Algorithm Homework 11

Daoyu Wang

November 23

Contents

1 Alarm

In order to **save space** and **simplify expression**, we use (a, b) instead of $gcd(a, b)$.

2 31.1-10

To show the **gcd** opration is independtent of the order of its argument, we prove the following swap property, for all a, b, c, $(a,(b,c)) = ((a,b),c)$.

2.1 Lemma

Let a_i be the power of the ith prime in the prime factorization of a. Similarly as b_i and c_i.

Then, we have that:

$$\begin{aligned} a &= \prod_{i=1}^{\infty} p_i^{a_i} \\ b &= \prod_{i=1}^{\infty} p_i^{b_i} \\ c &= \prod_{i=1}^{\infty} p_i^{c_i} \end{aligned} \tag{1}$$

2.2 Prove

$$\begin{aligned} (a,(b,c)) &= \prod_{i=1}^{\infty} p_i^{min(a_i,min(b_i,c_i))} \\ &= \prod_{i=1}^{\infty} p_i^{min(a_i,b_i,c_i)} \\ &= \prod_{i=1}^{\infty} p_i^{min(min(a_i,b_i),c_i)} \\ &= ((a,b),c) \end{aligned} \tag{2}$$

So, the **gcd** opration is independtent of the order of its argument.

3 31.2-5

3.1 Lemmas

3.1.1 Fibonacci sequence

We acknowledge that:

$$\begin{aligned} F_k &= \frac{1}{\sqrt{5}}((\frac{\sqrt{5}+1}{2})^k - (1-\frac{\sqrt{5}}{2})^k) \\ &= \frac{1}{\sqrt{5}}(\Phi^k - (-\Phi^{-1})^k) \end{aligned} \tag{3}$$

And $\Phi = \frac{\sqrt{5}+1}{2}$.

3.1.2 Thereom 31.10

We acknowledge that: If $a > b \geq 0$ and $EUCLID(a, b)$ performs k recursive calls, then

$$\begin{aligned} b &\geq F_{k+1} \\ &= \frac{1}{\sqrt{5}}(\Phi^{k+1} - (-\Phi^{-1})^{k+1}) \end{aligned} \tag{4}$$

This equation can be expressed in another way as follows:

$$\sqrt{5}b \geq \Phi^{k+1} + \frac{(-1)^k}{\Phi^{k+1}} \tag{5}$$

Let $f(x) = x^2 - \sqrt{5}bx + (-1)^k$, we have:

$$f(\Phi^{k+1}) \leq 0 \tag{6}$$

3.1.3 Basic conclusions of number theory

There are two basic conclusions:

$$(\frac{a}{(a,b)}, \frac{b}{(a,b)}) = 1 \tag{7}$$

This is because (a, b) is the gcd of a and b.

$$\frac{a}{(a,b)} \mod \frac{b}{(a,b)} = \frac{a \mod b}{(a,b)} \tag{8}$$

Assume that $\frac{a}{(a,b)} = k \cdot \frac{b}{(a,b)} + r$, we know $r < \frac{b}{(a,b)}$.

So we have $a = kb + r\cdot(a, b)$. And because $r\cdot(a, b) < b$, $a \mod b = r\cdot(a, b)$. So, $r = \frac{a \mod b}{(a,b)}$.

3.2 Prove

Assume that $a > b \geq 0$ and $EUCLID(a, b)$ performs k recursive calls.

3.2.1 $k \leq 1 + \log_\Phi(b)$

To prove that $k \leq 1 + \log_\Phi(b)$, we can just prove that $\Phi^{k+1} \leq \Phi^2 b$. Consider function $f(x) = x^2 - \sqrt{5}bx + (-1)^k$ defined above is a quadratic function that opens upward and $f(\Phi^{k+1}) \leq 0$, we have:

$$\begin{aligned} f(\Phi^2 b) &= \Phi^4 b^2 - \sqrt{5}b \cdot \Phi^2 b + (-1)^k \\ &= \Phi^2(\Phi^2 - \sqrt{5})b^2 + (-1)^k \\ &= \frac{3+\sqrt{5}}{2} \cdot \frac{3-\sqrt{5}}{2} b^2 + (-1)^k \\ &= b^2 + (-1)^k \geq 1 - 1 \geq 0 \end{aligned} \tag{9}$$

And $\Phi^2 b = \frac{3+\sqrt{5}}{2} b > \frac{\sqrt{5}}{2} b$, which is the midline of $f(x)$. So, we have $\Phi^{k+1} \leq \Phi^2 b$ which represents that $k \leq 1 + \log_\Phi(b)$.

3.2.2 $k \leq 1 + \log_\Phi(\frac{b}{(a,b)})$

We acknowledge that for $a > b \geq 0$, $EUCLID(a, b)$ performs until $EUCLID((a, b), 0)$. Similarly, we acknowledge that for $\frac{a}{(a,b)} > \frac{b}{(a,b)} \geq 0$, $EUCLID(\frac{a}{(a,b)}, \frac{b}{(a,b)})$ performs until $EUCLID(1, 0)$.

Assume that $EUCLID(a, b)$ performs k recursive calls while $EUCLID(\frac{a}{(a,b)}, \frac{b}{(a,b)})$ performs k' recursive calls, we can prove $k = k'$ as follows:

After a recursive, for $EUCLID(\frac{a}{(a,b)}, \frac{b}{(a,b)})$, according to **Lemma** 3, we have:

$$\begin{aligned} EUCLID(\frac{a}{(a,b)}, \frac{b}{(a,b)}) &\to EUCLID(\frac{b}{(a,b)}, \frac{a}{(a,b)} \mod \frac{b}{(a,b)}) \\ &= EUCLID(\frac{b}{(a,b)}, \frac{a \mod b}{(a,b)}) \end{aligned} \tag{10}$$

Considering the end condition of the recursion, we can find that for $EUCLID(a, b)$ and $EUCLID(\frac{a}{(a,b)}, \frac{b}{(a,b)})$, the recursive process of the two of them corresponds to each other one by one in the following table:

	(a,b)		$(\frac{a}{(a,b)},\frac{b}{(a,b)})$
0	$EUCLID(a,b)$	0	$EUCLID(\frac{a}{(a,b)},\frac{b}{(a,b)})$
1	$EUCLID(b,a \mod b)$	1	$EUCLID(\frac{b}{(a,b)},\frac{a \mod b}{(a,b)})$
$\cdots$	$\cdots$	$\cdots$	$\cdots$
$k-1$	$EUCLID(\cdots,(a,b))$	$k'-1$	$EUCLID(\cdots,1)$
k	$EUCLID((a,b),0)$	k'	EUCLID(1, 0)

According to the table, we can find that $k = k'$. According **Lemma** 2, for $EUCLID(\frac{a}{(a,b)},\frac{b}{(a,b)})$, we have $\frac{b}{(a,b)}\ geqF_{k'+1}$. So, use the conclusion $k \leq 1 + \log_\Phi(b)$ mentioned above, we have:

$$k = k' \leq 1 + \log_\Phi(\frac{b}{(a,b)}) \tag{11}$$

4 31.4-1

Firstly, we can solve the problem $7x \equiv 2 \mod 10$. Obviously, the result is $x \equiv 6 \mod 10$.

Back to the initial problem $35x \equiv 10 \mod 50$, the result is as follows: $x \equiv 6 \mod 50$ or $16 \mod 50$ or $26 \mod 10$ or $36 \mod 10$ or $46 \mod 10$.

5 31.5-2

The problem is as follows:

$$\begin{cases} x \equiv 1 \mod 9 \\ x \equiv 2 \mod 8 \\ x \equiv 3 \mod 7 \end{cases} \tag{12}$$

Obviously, $x = 10$ is a solution of the problem. According to the **Chinese Remainder Theorem**: the general solution of the equation is $x \equiv 10 \mod 7 \times 8 \times 9$, which is $x \equiv 10 \mod 504$.

6 31.7-2

According to RSA public key encryption system, we know that $ed \equiv 1 \mod \Phi(n)$, $n = pq$ and $\Phi(n) = (p-1)(q-1)$.

If $e = 3$, we have $3d \equiv 1 \mod \Phi(n)$. According to the topic, $0 < d < \Phi(n)$, so $3d-1 = \Phi(n)$ or $3d-1 = 2\Phi(n)$. Consider the relationship between n and $3d-1$, we can determine $\Phi(n)$ as follows:

$$\begin{cases} \Phi(n) = 3d - 1 & 3d - 1 < n \\ \Phi(n) = \frac{3d-1}{2} & 3d - 1 > n \end{cases} \tag{13}$$

So, we can get the following conclusion:

$$\begin{cases} \Phi(n) = \frac{3d-1}{k} = (p-1)(q-1) \\ n = pq \end{cases} \tag{14}$$

where $k = 1$ or $k = 2$. Consider that $\Phi(n) = pq - p - q + 1 = n - p - q + 1$. So, we have:

$$\begin{cases} pq = n \\ p + q = n - \frac{3d-1}{k} + 1 \end{cases} \tag{15}$$

Once we determine k (which can be solved just by **ADDITION**, **SUBTRACTION**, **MULTIPLICATION**), the equation can be solved just by **ADDITION**, **SUBTRACTION**, **MULTIPLICATION** and **DIVISION**.

As we know, **ADDITION**, **SUBTRACTION**, **MULTIPLICATION** and **DIVISION** are all polynomial time operations with respect to number of n digits. So, the problem can be solved in polynomial time with respect to number of n digits.

7 31.8-3

Since x is the nontrivial square root of 1 modulo n, we can get that $x^2 \equiv 1 \mod n$. Since x is nontrivial, $x \neq kn \pm 1$.

Since $x^2 \equiv 1 \mod n$, we have $(x-1)(x+1) \equiv 0 \mod n$.

Assume $(x-1, n) = 1$, in other word, $x + 1 \equiv 0 \mod n$. However, $x \neq kn \pm 1$ (means $x + 1 \neq kn$) which is a contradiction. Similarly, we can prove that $(x+1, n) = 1$ is also a contradiction. According to the above, we can get that $(x-1, n) \neq 1$ and $(x+1, n) \neq 1$. Meanwhile, $(x-1, n) \neq n$ (because it needs $x = kn+1$ which is contradict) and $(x+1, n) \neq n$ (because it needs $x = kn-1$ which is contradict).

Obviously, $(x-1, n)|n$ and $(x+1, n)|n$, which means $(x-1, n)$ and $(x+1, n)$ are factors of n. And they are not trivial factors of n as mentioned above, so they are nontrivial factors of n.