Week 2

Lecture 1

University of Melbour

Lecture 1

Part -1 Extended GCD Algorithm and Related Computations

Part -2 Symmetric key Cryptography

Lect

Properties of Numbers

Workshop 2: Workshops start from this week

Quizz 2

1.1 Ex
1.2 In
1.3 Ex

Assignment Project Exam Help

1.1 Ex

https://eduassistpro.github.i

Add WeChat edu_assist_pr

Let us look at the gcd computation again with general numbers a and b with $a > b > 0$. Let $a_0 = a$; $a_1 = b$ and $q_1 = \lfloor a_0/a_1 \rfloor$

			$gcd(a_0, a_1)$	
a_0				$a_1 \rfloor$
a_1				$a_2 \rfloor$
a_2				$a_3 \rfloor$
	$\vdots$			
a_{t-2}	$=$	$q_{t-1} \times a_{t-1} + a_t$	$gcd(a_{t-}$	$_1 \rfloor$
a_{t-1}	$=$	$q_t \times a_t + 0$	$gcd(a$	

Table: Computation of $gcd(a, b)$

By using the fact on *gcd* before, we have

$$gcd(a, b) = gcd(a_0, a_1) = gcd(a_1, a_2) = \cdots = gcd(a_{t-1}, a_t) = gcd(a_t, 0)$$

Solvi

the fir nd a_1.

The following example illustrates the above poin
proving version of the algorithm is given at the end of t
slides.

Consider $gcd(33, 21)$:

$$
\begin{aligned}
33 &= 1 \times 21 + 12 & gcd(21, 12) && (A)\\
21 &= 1 \times 12 + 9 & gcd(12, 9) && (B)\\
12 &= 1 \times 9 + 3 & gcd(9, 3) && (C)
\end{aligned}
$$

$$3 = 12 - 1 \times 9$$

$$3 = 12 - 1 \times (21 - 1 \times 12$$

$$3 = 2 \times 12 - 1 \times 21$$

$$3 = 2 \times (33 - 1 \times 21) - 1 \times 21 \qquad From(A)$$

$$3 = 2 \times 33 + (-3) \times 21 \qquad Simplification$$

1.2 In

Let a and b be integers and let n be a positive integer.
We say "a" is congruent to "b", modulo n and write

$$a \equiv b \ (mod \ \ n),$$

if a and b differ by a multiple of n; i.e ; if n is a factor of $|b - a|$.
Ever
the se

We can define the following operations:

$$x \oplus_n y = (x + y)$$

$$x \otimes_n y = (xy) \ mod \ \ n$$

When the context is clear we use the above special addition and multiplication symbols interchangeably with their counterpart regular symbols.

Definition

Let x Z_n, if there is an integer y such that

then

$y = x^{-1}$ *usually.*

Example: let $n = 5$, 2 is inverse of 3 in
inverse of 3 modulo 5.

Fact

For a ... *hat*

You can determine x and y by modifyin ... $gcd(a, b)$. Thus we can say that we can find inverse o ... lo b provided $gcd(a, b) = 1$.

If $gcd(a, n)$ is 1 then we can use extended Euclid's algorithm on a and n and get two integers x and y such that

Taki

$$xa = 1 \bmod$$

Clearly x is the inverse of a mod n.

If $gcd(n, a)$ is 1 then we can use extended Euclid's algorithm on a and n and get two integers x and y such that

$$xn + ya = 1.$$

Taki

Clearly y is the inverse of a mod n. Not unique. Also it is clear that if $gcd(n, a$

exist. **Note**: *The output of the extended gcd al* *the inverse of a given integer depends on the order of the input arguments.*

Consider $gcd(13, 25)$:

$$
\begin{aligned}
25 &= 1 \times 13 + 12 \quad gcd(13, 12) \quad (A) \\
13 &= 1 \times 12 + 1 \quad gcd(12, 1) \quad (B)
\end{aligned}
$$

$$
\begin{aligned}
1 &= 13 - 1 \times 12 \\
1 &= 13 - 1 \times (25 - 1 \times 13) \\
1 &= 2 \times 13 - 1 \times 25 \\
1 &= 2 \times 13 + (-1) \times 25 \qquad \textit{Simplification}
\end{aligned}
$$

It is easy to see now, 2 is inverse of 13 mod 25.

Magma

Magma is a symbolic mathematical software package which can help you to do computations in algebra, number theory and geometry.

http://magma.maths.usyd.edu.au/magma/

An online calculator is available here:

http:

===

Exte

→Rn

RngIntElt, RngIntElt

Xgcd(m, n) : RngIntElt, RngIntElt → Rn

XGCD(m, n) : RngIntElt, RngIntElt → R

=======================

The extended GCD of m and n; returns integers g, x and y such that g is the greatest common divisor of the integers m and n, and g = x.m + y.n. If m and n are both zero, g is zero; otherwise g is always positive. If m and n are both non-zero, the multipliers x and y are unique.

1.3 Ex

Theorem

Given two positive integers a and b with $a > b$, let $a_0 = a$, $a_1 = b$ and $q_1 = \lfloor a_0/a_1 \rfloor$. Perform the following matrix equations for

$r = 1$

$q_r = $

$$ \quad r \qquad\qquad\qquad\qquad r-1 $$

$$ a_{r+1} \quad = \quad 1 \quad -q_r $$

until $a_{n+1} = 0$, where n is an integer. Then ... and b.

Proof: You can convince that the termination of the algorithm is well defined since $a_{r+1} < a_r$. So eventually, for some n, $a_{n+1} = 0$.

- hence we can write the recursion as the following matrix equation:

$$\begin{bmatrix} a_n \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & -q_n \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & -q_{n-1} \end{bmatrix} \cdots \begin{bmatrix} 0 & 1 \\ 1 & -q_1 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix}.$$

Hence, we have

$l=n$

Where $\prod_l$ is the symbol for multiplicatio
only the first row of the above matrix equatio
$a_n = A_{1,1}\ a_0 + A_{1,2}\ a_1$, where is the
RHS of the above equation. Thus any divisor of both $a_0 = a$ and $a_1 = b$ divides a_n. Hence, greatest common divisor $gcd(a, b)$ also divides a_n.

- Further observe that,

$$\begin{bmatrix} 0 & 1 \\ 1 & -q_r \end{bmatrix}^{-1} = \begin{bmatrix} q_r & 1 \\ 1 & 0 \end{bmatrix}$$

a_1 $l=1$ 1 0 0

So a_n must divide both $a_0 = a$ an
$gcd(a, b)$.
Thus $a_n = gcd(a, b)$.

Some implications of the theorem. Let

$$A^r = \left\{ \prod_{l=r}^{1} \begin{bmatrix} 0 & 1 \\ 1 & -q_l \end{bmatrix} \right\} = \begin{bmatrix} 0 & 1 \\ 1 & -q_r \end{bmatrix} A^{r-1}.$$

The

For a ... *that*
gcd(

Proof

From Theorem 1, we have

$$\begin{bmatrix} a_n \\ 0 \end{bmatrix} = A^n \begin{bmatrix} a \\ b \end{bmatrix}.$$

Hence $gcd(a, b) := a_n = A^n_{11}\ a + A^n_{12}\ b$.

Similarly prove the following theorem.

Theorem

The m

$a = ($

$b = ($

21

Lecture 1

Part -1 Extended GCD Algorithm and Related Computations

Part -2 Symmetric key Cryptography

Lect

Properties of Numbers

Workshop 2: Workshops start from this week

Quizz 2