Algorithms for Numbers and Public-key cryptography

Part 2

Jean-Sébastien Coron

Université du Luxembourg

March 13, 2014

Summary

- C programming
  - Pointers and dynamic arrays.
  - Functions
- Number theory
  - Congruence.
  - Euclid's extended algorithm
  - Modular arithmetic.
  - Solving linear congruence equations.
  - Chinese remainder theorem.

Pointers

- A pointer is a memory address.
  - When a variable is declared, some memory is allocated to it.
  - The address is obtained using &

```
// allocated memory for a
int a;

// prints the address of a
// (for ex: 2678673).
printf("%d\n",&a);
```

Pointers

- Pointer declaration:
  - Integer pointer: `int *p;`
  - Char pointer: `char *pc;`
  - Float pointer: `float *pf;`
- Access to content:
  - `*p` is the value at address *p*.

Example

```
int a; // allocate memory for a
a=2;

int *p; //
p=&a; // p is now a pointer to a

printf("%d\n",*p);
// prints the content at address p
// 2.

*p=3; // now a=3
```

Memory allocation

- Pointer declaration:
  - `int *p;`
  - Does not allocate memory at address `p`.
  - `*p=2;` can give an error.
- `p` can become a pointer to an existing variable:
  - `int a; int *p; p=&a;`
- Or one can allocate memory for `p`.
  - Using `malloc`.
  - ```
    int *p;
    p=malloc(sizeof(int));
    ```
    ```

Dynamic arrays

- Dynamic array of size `n`:
  - ```
    int *t;
    t=malloc(n*sizeof(int));
    ```
  - `t[0]` to `t[n-1]`
- Dynamic size.
  - Not necessarily known at compilation time.
  - Known at execution time.
  - As opposed to
    ```
    int t[10];
    ```
    ```

Example

```
#include <stdio.h>
int main()
{
  int n;
  n=2*10;
  // n is known only at execution time

  int *p;
  p=malloc(n*sizeof(int));

  int i;
  for(i=0;i<n;i++) p[i]=0;
}
```

Freeing memory

- Function `free`.
  - `int *t=malloc(n*sizeof(int)); free(t);`

Functions

- Syntax :
  - ```
    rtype fname(para1,para2,...)
    {
      localvariables
      functioncode
    }
    ```
- Example :
  - ```
    double max(double a,double b)
    {
      double m;
      if(a>b) m=a; else m=b;
      return m;
    }
    ```

Using the function

- ```c
  #include <stdio.h>
  double max(double a,double b)
  {
    double m;
    if(a>b) m=a; else m=b;
    return m;
  }
  int main()
  {
    double x=3.5;
    double y=3.2;
    double z=max(x,y);
  }
  ```
  ```

void function

- A `void` function is a function that returns nothing.

```
#include <stdio.h>
void affiche(int a)
{
  printf("La valeur est:%d\n",a);
}

int main()
{
  int u=3;
  affiche(u);
}
```

Printing an array

```
#include <stdio.h>
void affiche(int tab[],int n)
{
  int i;
  for(i=0;i<n;i++) printf("%d ",tab[i]);
  printf("\n");
}

int main()
{
  int t[5]={1,3,6,5,1}
  affiche(t,5);
}
```

Number Theory

- Congruence.
- Euclid's extended algorithm
- Modular arithmetic.
- Solving linear congruence equations.
- Chinese remainder theorem.

Basic Properties of Integers

Theorem (Fundamental theorem of arithmetic)

Every non-zero integer n can be expressed as

$$n = \pm p_i^{e_1} \cdots p_r^{e_r}$$

where the p_i's are distinct primes and the e_i are positive integers. Moreover the decomposition is unique, up to reordering of the primes.

- Proof: existence is easy by recursion; unicity: see any standard textbook.

Basic Properties of Integers

Theorem (Division with remainder property)

For $a, b \in \mathbb{Z}$ with $b > 0$, there exist unique $q, r \in \mathbb{Z}$ such that $a = bq + r$ and $0 \leq r < b$.

Congruence

- Definition
  - Let $n > 0$, and $a, b \in \mathbb{Z}$.
  - a is *congruent* to b if $n | (a - b)$.
  - $a \equiv b \pmod{n}$.
  - n is called the *modulus*.
  - Should not be confused with the *mod* of Euclidean division.

Theorem

Let $n > 0$. For any integer a, there exists a unique integer b such that $a \equiv b \pmod{n}$ and $0 \leq b < n$, namely $b := a \bmod n$.

Examples and properties

- Examples :
  - $2 \equiv 8 \mod 3$ since $3|(8-2)$.
  - $12 \equiv 2 \mod 5$ since $5|(12-2)$.
- Properties :
  - $a \equiv b \mod n \Leftrightarrow \exists k \in \mathbb{Z}, a = b + k \cdot n$.
  - $a \equiv a \mod n$
  - $a \equiv b \mod n \Rightarrow b \equiv a \mod n$
  - $a \equiv b \mod n$ and $b \equiv c \mod n$ implies $a \equiv c \mod n$

Properties

- Addition and multiplication
  - If $a \equiv a' \mod n$ and $b \equiv b' \mod n$, then
  - $a + b \equiv a' + b' \mod n$ and $a \cdot b \equiv a' \cdot b' \mod n$.
- When computing modulo n, one can substitute to x a value x' congruent to x modulo n.
  - Computing a with $0 \leq a < 8$ such that $a \equiv 83 \cdot 72 \mod 7$.
  - First solution: $83 \cdot 72 = 5976$
    $a = 5976 \mod 7 = 5$.
  - Second solution: $83 \equiv 6 \mod 7$, $72 \equiv 2 \mod 7$,
    $83 \cdot 72 \equiv 6 \cdot 2 \equiv 12 \equiv 5 \mod 7$.

Multiplicative inverse

- Multiplicative inverse :
  - Let $n > 0$ and $a \in \mathbb{Z}$. An integer a' is a *multiplicative inverse* of a modulo n if $a \cdot a' \equiv 1 \mod n$.
- Theorem :
  - Let $n, a \in \mathbb{Z}$ with $n > 0$. Then a has a multiplicatif inverse modulo n iff $\mathrm{PGCD}(a, n) = 1$.
  - Proof $(\Rightarrow)$
    - If a' is a multiplicative inverse of a modulo n, then $a \cdot a' \equiv 1 \mod n$.
    - Let $k \in \mathbb{Z}$ such that $a \cdot a' = 1 + k \cdot n$.
    - If $d|a$ and $d|n$, then $d|1$. Therefore $\mathrm{PGCD}(a, n) = 1$.

Example

- A multiplicative inverse of 5 modulo 7 is 3 because

$$3 \cdot 5 \equiv 15 \equiv 1 \mod 7$$

- 2 has no multiplicative inverse modulo 6 :
  - $2 \cdot 1 \equiv 2 \mod 6$
  - $2 \cdot 2 \equiv 4 \mod 6$
  - $2 \cdot 3 \equiv 0 \mod 6$
  - $2 \cdot 4 \equiv 2 \mod 6$
  - $2 \cdot 5 \equiv 4 \mod 6$

Euclid's extended algorithm

- Euclid's extended algorithm
  - Let $a, b \in \mathbb{Z}$ and $d = \mathrm{PGCD}(a, b)$.
  - Computes $s, t \in \mathbb{Z}$ such that $a \cdot s + b \cdot t = d$.
- Multiplicative inverse.
  - Let a, n with $n > 0$ and $\mathrm{PGCD}(a, n) = 1$.
  - With Euclid's extended algorithm, one computes s, t such that

$$a \cdot s + n \cdot t = 1$$

  - Then $a \cdot s \equiv 1 \mod n$
  - s is one multiplicative inverse of a modulo n.

Euclid's extended algorithm

- Euclid's extended algorithm, for $a > 0$ and $b \geq 0$.
  - Two additional sequences u_i and v_i.
  - $r_0 = a$ and $r_1 = b$.
  - For $i \geq 0$, let $r_i = q_i \cdot r_{i+1} + r_{i+2}$
  - $u_0 := 1,\ v_0 := 0,\ u_1 := 0,\ v_1 := 1$ and for $i \geq 2$, one defines $u_i = u_{i-2} - q_{i-2} \cdot u_{i-1}$ and $v_i = v_{i-2} - q_{i-2} \cdot v_{i-1}$.
- There exists $k > 0$ such that $r_k = 0$.
  - Then $\mathrm{PGCD}(a, b) = r_{k-1} = u_{k-1} \cdot a + v_{k-1} \cdot b$.

Proof

- We always have $r_i = u_i \cdot a + v_i \cdot b$.
  - True for $r_0 = a = 1 \cdot a + 0 \cdot b$.
  - True for $r_1 = b = 0 \cdot a + 1 \cdot b$.
  - If $r_{i-2} = u_{i-2} \cdot a + v_{i-2} \cdot b$ and $r_{i-1} = u_{i-1} \cdot a + v_{i-1} \cdot b$, then :

$$\begin{aligned} u_i \cdot a + v_i \cdot b &= (u_{i-2} - q_{i-2} \cdot u_{i-1}) \cdot a + \\ &\quad (v_{i-2} - q_{i-2} \cdot v_{i-1}) \cdot b \\ &= r_{i-2} - q_{i-2} \cdot r_{i-1} \\ &= r_i \end{aligned}$$

Modular arithmetic

- Let an integer $n > 1$ called the modulus.
- Modular reduction
  - $r := a \mod n$, remainder of the division of a by n.
  - $0 \leq r < n$
  - Ex: $11 \mod 8 = 3$, $15 \mod 5 = 0$.
- Congruence:
  - $a \equiv b \mod n$ if $n|(a - b)$.
  - $a \equiv b \mod n$ iif a and b have same remainder modulo n.
  - Ex: $11 \equiv 19 \mod 8$.
  - If $r := a \mod n$, then $r \equiv a \mod n$.

Modular arithmetic

- If $a_0 \equiv b_0 \mod n$ and $a_1 \equiv b_1 \mod n$
  - $a_0 + a_1 \equiv b_0 + b_1 \mod n$
  - $a_0 - a_1 \equiv b_0 - b_1 \mod n$
  - $a_0 \cdot a_1 \equiv b_0 \cdot b_1 \mod n$
- Integers modulo n
  - Integers modulo n are $\mathbb{Z}_n = \{0, 1, \ldots, n-1\}$
  - Addition, subtraction or multiplication in $\mathbb{Z}_n$ is done by first doing it in $\mathbb{Z}$ and then reducing the result modulo n.
  - For example in $\mathbb{Z}_7$:
    - $6 + 4 = 3$, $3 - 4 = 6$, $3 \cdot 6 = 4$.

Solving linear congruence

- Theorem: let two integers a, n with $n > 0$ such that $\mathrm{PGCD}(a, n) = 1$. Let $b \in \mathbb{Z}$. The equation $a \cdot x \equiv b \mod n$ has a unique solution x modulo n.
  - Let a^{-1} by the multiplicative inverse of a modulo n.

$$a \cdot a^{-1} \cdot x \equiv x \equiv a^{-1} \cdot b \mod n$$

- Example :
  - Find x such that $5 \cdot x \equiv 6 \mod 7$
  - 3 is the inverse of 5 modulo 7 because $5 \cdot 3 \equiv 1 \mod 7$.
  - $3 \cdot 5 \cdot x \equiv 15 \cdot x \equiv 1 \cdot x \equiv 3 \cdot 6 \equiv 4 \mod 7$
  - $x \equiv 4 \mod 7$

Modular division

- Modular quotient $b/a \mod n$.
  - Let $a, b \in \mathbb{Z}$, and n a modulus.
  - If $\text{PGCD}(a, n) = 1$, then one defines the *modular quotient* $b/a \mod n$ as $b \cdot a^{-1} \mod n$.
  - With a^{-1} the multiplicative inverse of a modulo n.
- If $c \equiv b/a \mod n$, then $a \cdot c \equiv b \mod n$
  - c is solution of $a \cdot x \equiv b \mod n$
- Example :
  - $5/3 \equiv 4 \mod 7$

Chinese remainder theorem

- Chinese remainder theorem
  - Let two integers $n_1 > 1$ and $n_2 > 0$ with $\mathrm{PGCD}(n_1, n_2) = 1$.
  - For all $a_1, a_2 \in \mathbb{Z}$, there exists an integer z such that

$$\begin{aligned} z &\equiv a_1 \mod n_1 \\ z &\equiv a_2 \mod n_2 \end{aligned}$$

  - z is unique modulo $n_1 \cdot n_2$.

Proof

- Existence :
  - Let $m_1 = (n_2)^{-1} \mod n_1$ and $m_2 = (n_1)^{-1} \mod n_2$

$$z := n_2 \cdot m_1 \cdot a_1 + n_1 \cdot m_2 \cdot a_2$$

  - $z \equiv (n_2 \cdot m_1) \cdot a_1 \equiv a_1 \mod n_1$
  - $z \equiv (n_1 \cdot m_2) \cdot a_2 \equiv a_2 \mod n_2$
- Unicity modulo $n_1 \cdot n_2$
  - Let $z'' = z - z'$. Then $n_1 | z''$ and $n_2 | z''$.
  - Since $\text{PGCD}(n_1, n_2) = 1$, $n_1 \cdot n_2 | z''$.
  - $z \equiv z' \mod (n_1 \cdot n_2)$