \equiv b\ (mod\ p) \Rightarrow a \times c \equiv b \times c\ (mod\ p)$ (c는 임의의 정수)
- $a \equiv b\ (mod\ p) \Rightarrow a \pm c \equiv b \pm c\ (mod\ p)$ (c는 임의의 정수)

(2) 항등식과 합동식의 차이점

- $a \times c \equiv b \times c\ (mod\ p)$를 만족하더라도 $a \equiv b\ (mod\ p)$ 를 보장하지 않는다.
- $a \times c \equiv b \times c\ (mod\ p)$를 만족하면 $a \equiv b\ (mod\ \frac{p}{\gcd(c,p)})$ 를 보장한다.

예) 8을 6으로 나눈 나머지는 2이고, 14을 6으로 나눈 나머지는 같다. $(8 \equiv 14(mod\ 6))$
4를 3으로 나눈 나머지와 7을 3으로 나눈 나머지는 같다. $\left(4 \equiv 7\ (mod\ 3)\right)$
4를 6으로 나눈 나머지와 7을 6으로 나눈 나머지는 같지않다. $(4 \not\equiv 7\ (mod\ 6))$

연습문제

(1) a를 p로 나눈 나머지와 b를 p로 나눈 나머지가 같으면, 임의의 정수 c에 대해 $a+c$를 p로 나눈 나머지와 $b+c$를 p로 나눈 나머지가 같음을 증명하시오.

(2) a를 p로 나눈 나머지와 b를 p로 나눈 나머지가 같으면, 임의의 정수 c에 대해 $a \times c$를 p로 나눈 나머지와 $b \times c$를 p로 나눈 나머지가 같음을 증명하시오.

(3) $a \times c$를 p로 나눈 나머지와 $b \times c$를 p로 나눈 나머지가 같아도 a와 b를 p로 나눈 나머지가 같지 않은 반례를 찾으시오.

연습문제 접근법

- a를 p로 나눈 나머지가 r_1, b를 p로 나눈 나머지가 r_2인 경우
  $a = q_1 \times p + r_1$, $b = q_2 \times p + r_2$ 꼴로 나타낼 수 있다.

연습문제

(1) $a = q_1 \times p + r,\ b = q_2 \times p + r$ 라고 하면

$a + c = q_1 \times p + (r + c),\ b + c = q_2 \times p + (r + c)$ 임을 알 수 있다. 따라서

$a + c \equiv q_1 \times p + (r + c) \equiv q_2 \times p + (r + c) \equiv b + c (mod\ p)$

(2) $a = q_1 \times p + r,\ b = q_2 \times p + r$ 라고 하면

$a \times c = q_1 \times p \times c + r \times c,\ b \times c = q_2 \times p \times c + r \times c$ 임을 알 수 있다. 따라서

$a \times c \equiv q_1 \times c \times p + r \times c \equiv q_2 \times c \times p + r \times c \equiv b + c (mod\ p)$

(3) $a = 2,\ b = 1,\ c = 6,\ p = 3$ 등등..

연습문제

1. 5이상의 소수 P는 $P = \sqrt{24N + 1}$ 의 꼴로 표현될 수 있음을 증명하시오.

2. 모든 자연수 N에 대하여 N^2을 3으로 나눈 나머지는 항상 0또는 1임을 증명하시오.

3. 음수가 아닌 정수 N에 대하여 N과 $N^2 + 2$가 소수일 때, $N^3 + 2$도 소수임을 증명하시오.

연습문제 접근법

(1) $P^2 - 1 = 24N$의 형태로 변형하여 생각해본다.

(2) N을 3으로 나누었을 때 나올 수 있는 나머지의 경우에 대해 생각해 본다.

(3) N을 3으로 나누었을 때 나올 수 있는 나머지의 경우에 대해 생각해 본다.

연습문제

(1) P를 임의의 소수라고 가정하면 $P^2-1=(P-1)(P+1)$이 24의 배수임을 보이면 된다.
또한 P는 5보다 큰 소수이기 때문에 홀수이고, 3으로 나누었을 때 나머지가 1 또는 2이다.
따라서 위 두 가지 경우에 대해 각각 살펴보면 P^2-1은 24배수임을 쉽게 알 수 있다.

(2) N이 3의 배수인 경우 N^2을 3으로 나눈 경우 나머지가 0이다.
$N=3k+1$인 경우 나머지가 1이고, $N=3k+2$인 경우 나머지도 1이다.

(3) $N=3k\pm1$인 경우 N이 1인 경우를 제외하면 N^2+2가 모두 3의 배수이다.
$N=3k$인 경우 N이 0인 경우를 제외하면 N이 모두 3의 배수이다.

사전 문제

1. 마을의 항구에는 총 3대의 배가 있고 배들이 한번 항구를 떠난 후 다시 돌아오는데 까지 걸리는 시간은 각각 2일 5일 7일이다. 모든 배는 들어온 날 다시 출항한다. 첫째 날 동시에 모든 배가 항구를 떠나면 다시 3대의 배가 동시에 항구로 들어오는 날은 언제인지 구하시오

사전 문제

1. 주기가 2일인 배는 1일, 3일, 5일 항구로 돌아오고, 5일인 배는 1일, 6일, 11일에 7일인 배는 1일, 8일, 15일에 돌아오게 된다.

사전 문제

2로 나눈 나머지가 1인 날 첫 번째 배가 항구에 있고,
5으로 나눈 나머지가 1인 날 두 번째 배가, 7로 나눈 나머지가 1인 날 세 번째 배가 항구에 있다.

따라서 2, 5, 7의 최소공배수인 70일이 배가 항구로 들어오는 주기가 되며,
처음으로 모든 배가 항구로 들어오는 날은 71일이 된다.

유클리드 호제법

유클리드 호제법은 2개의 자연수의 최대공약수를 구하는 알고리즘의 하나이다. 2개의 자연수 a, b에 대해서 **a를 b로 나눈 나머지를 r이라 하면(단, a>b), a와 b의 최대공약수는 b와 r의 최대공약수와 같다**는 성질을 이용한다.

☞ 유클리드 호제법을 증명해보아라.

연습문제

1. 기약분수란, 분수로 표현된 분자와 분모가 서로 소 즉, 1 이외의 공통된 약수로 더 이상 나누어 떨어지지 않는 형태를 말한다. 임의의 정수 A, B가 주어질 때, A/B 가 기약 분수의 형태가 되도록 하는 코드를 짜보자.

2. 세 개 이상의 정수의 최대공약수를 구하는 방법을 생각해보아라.

연습문제

1. 임의의 정수 A, B에 대해 분수 A/B가 기약 분수의 형태가 되게 하려면, 정의에 따라 a와 b 는 서로소가 되어야 한다. 이를 위해 a와 b의 공통 약수들을 모두 나누어, 공통된 약수가 1이 되도록 하여야 한다. 공통 약수 중의 가장 큰 최대공약수로 a 와 b 를 나눈 각각의 숫자가 a/b가 기약분수가 되게 하는 숫자일 것이다.

연습문제

1. a = 36, b = 24

유클리드 호제법을 사용하여 a와 b의 최대공약수를 쉽게 구할 수 있다.

gcd(a, b) = gcd(b, a%b) (단, a>=b) 이므로,

gcd(36, 24)

= gcd(24, 12)

= gcd(12, 0)

즉, 최대 공약수로 12가 구해지므로, 36/24의 기약 분수 형태는 각각의 수를 12 로 나눈 3/2 로 나타낼 수 있다.

연습문제

2. 세 개 이상의 정수의 최대공약수를 구하는 방법으로 소인수분해가 있다.

모든 정수는 소수의 곱으로 표현되는 방법이 유일하게 주어진다.
주어진 정수들을 각각 소수의 곱으로 표현한 후, (후에 나올 에라토스테네스의 체를 참고)
각 정수마다 공통으로 곱해진 소수 중 지수가 가장 최소인 것을 곱한 결과가 최대공약수이다.
예를 들어, 132, 36, 24, 72 의 최대공약수를 소인수분해로 구하면 아래와 같다.

정수	소수들의 곱	2의 곱 횟수	3의 곱 횟수	11의 곱 횟수
132	2^2*3*11	2	1	1
36	2^2*3^2	2	2	0
24	2^3*3	3	1	0
72	2^3*3^2	3	2	0
최대공약수	2^2*3	2	1	0

연습문제

2. 다른 방법으로는 유클리드 호제법을 재귀적으로 사용하는 방법이 있을 것이다.

gcd(A,B,C) = gcd(gcd(A,B),C) 를 증명해보자.

두 개의 정수 A, B가 있을 때 gcd(A,B) 에 대해 생각해보자.

1. gcd(A,B) 는 A 와 B 의 약수이다.
2. A와 B 의 약수는 gcd(A,B) 의 약수이다.

위의 공리를 gcd(A,B,C)에 적용해보면, gcd(A, B, C) = G 는 A의 약수, B의 약수, C의 약수이다.
A의 약수이면서 B의 약수인 숫자는 gcd(A,B)의 약수이므로, G는 gcd(A,B)의 약수이면서 C의 약수이므로 G는 gcd(gcd(A,B),C) 의 약수 임을 알 수 있다.
반대로 gcd(gcd(A,B), C) = G′ 는 gcd(A,B) 의 약수이면서 C의 약수이다. gcd(A,B) 는 A와 B의 약수이므로, G′는 gcd(A,B) 의 약수이면서 C의 약수, 즉, A의 약수, B의 약수, C의 약수이다. 이는 또한 위의 논리에 따라 G′ 는 G의 약수임을 알 수 있다.

따라서 G 는 G′ 의 약수이면서 동시에 G′ 는 G 의 약수이므로 G와 G′ 는 서로 같은 수임이 증명이 된다.

- **부정 방정식**
  해가 무수히 많은 방정식.
  미지항의 개수보다 방정식의 개수가 작을 경우도 부정방정식에 포함된다.

- **디오판토스 방정식**
  해가 정수인 경우의 부정 방정식 (3x + 2y = 5)

- **베주 항등식**

정수 x, y 의 최대공약수를 gcd(x,y)로 나타낼 때, 확장 유클리드 호제법을 이용하여, **ax + by = gcd(x,y)의 해가 되는 정수 a, b 짝을 찾아낼 수 있다.**(a, b 중 한개는 보통 음수가 된다.) 이 식을 베주의 항등식이라고 한다.

위에서 든 예의 경우, 특히, x,y가 서로소(gcd(x,y) = 1)인 경우 유용한데, 그럼 위의 식은 ax + by = 1이 되고, 여기서 a는 모듈로 연산의 곱의 역원(modular multiplicative inverse)이 되기 때문이다.

확장 유클리드 호제법

ax+by=c 형태의 방정식이 주어질 때 이 방정식을 만족 시키는 정수 x, y를 각각 s, t라고 정의하고, 우변의 상수값에 유클리드 호제법(r″-qr′=r)을 사용하여 반복적으로 구하는 값들을 r이라고 하면 아래 방정식을 만족시키는 정수 해 s와 t가 반드시 존재한다.

. ax+by=a . ax+by=b

여기서 부터 아래 단계를 반복적으로 적용하면서 r값을 줄여나가면 최종적으로 우변의 값을 만들 수 있는 r값을 찾을 수 있고, 이 때 s와 t값을 통해 해를 구할 수 있다.

① r = r″-qr′ 식을 통해 q와 r을 계산한다.

① s = s″-qs′, t = t″-qt′ 식을 통해 s와 t를 계산한다.

. 9x+5y=1 → x=-1, y=2, 9x+5y=2 → x=-2, y=4,

s	t	r	q
1	0	9	
0	1	5	
1	-1	4	1
-1	2	1	1

2단계

1단계

확장 유클리드 호제법

식 9x+5y=2의 해를 찾는 과정을 통해 확인해 보면 r 값을 유클리드 호제법으로 아래 표와 같이 반복적으로 계산할 수 있으며 가장 작은 r은 두 수의 최대공약수가 됨을 알 수 있다.

s	t	r	q
1	0	9	
0	1	5	
		4	**1**
		1	**1**

여기서 r 값은 아래와 같은 관계를 가진다.

$$r = r'' - qr'$$
$$r = s''a + t''b - q(s'a + t'b)$$
$$r = (s''-qs')a + (t'' - qt')b$$

따라서 새로운 r을 만드는 s, t의 정수해가 반드시 존재하며 이는 각각 아래와 같이 계산할 수 있다.

$$s = s'' - qs'$$
$$t = t'' - qt'$$

s	t	r	q
1	0	9	
0	1	5	
1	**-1**	4	1
-1	**2**	1	1

1. 다음 방정식의 정수해를 구하시오

. 5x+9y=2

. 4x+22y=2

. 61x+67y=12

. 213x+720y=3

. 324x+84y=38

1. 좌변과 우변의 공약수가 있는 경우 약분하고, 우변의 값이 gcd(a, b)의 배수인 경우 gcd(a, b)로 변경한 후 해를 구한다.

. 5x+9y=2 → 5x+9y=1
. 4x+22y=2 → 2x+11y=1
. 61x+67y=12 → 61x+67y=1
. 213x+720y=3 → 71x+240y=1
. 324x+84y=38 → 162x+42y=19 → 19가 gcd(162, 42)=6의 배수가 아님

연습문제

. 9x+5y=1 → x=-1, y=2, 9x+5y=2 → x=-2, y=4,

s	t	r	q
1	0	9	
0	1	5	
1	-1	4	1
-1	2	1	1

연습문제

. 2x+11y=1 → x=-5, y=1, 4x+22y=2 → x=-5, y=1

s	t	r	q
1	0	2	
0	1	11	
1	0	2	0
-5	1	1	5

. 61x+67y=1 → x=11, y=-10, 61x+67y=12 → x=132, y=-120

s	t	r	q
1	0	61	
0	1	67	
1	0	61	0
-1	1	6	1
11	-10	1	10

연습문제

. 71x+240y=1 → x=71, y=-21, 213x+720y=3 → x=71, y=-21

s	t	r	q
1	0	71	
0	1	240	
1	0	71	0
-3	1	27	3
7	-2	17	2
-10	3	10	1
17	-5	7	1
-27	8	3	1
71	-21	1	2

연습문제

. 162x+42y=19 → 해가 없음

s	t	r	q
1	0	162	
0	1	42	
1	-3	36	3
-1	4	6	1
7	-27	0	6

※ 우변이 a와 b의 최대공약수의 배수가 아닌 경우 방정식을 만족하는 정수해가 존재하지 않는다.

사전문제

1. 1742년, 독일의 아마추어 수학가 크리스티안 골드바흐는 레온하르트 오일러에게 다음과 같은 추측을 제안하는 편지를 보냈다.

   "4보다 큰 모든 짝수는 두 홀수 소수의 합으로 나타낼 수 있다."

   예를 들어 8은 3 + 5로 나타낼 수 있고, 3과 5는 모두 홀수인 소수이다.

   또, 20 = 3 + 17 = 7 + 13, 42 = 5 + 37 = 11 + 31 = 13 + 29 = 19 + 23 이다.

   이 추측은 아직도 해결되지 않은 문제이다.

   백만 이하의 모든 짝수에 대해서만 이 추측을 검증해 보자

2. 하나 이상의 연속된 소수의 합으로 나타낼 수 있는 자연수들이 있다.

   2 이상의 자연수가 주어졌을 때, 이 자연수를 연속된 소수의 합으로 나타낼 수 있는 경우의 수를 구하시오.

   •3 : 3 (한 가지)

   •41 : 2+3+5+7+11+13 = 11+13+17 = 41 (세 가지)

   •20 : 7+13 (11이 빠져 연속된 소수의 합이 아니기 때문에 만들 수 없는 경우)

사전문제

1. 백만 이하의 모든 소수들을 찾을 수 있어야 한다.

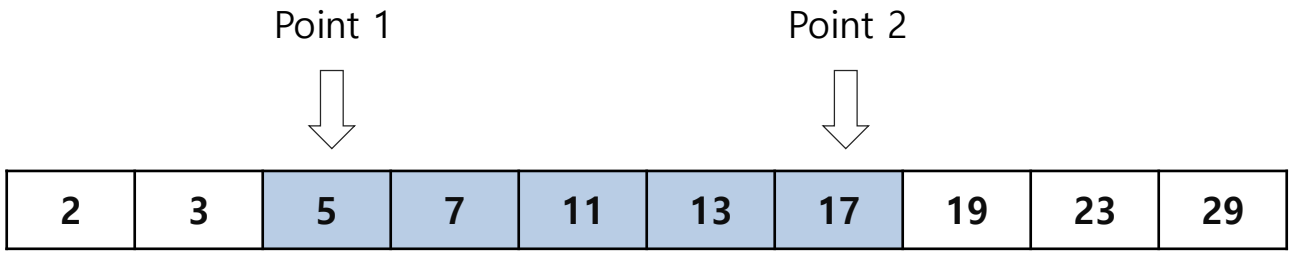
2. 연속된 소수의 합을 빠르게 계산할 수 있어야 한다.

사전문제

1. 백만 이하의 모든 소수의 목록은 에라토스테네스의 체를 사용하여 구할 수 있으며, 소수를 구하는 자세한 내용은 본 학습에서 다루도록 한다.

2. 계산되어 있는 소수목록에서 연속된 소수의 합이 주어진 수 N이 되는 경우를 찾아야 한다.

   2 pointers 기법을 활용할 수 있다.



주어진 소수 목록에서 구간의 합이 N보다 작으면 Point2를 증가시키고 N보다 크면 Point1을 증가시키면서 합이 N이 되는 경우 경우의 수를 증가시켜준다.

소수의 정의와 활용

1보다 큰 정수 p의 양의 약수가 1과 p뿐일 때 p를 소수(Prime number)라 하며 1보다 큰 정수 n이 소수가 아닐 때 n을 합성수(Composite number)라 한다.

- **소수의 분포**
  어떤 수 N이 주어질 때 N보다 작은 소수의 개수는 $\pi(N)$으로 표현한다.
  $$\pi(N) \approx N/\ln N$$
- **서로소**
  두 수가 1 이외의 양의 공약수를 가지지 않는 경우 서로소라고 한다.
  어떤 수 N이 주어질 때 N보다 작은 N과 서로소인 수들의 개수를 $\varphi(N)$으로 표현한다.
  $$\varphi(N) = N \times \prod_{pf}(1 - \frac{1}{pf}) \qquad (pf = prime\ factors\ of\ N)$$
  φ 함수는 아래와 같은 특징을 가진다.
  $$\varphi(ab) = \varphi(a)\varphi(b) \quad (a와 b는 서로소)$$
  $$\varphi(p^m) = p^m\left(1 - \frac{1}{p}\right) \qquad (p는 소수)$$
- **인수분해**
  1보다 큰 정수 N의 서로 다른 소인수 $p_1, \ldots, p_k$가 있을 때 N은 단 한가지 방법으로 다음과 같이 소인수 분해되며 이를 표준분해라고 한다.
  $$N = {p_1}^{e_1} \ldots {p_k}^{e_k}$$

소수의 정의와 활용

1. 숫자 N이 주어졌을 때 $k(k \in \{2 \cdots N-1\})$로 나누어 보면 소수인지 $O(N)$ 시간에 판단할 수 있다.더 빠르게 소수를 판단할 수 있는 방법들을 구하시오.

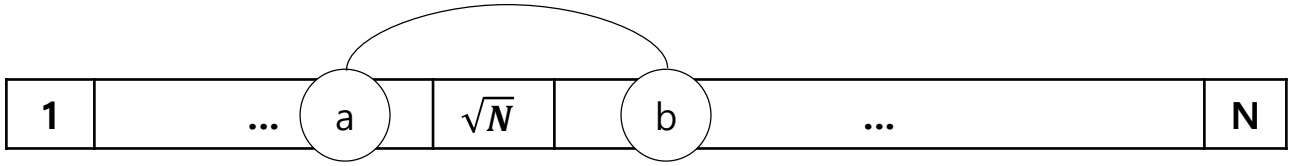
2. N개의 물질이 있고 각 물질마다의 특성을 숫자 1부터 N까지 정의하였다. 만약 5개의 물질이 있는 경우 물질의 특성은 1,2,3,4,5가 된다. N개의 물질 중 임의로 두 개를 선택하였을 때 두 물질 특성의 최대공약수가 1이 되면 두 물질을 합성할 수 있다고 하자. 물질의 개수 N이 주어질 때 합성할 수 있는 경우의 수는 몇 가지가 되는지 계산하시오.

소수판별

N보다 작은 모든 숫자들로 나누어보는 것은 비효율 적이다.

좀 더 효율적인 방법은 $k(k \in \{2 \cdots \sqrt{N}\})$로 나누어 보는 것이다. N이 소수가 아닌 경우 $N = a \times b$로 나타낼 수 있고, a가 b보다 작은 경우 N이 b로 나누어지기 전에 a에 의해 먼저 나누어 질 것이다.

이런 방법으로 수행하는 경우 $O(\sqrt{N})$의 시간복잡도를 가지게 된다.



추가적으로 2를 제외한 짝수의 경우도 나누어 보는 대상에서 제외할 수 있다는 것을 알 수 있다. 그러면 위 방법보다 2배 빠른 $\sqrt{N}/2$의 수행횟수를 가지지만 이것 역시 $O(\sqrt{N})$의 시간복잡도를 가지게 된다.

더 빠르게 알고리즘을 개선하는 방법을 찾아보자.

연습문제

가장 빠르게 소수를 판별하는 방법은 $\sqrt{N}$보다 작은 소수들로만 나눠보는 것이다. 소수가 아닌 합성수로 N이 나누어지는 경우 이미 그 수보다 더 작은 소수가 먼저 N을 나눌 것이다. 이 때 수행횟수는 $\pi(\sqrt{N})$이므로 시간복잡도는 아래와 같다.

$$O(\sqrt{N}/\ln(\sqrt{N}))$$

추가로 하나의 숫자가 아닌 N보다 작은 모든 소수의 목록을 찾는 경우 에라토스테네스의 체를 이용할 수 있는데 아이디어는 위의 풀이법과 유사하다.

소수를 찾으면 해당하는 소수의 배수들을 모두 지워줌으로써 지워지지 않은 숫자를 찾는 것이다.

물질 합성

두 수를 선택하였을 때 최대공약수가 1이라는 말은 두 수가 서로소라는 말이다. 아래와 같이 N이 10일 때 6을 선택하고 6보다 작은 수들 중 서로소인 숫자를 고르면 1,5 2가지 경우를 찾을 수 있다. 이런 방법으로 i번째보다 작고 i와 서로소인 숫자의 개수를 $i \in \{1, \cdots, 10\}$에 대해 구하여 모두 더하면 문제의 답을 계산할 수 있다.

1	2	3	4	5	6	7	8	9	10

단, 여기서 6과 서로소인 숫자를 찾을 때 6이하의 숫자들과 gcd를 구하여 하나씩 비교하면 유클리드 호제법이 상수시간을 가진다고 가정하여도 $O(N^2)$의 시간복잡도를 가지게 된다.

더 빠르게 서로소를 구할 수 있는 방법을 생각해보자.

연습문제

우리는 아래 오일러피 함수를 사용하여 서로소를 더 빠르게 구할 수 있다.

$$\varphi(N) = N \times \prod_{pf}(1 - \frac{1}{pf}) \qquad (pf = prime\ factors\ of\ N)$$

N이 6일 때 서로소의 개수는 위 식에 의해 2가 됨을 알 수 있다.

$$\varphi(6) = 6 \times (1 - \frac{1}{2}) \times (1 - \frac{1}{3}) = 2$$

결국 오일러피를 사용하면 하나의 숫자에 대해 $\pi(N)$번 연산이 일어나기 때문에 전체 시간복잡도는 아래와 같이 될 것이다.

$$O(N^2 / ln\, N)$$

※ 오일러피 함수를 증명해보자.

문제. 암호제작 (NCPC 2005 D번)

원룡이는 한 컴퓨터 보안 회사에서 일을 하고 있다. 그러던 도중, 원룡이는 YESWOA.COM 으로부터 홈페이지 유저들의 비밀키를 만들라는 지시를 받았다. 원룡이는 비밀 키를 다음과 같은 방법으로 만들었다.

개인마다 어떤 특정한 소수 p와 q를 주어 두 소수의 곱 pq를 비밀 키로 두었다. 이렇게 해 주면 두 소수 p,q를 알지 못하는 이상, 비밀 키를 알 수 없다는 장점을 가지고 있다.

하지만 원룡이는 한 가지 사실을 잊고 말았다. 최근 컴퓨터 기술이 발달함에 따라, 소수가 작은 경우에는 컴퓨터로 모든 경우의 수를 돌려보아 비밀 키를 쉽게 알 수 있다는 것이다.

원룡이는 주성조교님께 비밀 키를 제출하려던 바로 직전에 이 사실을 알아냈다. 그래서 두 소수 p, q 중 하나라도 K보다 크지 않은 경우 제출하지 않기로 하였다. 이것을 손으로 직접 구해보는 일은 매우 힘들 것이다. 당신은 원룡이를 도와 두 소수의 곱으로 이루어진 암호와 K가 주어져 있을 때, 그 암호가 좋은 암호인지 좋지 않은 암호인지 구하는 프로그램을 작성하여야 한다.

[입력]
암호 P($4 \le P \le 10^{100}$)와 K($2 \le K \le 10^{6}$) 이 주어진다.

[출력]
만약에 그 암호가 좋은 암호이면 첫째 줄에 GOOD을 출력하고, 만약에 좋지 않은 암호이면 BAD와 소수 p를 출력하는데 p는 암호를 이루는 두 소수 중 작은 소수를 의미한다.

실전문제 접근법

1. 문제에서 주어지는 암호 P는 굉장히 크기 때문에 P를 인수분해 할 수는 없다. 따라서 K값을 기준으로 해결방법을 찾아야 한다.

실전문제

1. 암호 P가 굉장히 크기 때문에 인수분해를 하여 해결할 수 없다.

2. 주어지는 수 K보다 작은 소수를 인수로 가지는 지만 확인해 보면 되기 때문에 K보다 작은 소수들로 P를 나누어 나머지가 0이 되는 숫자가 있는지 확인한다.

   ※ K보다 작은 소수들은 $\pi(K)$개가 있기 때문에 K보다 작은 모든 소수들에 대해 $O(K/\ln(K))$ 시간에 검토할 수 있다.

3. 마지막으로 P를 임의의 소수 k로 나눈 나머지를 구할 수 있어야 한다.
   주어지는 101자리 암호 P에 대해 K보다 작은 소수 p로 나누는 경우 P를 문자열로 두고 각각의 자리에 대해 분리하여 접근할 수 있다.

$$
\begin{aligned}
& a1 \times 10^{100} \pmod p \\
+\ & a2 \times 10^{99} \pmod p \\
+\ & a3 \times 10^{98} \pmod p \\
& \cdots
\end{aligned}
\quad \Longrightarrow \quad
\begin{aligned}
& ① \ a1 \pmod p \\
& ② \ (① \times 10) + a2 \pmod p \\
& ③ \ (② \times 10) + a3 \pmod p
\end{aligned}
$$

Summary

- 합동식의 성질을 이해하고 계산에 응용할 수 있다.
- 두 양의 정수가 주어졌을 때 유클리드호제법을 사용하여 최대공약수, 최소공배수를 계산할 수 있고, a/b 형태의 분수를 기약분수로 만들 수 있다.
- 확장 유클리드 호제법을 사용하여 ax + by = gcd(a,b) 방정식을 만족하는 x, y를 구할 수 있다.
- 에라토스테네스의 체를 이용하여 양의 정수 n에 대해 소수 여부를 판단하거나 n보다 작은 소수의 목록을 만들 수 있다.
- 오일러피를 사용하여 양의 정수 n보다 작은 양의 정수들 중 n과 서로 소인 수들의 개수를 빠르게 계산한다.