ریاضیات گسسته

دکتر منصوره میرزایی

فصل چهارم

نظریه اعداد

تقسیم پذیری

- تعریف: فرض کنید a و b دو عدد صحیح باشند و $a \neq 0$، گوییم عدد a عدد b را می شمارد، یا b مضربی از a است، اگر عدد صحیح c وجود داشته باشد به طوری که $b = ac$ و می نویسیم $a|b$

تقسیم پذیری

- قضیه: فرض کنید a و b و c اعداد صحیح باشند، بطوریکه $a \neq 0$ در این صورت:
  - اگر a|b و a|c آنگاه a|b+c
  - اگر a|b آنگاه برای همه اعداد صحیح c داریم: a|bc
  - اگر a|b و b|c آنگاه a|c
- نتیجه: اگر a و b و c اعداد صحیح و $a \neq 0$ و a|b و a|c، a|mb+nc که m و n اعداد صحیح هستند.

تقسیم پذیری

- قضیه: اگر a یک عدد صحیح و d یک عدد صحیح مثبت باشد، آنگاه اعداد یکتای q و r وجود دارند بطوریکه $0 \leq r < d$ و a=dq+r و به صورت زیر نشان میدهیم:

q = a **div** d

r = a **mod** d

تقسیم پذیری

- فرض کنید a و b اعداد صحیح و m یک عدد صحیح مثبت باشد، در این صورت گوییم a همنهشت b در پیمانه m است اگر m|a-b و آن را به صورت $a \equiv b \pmod{m}$
- قضیه: اگر a و b اعداد صحیح و m یک عدد صحیح مثبت باشد، آنگاه $a \equiv b \pmod{m}$ اگر و فقط اگر $a \bmod m = b \bmod m$

تقسیم پذیری

- قضیه: فرض کنید m یک عدد صحیح مثبت باشد، اعداد صحیح a و b در پیمانه m همنهشت هستند اگر و فقط اگر عدد صحیح k وجود داشته باشد بطوریکه $a=mk+b$
- تعریف: به مجموعه همه اعداد صحیحی که همنشهت a در پیمانه m هستند، کلاس مانده های a در پیمانه m می نامند.

نمایش اعداد صحیح

- فرض کنید b یک عدد صحیح بزرگتر از ۱ باشد. آنگاه اگر n یک عدد صحیح مثبت باشد، میتوان آن را به صورت یکتا به شکل زیر نمایش داد:

$$n = a_k b^k + a_{k-1} b^{k-1} + \cdots + a_1 b + a_0,$$

که k یک عدد صحیح نامنفی است و a_0، a_1، ... a_k اعداد صحیح نامنفی کمتر از b هستند و $a_k \neq 0$

در این حالت به $(a_k a_{k-1} \ldots a_0)_b$ نمایش مبنای b عدد n گفته میشود.

تبدیل به مبنای ۱۰

- تبدیل عدد از مبنای دلخواه به مبنای ۱۰ با استفاده از ارزش مکانی هر رقم انجام میشود. مجموع حاصلضرب هر رقم در ارزش مکانی هر رقم باید محاسبه شود.

مثال:

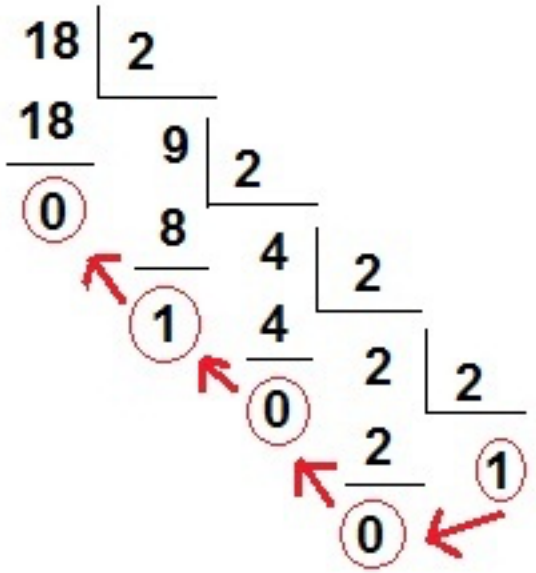
$(245)_8 = 2 . 8^2 + 4 . 8 + 5 = 165$

$(101011)_2 = 1 . 2^5 + 0 . 2^4 + 1 . 2^3 + 0 . 2^2 + 1 . 2 + 1 = 43$

$(2AE0B)_{16} = 2 . 16^4 + 10 . 16^3 + 14 . 16^2 + 0 . 16 + 11 = 175627$

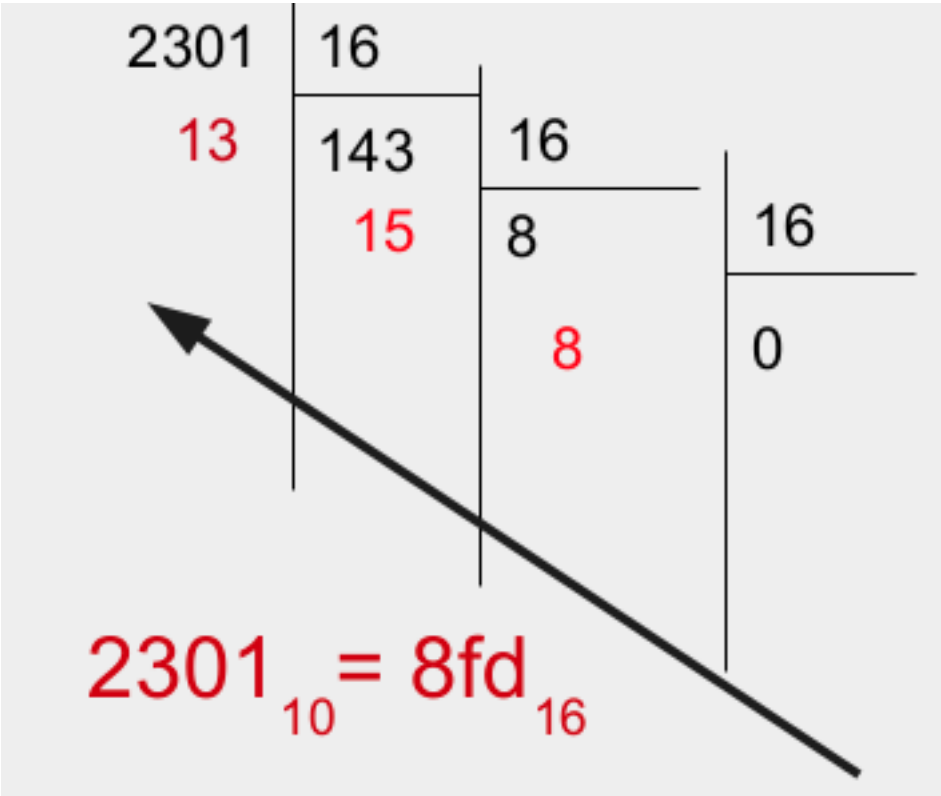
تبدیل از مبنای ۱۰ به یک مبنای دیگر

- برای تبدیل از مبنای ۱۰ به مبنای b با تقسیم متوالی بر b انجام میشود.



تبدیل از مبنای ۱۰ به یک مبنای دیگر

- تبدیل به مبنای ۱۶



تبدیل بین مبنای ۲، ۸ و ۱۶

- تبدیل مبنای ۲ به 2^n: از سمت راست n رقم n رقم جدا میکنیم و به جای هر n رقم معادل آن در مبنای 2^n را قرار میدهیم.

مثال:

$(110101001)_2 = (110\ \ 101\ \ 001)_2 = (651)_8$

$(110101001)_2 = (0001\ \ 1010\ \ 1001)_2 = (1A9)_{16}$

تبدیل بین مبنای ۲، ۸ و ۱۶

- تبدیل از مبنای 2^n به مبنای ۲: به جای هر رقم، n رقم در مبنای ۲ قرار میدهیم.

مثال:

$(7103)_8 = (111\ \ 001\ \ 000\ \ 011)_2$

$(A19E)_{16} = (1010\ \ 0001\ \ 1001\ \ 1110)_2$

جمع اعداد درمبنای ۲

- الگوریتم جمع اعداد مبنای ۲

ALGORITHM 2 Addition of Integers.

procedure *add*(a, b: positive integers)
{the binary expansions of a and b are $(a_{n-1}a_{n-2}\ldots a_1a_0)_2$
and $(b_{n-1}b_{n-2}\ldots b_1b_0)_2$, respectively}
$c := 0$
for $j := 0$ **to** $n-1$
$d := \lfloor (a_j + b_j + c)/2 \rfloor$
$s_j := a_j + b_j + c - 2d$
$c := d$
$s_n := c$
return $(s_0, s_1, \ldots, s_n)$ {the binary expansion of the sum is $(s_n s_{n-1}\ldots s_0)_2$}

جمع اعداد درمبنای ۲

مثال: حاصلجمع دو عدد $(1110)_2$ و $(1011)_2$:

$$\begin{array}{r} {\scriptstyle 1\ 1\ 1\ \ \ } \\ 1\ 1\ 1\ 0 \\ +\ 1\ 0\ 1\ 1 \\ \hline 1\ 1\ 0\ 0\ 1 \end{array}$$

$$a_0 + b_0 = 0 + 1 = 0 \cdot 2 + 1,$$

so that $c_0 = 0$ and $s_0 = 1$. Then, because

$$a_1 + b_1 + c_0 = 1 + 1 + 0 = 1 \cdot 2 + 0,$$

it follows that $c_1 = 1$ and $s_1 = 0$. Continuing,

$$a_2 + b_2 + c_1 = 1 + 0 + 1 = 1 \cdot 2 + 0,$$

so that $c_2 = 1$ and $s_2 = 0$. Finally, because

$$a_3 + b_3 + c_2 = 1 + 1 + 1 = 1 \cdot 2 + 1,$$

follows that $c_3 = 1$ and $s_3 = 1$. This means that $s_4 = c_3 = 1$. Therefore, $s = a + b = (1\,1001)_2$.

ضرب اعداد در مبنای ۲

- الگوریتم ضرب اعداد در مبنای ۲

ALGORITHM 3 Multiplication of Integers.

procedure *multiply*(a, b: positive integers)
{the binary expansions of a and b are $(a_{n-1}a_{n-2}\ldots a_1a_0)_2$ and $(b_{n-1}b_{n-2}\ldots b_1b_0)_2$, respectively}
for $j := 0$ **to** $n-1$
    if $b_j = 1$ **then** $c_j := a$ shifted j places
    else $c_j := 0$
{$c_0, c_1, \ldots, c_{n-1}$ are the partial products}
$p := 0$
for $j := 0$ **to** $n-1$
    $p := p + c_j$
return p {p is the value of ab}

ضرب اعداد در مبنای ۲

مثال: حاصلضرب دو عدد $(۱۱۰)_۲$ و $(۱۰۱)_۲$ را پیدا کنید:

$$
\begin{array}{r}
1\,1\,0 \\
\times\ 1\,0\,1 \\
\hline
1\,1\,0 \\
0\,0\,0 \\
1\,1\,0 \\
\hline
1\,1\,1\,1\,0
\end{array}
$$

$$ab_0 \cdot 2^0 = (110)_2 \cdot 1 \cdot 2^0 = (110)_2,$$
$$ab_1 \cdot 2^1 = (110)_2 \cdot 0 \cdot 2^1 = (0000)_2,$$

and

$$ab_2 \cdot 2^2 = (110)_2 \cdot 1 \cdot 2^2 = (11000)_2.$$

add $(110)_2$, $(0000)_2$, and $(11000)_2$.

$$ab = (1\ 1110)_2$$

محاسبه باقیمانده و تقسیم صحیح

ALGORITHM 4 **Computing div and mod.**

procedure *division algorithm*(a: integer, d: positive integer)
$q := 0$
$r := |a|$
while $r \geq d$
    $r := r - d$
    $q := q + 1$
if $a < 0$ and $r > 0$ **then**
    $r := d - r$
    $q := -(q + 1)$
return (q, r) {$q = a$ **div** d is the quotient, $r = a$ **mod** d is the remainder}

توان رسانی پیمانه ای (modular exponentiation)

- فرض کنید میخواهیم $b^n \bmod m$ را محاسبه کنیم.
- ابتدا عدد n را در مبنای ۲ مینویسیم. $n = (a_{k-1}a_{k-2}...a_1a_0)_2$. بنابراین

$$b^n = b^{a_{k-1}\cdot 2^{k-1}+\cdots+a_1\cdot 2+a_0} = b^{a_{k-1}\cdot 2^{k-1}} \ldots b^{a_1\cdot 2} \cdot b^{a_0}.$$

بنابراین فقط نیاز به محاسبه b^{2^j} برای j های مختلف داریم. برای هر کدام باقیمانده را به پیمانه m حساب میکنیم.

توان رسانی پیمانه ای

- الگوریتم توان رسانی پیمانه ای

ALGORITHM 5 Modular Exponentiation.

procedure *modular exponentiation*(b: integer, $n = (a_{k-1}a_{k-2}\ldots a_1a_0)_2$,
m: positive integers)
$x := 1$
$power := b$ **mod** m
for $i := 0$ **to** $k - 1$
  if $a_i = 1$ **then** $x := (x \cdot power)$ **mod** m
  $power := (power \cdot power)$ **mod** m
return x {x equals b^n **mod** m}

توان رسانی پیمانه ای

مثال: $3^{644} \bmod 645$ را با استفاده از الگوریتم پیدا کنید.

$644 = (1010000100)_2$

$i = 0$: Because $a_0 = 0$, we have $x = 1$ and $power = 3^2 \bmod 645 = 9 \bmod 645 = 9$;
$i = 1$: Because $a_1 = 0$, we have $x = 1$ and $power = 9^2 \bmod 645 = 81 \bmod 645 = 81$;
$i = 2$: Because $a_2 = 1$, we have $x = 1 \cdot 81 \bmod 645 = 81$ and $power = 81^2 \bmod 645 = 6561 \bmod 645 = 111$;
$i = 3$: Because $a_3 = 0$, we have $x = 81$ and $power = 111^2 \bmod 645 = 12{,}321 \bmod 645 = 66$;
$i = 4$: Because $a_4 = 0$, we have $x = 81$ and $power = 66^2 \bmod 645 = 4356 \bmod 645 = 486$;
$i = 5$: Because $a_5 = 0$, we have $x = 81$ and $power = 486^2 \bmod 645 = 236{,}196 \bmod 645 = 126$;
$i = 6$: Because $a_6 = 0$, we have $x = 81$ and $power = 126^2 \bmod 645 = 15{,}876 \bmod 645 = 396$;
$i = 7$: Because $a_7 = 1$, we find that $x = (81 \cdot 396) \bmod 645 = 471$ and $power = 396^2 \bmod 645 = 156{,}816 \bmod 645 = 81$;
$i = 8$: Because $a_8 = 0$, we have $x = 471$ and $power = 81^2 \bmod 645 = 6561 \bmod 645 = 111$;
$i = 9$: Because $a_9 = 1$, we find that $x = (471 \cdot 111) \bmod 645 = 36$.

اعداد اول (prime)

- مبحث اعداد اول در رمزنگاری کاربرد بسیاری دارد.
- عدد صحیح p که بزرگتر از ۱ است را اول گوییم اگر فقط بر ۱ و p بخشپذیر باشد. اعداد غیر اول را مرکب گوییم.
- قضیه اساسی حساب: هر عدد صحیح بزرگتر از ۱ میتواند بطور یکتا به صورت یک عدد اول یا حاصلضربی دو یا چند عدد اول بیان شود که در آن فاکتورهای اول به صورت غیر کاهشی نوشته شده اند.

آزمون تقسیم (trial division)

- قضیه اگر n یک عدد مرکب باشد، آنگاه یک مقسوم علیه اول کمتر یا مساوی با $\sqrt{n}$ دارد.
- آزمون تقسیم برای مشخص کردن اول بودن یک عدد استفاده میشود. به این ترتیب که عدد n را بر همه اعداد اول کوچکتر یا مساوی با $\sqrt{n}$ تقسیم میکنیم. اگر بر هیچکدام بخشپذیر نباشد، عدد اول است.
  - برای پیدا کردن فاکتورهای اول یک عدد هم به همین ترتیب عمل میکنیم.

اعداد اول

- قضیه: بی نهایت عدد اول وجود دارد.
- قضیه اعداد اول: اگر $\pi(x)$ تعداد اعداد اول کوچکتر از x باشد، آنگاه

$$\lim_{x\to\infty} \frac{\pi(x)}{x/\ln x} = 1$$

مثال: در میان اعداد صحیح مثبت با حداکثر ۱۰۰۰ رقم، حدود ۱ عدد از هر ۲۳۰۰ تا اول است.

بزرگترین مقسوم علیه مشترک

- فرض کنید a و b اعداد صحیح غیر صفر باشند. بزرگترین مقسوم علیه مشترک a و b برابر است با بزرگترین عدد d که d|a و d|b و آن را با gcd(a,b) نشان میدهیم.
- اعداد a و b را نسبت به هم اول گوییم، اگر ب.م.م آنها ۱ باشد.

بزرگترین مقسوم علیه مشترک

- برای محاسبه ب.م.م میتوان از تجزیه اعداد به عوامل اول استفاده کرد.

$$a = p_1^{a_1} p_2^{a_2} \cdots p_n^{a_n},\ b = p_1^{b_1} p_2^{b_2} \cdots p_n^{b_n},$$

$$\gcd(a, b) = p_1^{\min(a_1, b_1)} p_2^{\min(a_2, b_2)} \cdots p_n^{\min(a_n, b_n)},$$

کوچکترین مضرب مشترک

- اگر اعداد a و b اعداد صحیح مثبت باشند، کوچکترین مضرب مشترک a و b برابر است با کوچکترین عددی که بر هر دو عدد a و b بخشپذیر باشد و آن را با lcm(a,b) نشان میدهیم.
- ک.م.م را میتوان به صورت زیر مشخص کرد:

$$\operatorname{lcm}(a,b) = p_1^{\max(a_1,b_1)} p_2^{\max(a_2,b_2)} \cdots p_n^{\max(a_n,b_n)},$$

کوچکترین مضرب مشترک

- قضیه: فرض کنید a و b اعداد صحیح مثبت باشند، در این صورت:

$$ab = gcd(a,b) \cdot Lcm(a,b)$$

الگوریتم اقلیدس

- برای محاسبه ب.م.م دو عدد میتوان از الگوریتم اقلیدس استفاده کرد.
- قضیه: فرض کنید $a = bq + r$ که a، b، q و r اعداد صحیح هستند.
  در این صورت $gcd(a,b) = gcd(b,r)$

الگوریتم اقلیدس

مثال: ب.م.م ۴۱۴ و ۶۶۲ را با الگوریتم اقلیدس محاسبه کنید.

$$662 = 414 \cdot 1 + 248$$
$$414 = 248 \cdot 1 + 166$$
$$248 = 166 \cdot 1 + 82$$
$$166 = 82 \cdot 2 + 2$$
$$82 = 2 \cdot 41.$$

$$\gcd(414, 662) = 2$$

الگوریتم اقلیدس

ALGORITHM 1 The Euclidean Algorithm.

procedure $gcd(a, b$: positive integers)
$x := a$
$y := b$
while $y \neq 0$
    $r := x$ **mod** y
    $x := y$
    $y := r$
return $x\{\gcd(a, b)$ is $x\}$

اعداد اول و ب.م.م

- اگر a و b و c اعداد صحیح مثبت باشند بطوریکه $gcd(a,b)=1$ و $a|bc$، آنگاه $a|c$
- اگر p عدد اول باشد و $p|a_1a_2...a_n$، که هر a_i یک عدد صحیح است، آنگاه i ای وجود دارد که $p|a_i$
- با استفاده از این دو قضیه میتوان یکتا بودن تجزیه به عوامل اول را ثابت کرد.

اعداد اول و ب.م.م

- قضیه: فرض کنید m یک عدد صحیح مثبت باشد و a و b و c اعداد صحیح باشند. اگر $ac \equiv bc \ (mod\ m)$ و gcd(c,m) = 1، آنگاه $a \equiv b \ (mod\ m)$