

*DAUGAVPILS UNIVERSITĀTE*  
*Dabaszinātņu un matemātikas fakultāte*  
*Matemātikas katedra*

*Studiju kurss*

## Veselo skaitļu teorija

### 5.lekcija (datoriķiem)

*Docētājs: Dr. P. Daugulis*

*2008./2009.studiju gads*

# Saturs

<b>1. Atlikumu klašu operāciju (modulārās aritmētikas) īpašības</b>	<b>4</b>
<b>2. Modulārās aritmētikas pielietojumi</b>	<b>9</b>
2.1. Pozicionālais pieraksts . . . . .	9
2.2. Informācijas glabāšana - hashing . . . . .	21
2.3. Informācijas drošība - paritātes un cita veida kongruenču pārbaude . . . . .	23
2.4. Nejaušo skaitļu ģenerēšana . . . . .	25
2.5. Aritmētisko operāciju pārbaude . . . . .	26
2.6. Dalāmības pazīmes . . . . .	27
2.6.1. Dalāmība ar 2 . . . . .	27
2.6.2. Dalāmība ar 3 . . . . .	28
2.6.3. Dalāmība ar 4 un vispārināšana uz $2^l$ . . . . .	28
2.6.4. Dalāmība ar 5 un vispārināšana uz $5^k$ . . . . .	29
2.6.5. Dalāmība ar 6 . . . . .	29
2.6.6. Dalāmība ar 9 . . . . .	30

	3
2.6.7. Dalāmība ar 11 . . . . .	30
<b>3. 5.mājasdarbs</b>	<b>31</b>

# 1. Atlikumu klašu operāciju (modulārās aritmētikas) īpašības

**1.1. teorēma.** Atlikumu gredzenā  $\mathbb{Z}/m\mathbb{Z}$  ir spēkā šādas īpašības:

1. katram  $x \in \mathbb{Z}/m\mathbb{Z}$  eksistē viens un tikai viens  $y \in \mathbb{Z}/m\mathbb{Z}$  tāds, ka

$$x + y \equiv 0 \pmod{m}$$

(aditīvi inversā elementa eksistence un viennozīmīgums),

2. ja  $p$  ir pirmskaitlis, tad

$$xy \equiv 0 \pmod{p} \implies x \equiv 0 \pmod{p} \text{ vai } y \equiv 0 \pmod{p}$$

(nulles dalītāju neeksistence),

3. ja  $p$  ir pirmskaitlis, tad katram  $x \in \mathbb{Z}/m\mathbb{Z}$  tādām, ka

$$x \not\equiv 0 \pmod{p}$$

eksistē viens un tikai viens  $z \in \mathbb{Z}/m\mathbb{Z}$ , kas apmierina vienādību

$$xz \equiv 1 \pmod{p},$$

4. ja  $m$  nav pirmskaitlis, tad eksistē nenulles elementi  $x$  un  $y$  tādi, ka

$$xy \equiv 0 \pmod{m},$$

5.  $x$  ir invertējams attiecībā uz reizināšanu pēc moduļa  $m$  (eksistē viens un tikai viens  $y$  tāds, ka  $xy \equiv 1 \pmod{m}$ ) tad un tikai tad, ja  $LKD(x, m) = 1$  (*multiplikatīvi inversā elementa eksistence*).

## PIERĀDĪJUMS

1. Katram  $x \in \mathbb{Z}$  eksistē viens un tikai viens  $y \in \mathbb{Z}$ , tāds, ka  $x + y = m$ , tātad skaitļa  $x$  atlikumu klase summā ar  $y$  klasi dos 0 klasi. Ja  $x + y_1 \equiv x + y_2 \equiv 0 \pmod{m}$ , tad  $y_1 \equiv y_2 \pmod{m}$ .

2. Ja  $p$  ir pirmskaitlis, tad no tā, ka  $p|xy$  seko, ka  $p|x$  vai  $p|y$ . Pārtulkojot to atlikumu klašu terminos: ja  $xy \equiv 0 \pmod{p}$ , tad  $x \equiv 0 \pmod{p}$  vai  $y \equiv 0 \pmod{p}$ .

3. Ja  $p$  ir pirmskaitlis, tad jebkurš vesels skaitlis  $x$  robežās no 1 līdz  $p - 1$  un  $p$  ir savstarpēji pirmskaitļi -  $LKD(x, p) = 1$ , tātad

saskaņā ar *LKD* lineārās kombinācijas īpašību eksistē veseli skaitļi  $a$  un  $b$  tādi, ka  $ax + bp = 1$  un, tādējādi

$$ax + bp \equiv ax + b \cdot 0 \equiv 1 \pmod{p},$$

tas nozīmē, ka skaitļa  $a$  klase reizinājumā ar  $x$  dod klasi 1,

4. Ja  $m$  nav pirmskaitlis, tad eksistē vismaz divi skaitļi  $a > 1$  un  $b > 1$ , tādi, ka  $ab = m$ , no kurienes seko, ka

$$ab \equiv m \equiv 0 \pmod{m}.$$

5. Ja  $LKD(x, m) = 1$ , tad eksistē skaitļi  $a$  un  $b$  tādi, ka

$$ax + bm = 1$$

un reducējot abas puses pēc moduļa  $m$ , iegūsim, ka

$$ax + bm \equiv ax + b \cdot 0 \equiv ax \equiv 1 \pmod{m}.$$

Ja eksistē divas klases  $y_1$  un  $y_2$  tādas, ka

$$xy_1 \equiv xy_2 \equiv 1 \pmod{m},$$

tad

$$x(y_1 - y_2) \equiv 0 \pmod{m}.$$

Reizinot abas puses ar  $y_1$  vai  $y_2$ , iegūsim  $y_1 - y_2 \equiv 0 \pmod{m}$ , tātad  $y_1 \equiv y_2 \pmod{m}$ . Ja eksistē  $y$  tāds, ka  $xy \equiv 1 \pmod{m}$ , tad  $xy - 1 = mq$  un  $xy - mq = 1$ . Reducējot pēc moduļa  $d = LKD(x, m)$ , iegūsim  $0 \equiv 1 \pmod{d}$ , tātad  $d = 1$ .



Par naturāla skaitļa  $n$  *Eilera funkciju*  $\varphi(n)$  sauksim tādu skaitļu  $x$  skaitu, kuriem izpildās nosacījumi  $0 \leq x < n$  un  $LKD(x, n) = 1$ .

**1.1. piezīme.** No iepriekšējās teorēmas seko, ka to atlikuma klašu skaits pēc moduļa  $m$ , kurām eksistē multiplikatīvi inversais elements, ir vienāds ar  $\varphi(m)$ . Šādas atlikumu klases sauksim par *invertējamām pēc moduļa  $m$* . Jebkuru šādu klašu pārstāvju kopu sauksim par *reducētu atlikumu klašu kopu pēc moduļa  $m$* . Kopas  $\mathbb{Z}/m\mathbb{Z}$  multiplikatīvi invertējamo elementu kopu apzīmēsim ar  $(\mathbb{Z}/m\mathbb{Z})^\times$  vai  $U_m$ .

**1.1. piemērs.**  $\varphi(p) = p - 1$ , jo visi skaitļi kopā  $\{1, \dots, p - 1\}$  ir savstarpēji pirmskaitļi ar  $p$  un  $LKD(0, p) = p$ .

$$\varphi(4) = |\{1, 3\}| = 2.$$

$$3^{-1} \equiv 3.$$

$$\varphi(6) = |\{1, 5\}| = 2.$$

$$5^{-1} \equiv 5.$$

$$\varphi(8) = |\{1, 3, 5, 7\}| = 4.$$

$$3^{-1} \equiv 3. \quad 5^{-1} \equiv 5. \quad 7^{-1} \equiv 7.$$

$$\varphi(9) = |\{1, 2, 4, 5, 7, 8\}| = 6.$$

$$2^{-1} \equiv 5. \quad 5^{-1} \equiv 2. \quad 4^{-1} \equiv 7. \quad 7^{-1} \equiv 4. \quad 8^{-1} \equiv 8.$$



## 2. Modulārās aritmētikas pielietojumi

### 2.1. Pozicionālais pieraksts

Senajos laikos cilvēki izmantoja primitīvu skaitļu pierakstu, kas pēc būtības ir līdzīgs svītriņu vilkšanai (*nepozicionālās sistēmas*), piemēram:

- viena svītriņa - vieninieks vai viens objekts,
- pārsvītrotā svītriņa (X) - desmitnieks vai desmit objekti,
- īpaši simboli (hieroglifiskajās sistēmās), kas apzīmē 100 u.t.t.
- burti (alfabētiskās sistēmas senajā Grieķijā un Izraēlā)

Šādā pierakstā simbola vietai nav lielas nozīmes. Parasti simboli tika sakārtoti noteiktā kārtībā, piemēram, lielākā svāra simboli atradās pieraksta sākumā.

Problēmas - ar šādu pierakstu grūti veikt aritmētiskās operācijas.

Būtiskas izmaiņas notika tad, kad cilvēki sāka pierakstīt skaitļus tā, lai simbola atrašanās vietai būtu lielāka nozīme - *pozicionālajās sistēmās*. Tāds pieraksts tika ieviests Indijā ap 500 AD. Viduslaikos tas tika pārņemts Eiropā un tiek izmantots līdz pat mūsu dienām.

**2.1. teorēma.** Ja  $m > 1$  ir vesels skaitlis, tad jebkurš naturāls skaitlis  $n$  ir viennozīmīgi izsakāms formā

$$n = \sum_{i=0}^k a_i m^i,$$

kur  $a_k \neq 0$  un katram  $i$  izpildās nosacījums

$$0 \leq a_i < m.$$

PIERĀDĪJUMS Aprakstīsim algoritmu, ar kura palīdzību var atrast skaitļus  $a_i$ :

1. Izdalīsim  $n$  ar  $m$ :

$$n = q_1 m + a_0;$$

2. Izdalīsim  $q_1$  ar  $m$ :

$$q_1 = q_2m + a_1,$$

ievērosim, ka

$$n = q_1m + a_0 = (q_2m + a_1)m + a_0 = q_2m^2 + a_1m + a_0;$$

3. Izdalīsim  $q_2$  ar  $m$ :

$$q_2 = q_3m + a_2,$$

ievērosim, ka

$$\begin{aligned} n &= q_2m^2 + a_1m + a_0 = \\ &= (q_3m + a_2)m^2 + a_1m + a_0 = \\ &= q_3m^3 + a_2m^2 + a_1m + a_0; \end{aligned}$$

... ..

Algoritms tiek uzskatīts par pabeigtu, kad kārtējais dalījums ir vienāds ar 0 - pēdējais nenulles atlikums ir  $a_k$ . Ievērosim, ka algoritma izpilde vienmēr apstājas, jo dalījumu virkne  $q_1, q_2, \dots$  ir stingri dilstoša.

Algoritma izpildes rezultātā iegūsim skaitļu virkni  $(a_0, a_1, \dots, a_k)$ , kas apmierina vienādību

$$n = a_k m^k + a_{k-1} m^{k-1} + \dots + a_2 m^2 + a_1 m + a_0,$$

tātad skaitļu virkne, kas ir deklarēta teorēmas apgalvojumā, eksistē.

Pierādīsim šādas skaitļu virknes  $(a_0, a_1, \dots, a_k)$  vienīgumu. Pieņemsim, ka eksistē divi izvīzījumi

$$\begin{aligned} n = a_k m^k + a_{k-1} m^{k-1} + \dots + a_2 m^2 + a_1 m + a_0 = \\ b_k m^k + b_{k-1} m^{k-1} + \dots + b_2 m^2 + b_1 m + b_0 \end{aligned}$$

un sāksim salīdzināt skaitļus  $a_i$  un  $b_i$  sākot no  $i = 0$ :

1. Reducēsim  $n$  pēc moduļa  $m$ :  $n \equiv a_0 \equiv b_0 \pmod{m}$ , tāpēc

$$a_0 = b_0,$$

2. Reducēsim  $\frac{n-a_0}{m}$  pēc moduļa  $m$ :

$$\frac{n-a_0}{m} = a_k m^{k-1} + \dots + a_2 m + a_1 \equiv a_1 \equiv b_k m^{k-1} + \dots + b_2 m + b_1 \equiv b_1 \pmod{m},$$

tāpēc

$$a_1 = b_1,$$

3. Reducēsim  $\frac{n-a_0-a_1 m}{m^2}$  pēc moduļa  $m$ :

$$\frac{n-a_0-a_1 m}{m^2} = a_k m^{k-2} + \dots + a_3 m + a_2 \equiv a_2 \equiv b_k m^{k-2} + \dots + b_3 m + b_2 \equiv b_2 \pmod{m},$$

tāpēc

$$a_2 = b_2,$$



**2.1. piezīme.** Skaitļa izvirzījumu  $m$  pakāpju lineārās kombinācijas veidā saucim par skaitļa  $m$ -āro *pozicionālo pierakstu* (vai par  $m$ -*adisko pierakstu*) un apzīmēsim ar  $\overline{a_k a_{k-1} \dots a_0}_m$  vai kādā vienkāršākā veidā, ja nav riska pārprast pierakstu. Pēc noklusēšanas pieņemsim  $\overline{a_k a_{k-1} \dots a_0} = \overline{a_k a_{k-1} \dots a_0}_{10}$ . Skaitli  $m$  saucim par pieraksta *bāzi*.

**2.2. piezīme.** Cilvēki vienmēr strādā ar decimālo pierakstu ( $m = 10$ ), arī ciparu skaits ir saskaņots ar šo  $m$  vērtību. Virspusējs iemesls - cilvēkiem ir 10 pirkstu uz rokām, kaut arī nevar izslēgt dziļāku, pagaidām nezināmu pamatojumu. Plašāk pielietotie pieraksti datorzinātnēs un datortehnoloģijās -

- ja  $m = 2$ , tad pierakstu sauc par *bināro* pierakstu, simbolus 0, 1 sauc par *bitiem*,
- ja  $m = 8$ , tad par *oktālo* pierakstu,
- ja  $m = 16$  (ar cipariem 0,1,2,3,4,5,6,7,8,9,  $A = 10, B = 11, C = 12, D = 13, E = 14, F = 15$ ) - tad par *heksadecimālo* pierakstu.

Dažreiz izmanto arī *ternārās* sistēmas ( $m = 3$ ).

**2.3. piezīme.** Datortehnoloģijās izmanto arī jauktus pierakstus. Piemērs - *decimālais pieraksts ar bināri kodētiem cipariem (binary coded decimal)*- izmanto decimālo pierakstu, kurā katrs cipars tiek kodēts binārajā pierakstā ar četriem cipariem. Šādā pierakstā

$$354 = 0011, 0101, 0010.$$

**2.4. piezīme.** No binārā pieraksta seko šāds neacīmredzams fakts - katru naturālu skaitli var viennozīmīgi izteikt kā 2 pakāpju summu.

**2.5. piezīme.** Algoritms skaitļa  $n$  pārveidošanai no decimālās sistēmas uz  $m$ -āro sistēmu:

1. izdalīt  $n$  ar  $m$ :  $n \rightarrow (q_1, a_0)$ , kur  $n = q_1m + a_0$ , ja  $q_1 \neq 0$ , tad iet tālāk;
2. izdalīt  $q_1$  ar  $m$ :  $(q_1, a_0) \rightarrow (q_2, a_1)$ , kur  $q_1 = q_2m + a_1$ , ja  $q_2 \neq 0$ , tad iet tālāk;
3. izdalīt  $q_2$  ar  $m$ :  $(q_2, a_1) \rightarrow (q_3, a_2)$ , kur  $q_2 = q_3m + a_2$ , ja  $q_3 \neq 0$ , tad iet tālāk;
- ... izdalīt ...

k+1. Uzrakstīt simbolus pareizā kārtībā -  $\overline{a_k a_{k-1} \dots a_0}_m$ ;

k+2. Veikt pārbaudi:  $a_k m^k + a_{k-1} m^{k-1} + \dots + a_0 \stackrel{?}{=} n$ .



**2.1. piemērs.** Pārveidosim skaitli 2007 5-ārajā pierakstā:

1.  $2007 = 401 \cdot 5 + 2 \rightarrow a_0 = 2, q_1 = 401;$
2.  $401 = 80 \cdot 5 + 1 \rightarrow a_1 = 1, q_2 = 80;$
3.  $80 = 16 \cdot 5 + 0 \rightarrow a_2 = 0, q_3 = 16;$
4.  $16 = 3 \cdot 5 + 1 \rightarrow a_3 = 1, q_4 = 3;$
5.  $3 = 0 \cdot 5 + 3 \rightarrow a_4 = 3, q_5 = 0;$
6. Pierakstām rezultātu  $2007 = \overline{31012}_5;$
7. Veicam pārbaudi:  $3 \cdot 5^4 + 1 \cdot 5^3 + 0 \cdot 5^2 + 1 \cdot 5^1 + 2 = 1875 + 125 + 5 + 2 = 2007.$

**2.6. piezīme.** Algoritms skaitļa  $n$  pārveidošanai no  $m$ -ārās sistēmas uz decimālo sistēmu:

1. Ja ir dots skaitlis  $n = \overline{a_k a_{k-1} \dots a_0}_m$ , aprēķināt decimālajā pierakstā summu

$$n = a_k m^k + a_{k-1} m^{k-1} + \dots + a_0.$$

**2.2. piemērs.** Ja skaitlis 7-ārajā pierakstā ir  $\overline{3621}_7$ , tad decimālajā pierakstā tas ir  $3 \cdot 7^3 + 6 \cdot 7^2 + 6 \cdot 7^1 + 1 = 1338$ .

**2.7. piezīme.** Algoritms skaitļa  $n$  pārveidošanai no  $m_1$ -ārās sistēmas uz  $m_2$ -āro sistēmu:

1. Pārveidot skaitli  $n$  no  $m_1$ -ārā pieraksta uz decimālo pierakstu,
2. Pārveidot skaitli  $n$  no decimālā pieraksta uz  $m_2$ -āro pierakstu.

**2.3. piemērs.** Pārveidosim skaitli  $\overline{3621}_7$  uz heksadecimālo pierakstu:

$$\overline{3621}_7 \rightarrow 1338 \rightarrow \overline{53A}_{16}$$

**2.8. piezīme.** Pozicionālās sistēmas plusi:

- simbolu ekonomija,
- ērti veikt aritmētiskās operācijas - algoritmi visiem ir zināmi, tos var vispārināt no  $m = 10$  uz jebkuru  $m$  vērtību.

## 2.2. teorēma.

1. Maksimālais naturālais skaitlis, ko var ierakstīt  $m$ -ārajā sistēmā ar  $k$  simboliem ir vienāds ar  $m^{k+1} - 1$ .
2. Lai skaitli  $n$  ierakstītu  $m$ -ārajā sistēmā, ir nepieciešami

$$k_n = [\log_m n] + 1$$

simboli.

PIERĀDĪJUMS 1. Lielākais skaitlis ar  $k$  simboliem  $m$ -ārajā pierakstā ir

$$\begin{aligned} \overline{(m-1)(m-1)\dots(m-1)}_m &= (m-1)(1+m+\dots+m^k) = \\ &= (m-1)\frac{m^{k+1}-1}{m-1} = m^{k+1}-1. \end{aligned}$$

2.  $n = m^{\log_m n}$ , tāpēc

$$m^{[\log_m n]} \leq n < m^{[\log_m n]+1}.$$

$m$ -ārajā pierakstā skaitlis  $m^{\lceil \log_m n \rceil}$  ir  $\underbrace{1000\dots000}_{[\log_m n] \text{ reizes}}$  ( $[\log_m n] + 1$  cipari)  
 un  $m^{\lceil \log_m n \rceil + 1}$  ir  $\underbrace{1000\dots000}_{[\log_m n] + 1 \text{ reizes}}$  ( $[\log_m n] + 2$  cipari). Redzam, ka skaitli  
 $n$  var pierakstīt ar  $[\log_m n] + 1$  cipariem. ■

## 2.9. piezīme. Programmēšanas uzdevumi:

- (Automātiskā ieskaite) Uzrakstīt programmu, kas veic aritmētiskās darbības ar naturālajiem skaitļiem romiešu skaitļu sistēmā. Sākuma un beigu dati - skaitļi romiešu pierakstā, kas nepārsniedz 10000.
- (Automātiskā ieskaite) Uzrakstīt programmu, kas veic aritmētiskās darbības  $m$ -ārajā sistēmā. Sākuma un beigu dati -  $m$  un skaitļi  $m$ -ārajā pierakstā.

## 2.2. Informācijas glabāšana - hashing

Lielus informācijas daudzumus labāk ir glabāt sakārtotā veidā - lai varētu ātrāk atrast vajadzīgo informāciju.

Viens no informācijas glabāšanas veidiem - sašķirot informāciju noteiktā veidā, saliekot to *konteineros*. Šādā gadījumā, lai atrastu vajadzīgo informāciju, pietiek pārmeklēt vienu konteineru, nevis visu informācijas kopumu.

Viens no informācijas šķirošanas veidiem - *hešhings (hashing)* jeb hash-funkcijas izmantošana.

Metodes būtība:

- tiek definēta funkcija (hash-funkcija), kas katrai informācijas vienībai (skaitlim, vārdam, simbolu virknei u.c.) piekārto sakārtotas un bieži vien mazākas kopas (atslēgu kopas) elementus,
- katrai informācijas vienībai tiek pielietota hash-funkcija un informācija tiek glabāta konteineros (failos vai cita veida OS atkarīgās struktūrās) atbilstoši atslēgas vērtībai,

- lai atrastu doto informācijas vienību, pietiek pārmeklēt tikai to konteineru, kas atbilst tās atslēgas vērtībai.

Hash-funkciju piemēri:

- vārda pirmais burts,
- $f(x) = [xm]$ , ja  $0 \leq x \leq 1$ ,
- $f(x) = x \pmod{m}$ ,

## 2.3. Informācijas drošība - paritātes un cita veida kongruenču pārbaude

Ierakstot un pārraidot informāciju var rasties kļūdas.

Lai garantētu informācijas saglabāšanos, sākotnējai ierakstāmajai vai pārraidāmajai informācijai pievieno papildus informāciju, kuru izmanto pārbaudē - salīdzina ierakstīto papildus informāciju ar aprēķināto:

- sūtītājs fiksē pārraidāmo informāciju  $A$ ,
- sūtītājs aprēķina pārbaudošo papildus informāciju  $f(A)$  (parasti  $f(A)$  pēc apjoma ir daudz mazāka kā  $A$ ),
- sūtītājs pārraida vai ieraksta pāri  $(A, f(A))$ ,
- saņēmējs saņem informāciju  $(A', B')$  (iespējamās kļūdas),
- saņēmējs aprēķina  $f(A')$ , ja  $B' \neq f(A')$ , tad konstatē, ka notikusi pārraides vai ierakstīšanas kļūda.

Par papildus informācijas ģenerēšanai bieži izmanto modulāro aritmētiku.

Vienkāršākie piemēri:

- paritātes pārbaude - informācijas vienībai tiek pievienots paritātes bits (0 vai 1) atkarībā no tā, vai dotā informācijas vienība, iekodēta kā vesels skaitlis, ir pāra vai nepāra skaitlis, šī metode ļauj noteikt, ka ir kļūda vienā bitā;
- 10 ciparu ISBN numurs - informācija tiek iekodēta pirmajos 9 ciparos un pēdējais tiek aprēķināts pēc formulas

$$1 \cdot a_1 + 2 \cdot a_2 + \dots + 9 \cdot a_9 \pmod{11}$$



## 2.4. Nejaušo skaitļu ģenerēšana

Nejaušo skaitļu ģenerēšana - svarīga operācija. Nejaušie skaitļi tiek izmantoti *varbūtiskajos algoritmos* u.c.

Vienkāršs nejaušo skaitļu ģenerators:

$$x_{n+1} \equiv ax_n + b \pmod{m},$$

kur  $LKD(b, m) = 1$ ,  $x_0$  - sēkla (*seed*).

**2.4. piemērs.** Ja  $m = 13$ ,  $a = 2$ ,  $b = 4$ ,  $x_0 = 5$ , tad iegūsim virkni

5, 1, 6, 3, 10, 11, 0, 4, 12, 2, 8, 7, 5.

## 2.5. Aritmētisko operāciju pārbaude

Aritmētisko operāciju rezultātu pareizības pārbaudē var izmantot vienu no modulārās aritmētikas īpašībām: ja  $a = b$ , tad  $a \equiv b \pmod{m}$  visiem naturāliem  $m$ . Pretējais apgalvojums: ja eksistē  $m$  tāds, ka  $a \not\equiv b \pmod{m}$ , tad  $a \neq b$ .

Pārbaudes algoritms:

1. Atrodam operācijas rezultātu  $c = a \star b$ ,
2. Atrodam  $c' = a \star b \pmod{m}$  un  $c'' = c \pmod{m}$ ),
3. Ja  $c' \neq c''$ , tad konstatējam kļūdu.

## 2.6. Dalāmības pazīmes

*Dalāmības ar  $m$  pazīme* - īpašība, kas piemīt  $m$  daudzkārtņu cipariem (parasti 10-ārajā pierakstā). Mūsdienās - zināmā mērā anahronisms, jo ar datoru parādīšanos dalāmība tiek pārbaudīta ar datorprogrammu palīdzību.

Šajā sadaļā pieņemam, ka

$$n = \overline{a_k a_{k-1} \dots a_0} = a_k \cdot 10^k + \dots + a_1 \cdot 10 + a_0.$$

### 2.6.1. Dalāmība ar 2

Tā kā  $10^l = 2^l \cdot 5^l \equiv 0 \pmod{2}$ , ja  $l \geq 1$ , tad

$$n \equiv a \cdot 0 + a_{k-1} \cdot 0 + \dots + a_0 \equiv a_0 \pmod{2}.$$

*Dalāmības pazīme ar 2:*  $2|n$  tad un tikai tad, ja  $2|a_0$  (ja  $n$  pēdējais cipars dalās ar 2 - pieder kopai  $\{0, 2, 4, 6, 8\}$ ).

### 2.6.2. Dalāmība ar 3

Tā kā  $10^l \equiv 1 \pmod{3}$ , tad

$$n \equiv a_k \cdot 1 + a_{k-1} \cdot 1 + \dots + a_0 \equiv a_k + a_{k-1} + \dots + a_0 \pmod{3}.$$

*Dalāmības pazīme ar 3:*  $3|n$  tad un tikai tad, ja  $3|a_k + a_{k-1} + \dots + a_0$  (ja  $n$  ciparu summa dalās ar 3).

### 2.6.3. Dalāmība ar 4 un vispārināšana uz $2^l$

Tā kā  $10^j = 2^j \cdot 5^j \equiv 0 \pmod{2^l}$ , ja  $j \geq l$ , tad

$$n \equiv a_{l-1} \cdot 10^{l-1} + a_{l-1} \cdot 1 + \dots + a_0 = \overline{a_{l-1}a_{l-2}\dots a_0} \pmod{2^l}.$$

*Dalāmības pazīme ar  $2^l$ :*  $2^l|n$  tad un tikai tad, ja  $2^l|\overline{a_{l-1}a_{l-2}\dots a_0}$  (ja pēdējo  $l$  ciparu veidotais skaitlis dalās ar  $2^l$ ).

### 2.6.4. Dalāmība ar 5 un vispārināšana uz $5^k$

Tā kā  $10^j = 2^j \cdot 5^j \equiv 0 \pmod{5^l}$ , ja  $j \geq l$ , tad

$$n \equiv a_{l-1} \cdot 10^{l-1} + a_{l-1} \cdot 1 + \dots + a_0 = \overline{a_{l-1}a_{l-2}\dots a_0} \pmod{5^l}.$$

*Dalāmības pazīme ar  $5^l$ :  $5^l|n$  tad un tikai tad, ja  $2|\overline{a_{l-1}a_{l-2}\dots a_0}$  (ja pēdējo  $l$  ciparu veidotais skaitlis dalās ar  $5^l$ ).*

### 2.6.5. Dalāmība ar 6

$6 = 2 \cdot 3$  un  $LKD(2, 3) = 1$ , tāpēc  $6|n$  tad un tikai tad, ja  $2|n$  un  $3|n$ .

*Dalāmības pazīme ar 6:  $6|n$  tad un tikai tad, ja  $n$  pēdējais cipars ir pāra skaitlis un  $n$  ciparu summa dalās ar 3.*

### 2.6.6. Dalāmība ar 9

Tā kā  $10^l \equiv 1 \pmod{9}$ , tad

$$n \equiv a_k \cdot 1 + a_{k-1} \cdot 1 + \dots + a_0 \equiv a_k + a_{k-1} + \dots + a_0 \pmod{9}.$$

*Dalāmības pazīme ar 9:*  $9|n$  tad un tikai tad, ja  $9|a_k + a_{k-1} + \dots + a_0$  (ja  $n$  ciparu summa dalās ar 9).

### 2.6.7. Dalāmība ar 11

Tā kā  $10 \equiv -1 \pmod{11}$ , tad

$$10^{2j} \equiv (-1)^{2j} \equiv 1 \pmod{11}$$

un

$$10^{2j+1} \equiv (-1)^{2j+1} \equiv -1 \pmod{11}.$$

Redzam, ka

$$n \equiv a_k(-1)^k + \dots + a_2 - a_1 + a_0 \pmod{11}.$$

*Dalāmības pazīme ar 11:*  $11|n$  tad un tikai tad, ja  $11|a_0 - a_1 + a_2 + \dots + a_k(-1)^k$  (ja  $n$  ciparu alternējoša summa dalās ar 11).

### 3. 5.mājasdarbs

- 5.1 Atlikumu gredzeniem pēc moduļa  $p$  vienāda ar 5,7 un 11 atrodiet mazāko  $k$  tādu, ka  $2^k \equiv 1 \pmod{p}$ .
- 5.2 Skaitli 2007 pārveidojiet šādos pozicionālajos pierakstos:
- a) binārajā,
  - b) oktālajā,
  - c) heksadecimālajā,
  - d) ternārajā.
- 5.3 Atrodiet dalāmības pazīmi ar 12.