Aibių teorija.

1. Aibės apibrėžimas, aprašymo būdai.

Aibė - tai objektų, kuriems būdingas tam tikras požymis, visuma.

Aibės savybės:

- Visi aibės elementai yra skirtingi
- Aibės dažniausiai žymimos didžiosiomis lotyniškomis raidėmis
- Matematikoje nusistovėjusi simbolika:
  - N - natūraliųjų skaičių aibė,
  - Z - sveikųjų skaičių aibė,
  - Q - racionaliųjų skaičių aibė,
  - R - realiųjų skaičių aibė,
  - C - kompleksinių skaičių aibė;
  - ∅ - tuščioji aibė.

Aprašymo būdai:

- išvardinimo - aibė aprašoma išvardinant visus jos elementus. Šis būdas naudojamas kai aibės elementų kiekis yra nedidelis arba yra aiškus elementų dėsningumas.
- aprašymo - aprašant aibę šiuo būdu taip pat nurodomos savybės, kurias turi tenkinti kiekvienas aibės elementas. Savybės nurodomos logine funkcija, vadinama predikatu[1].
- grafinis - šiuo būdu vaizduoti aibes naudojamos Veno diagramos, jos paprastai naudojamos vaizdžiai parodyti santykius bei veiksmus su aibėmis.

  Pvz.:

2. Veiksmai su aibėmis. Veno diagramos, jų panaudojimas.

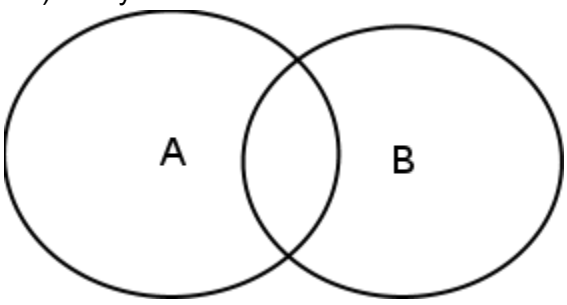
Veiksmai su aibėmis:

- Sąjunga $C = A \cup B = \{c \mid c \in A \cup c \in B\}$
- Sankirta $C = A \cap B = \{c \mid c \in A \cap c \in B\}$
- Skirtumas $C = A - B = A \setminus B = \{c \mid c \in A \cap c [?][?][?] B\}$
- Suma moduliu 2(simetrinis skirtumas): $A \oplus B = (A \setminus B) \cup (B \setminus A)$
- Papildymas $\overline{A} = U \setminus A$

[1] Predikatas - tai funkcija, kurios apibrėžimo sritis yra aibė M ir kuri kiekvienam aibės M elementui priskiria reikšmę “tiesa” arba “melas”, Kitaip tariant, tai funkcija, atvaizduojanti aibę M į aibė {tiesa,melas} ({true, false},{1,0}).

- Dekarto sandauga: $C = A \times B = \{(a,b) \mid a \in A \cap b \in B\}$

Veno diagrama - tai aibė plokštumos taškų, apribotų uždarąja kreive(paprastai apskritimu). Pavyzdžiui:



Universalioji aibė (visų dominančių elementų aibė) vaizduojama stačiakampiu.
Naudojimas:

Veno diagramos naudojamos atvaizduoti ryšius ir veiksmus tarp aibių grafiškai.

3.1. Santykiai

Aibių A ir B santykiu(binariniu sąryšiu) R vadinamas Dekarto sandaugos A×B poaibis: $R \subseteq (A \times B)$

Santykių rūšys:

1. **Refleksyvus**
   - Santykį vadiname refleksyviu, jei $\forall\, a \in A{:}aRa$
   - Santykis yra refleksyvus tada ir tik tada, kai jo matricos pagrindinėje įstrižainėje vienetai.
   - *pvz*: Sąryšis R = {(2, 2), (4, 4), (7, 7), (4, 2), (2, 7)} aibėje A = {2, 4, 7}.
2. **Antirefleksyvus**
   - Santykį vadiname antirefleksyviu, jei $\forall\, a \in A{:}\neg(aRa)$
   - Santykis yra antisimetrinis tada ir tik tada, kai jo matricos pagrindinėje įstrižainėje nuliai.
   - *pvz*: Sąryšis R = {(2, 4), (2, 7), (4, 7)} aibėje A = {2, 4, 7}
3. **Simetrinis**
   - Santykį vadiname simetriniu, jei $\forall(a,b \in A; a \neq b){:}(aRb \Rightarrow bRa)$
   - Santykis yra simetrinis tada ir tik tada, kai jo matrica M: $\forall\, i,j \in [?][?][?]{:}m_{ij} = m_{ji}$ (ant pagrindinės įstrižainės gali būti betkas).
   - *pvz*: Sąryšis R = {(2, 4), (2, 7), (4, 7)} aibėje A = {2, 4, 7}
4. **Antisimetrinis**
   - Santyname kį vadiantisimetriniu, jei $\forall(a,b \in A; a \neq b){:}(aRb \Rightarrow \neg(bRa))$
   - Santykis yra refleksyvus tada ir tik tada, kai jo matrica M: $\forall\, i,j \in [?][?][?]{:}(i \neq j \Rightarrow m_{ij} \neq m_{ji})$ (ant pagrindinės įstrižainės gali būti betkas).
   - *pvz*: Sąryšis R = {(2, 2), (2, 4), (4, 4), (7, 2)} aibėje A = {2, 4, 7}
5. **Tranzityvus**

- Santykį vadiname tranzityviu, jei $\forall a,b,c \in A: (aRb \cap bRc \Rightarrow aRc)$
- *pvz*: Sąryšis “daugiau” sveikųjų skaičių aibėje.

6. **Ekvivalentumo**
   - Refleksyvų, simetrinį ir tranzityvų sąryšį vadiname ekvivalentumo santykiu.
   - Sąvybė: Jei aibėje A apibrėžtas ekvivalentumo sąryšis R, tai aibę A galima suskaidyti į netuščius nesikertančius poaibius(ekvivalentumo klases) tokiu būdu, kad tame pačiame poaibyje esantys elementai yra susieti sąryšiu R, o esantys skirtinguose — nėra susieti.
   - Betkoks ekvivalentumo sąryšis kartu ekvivalentus sąryšiui “priklausyti tam tikram poaibiui”
   - *pvz*: Sąryšis “gyventi tame pačiame mieste”.
7. **Negriežtos(dalinės) tvarkos**
   - Refleksyvų, antisimetrinį ir tranzityvų sąryšį vadiname negriežtos tvarkos santykiu.
   - *pvz*: Sąryšis “būti ne didesniu” ($\leq$) natūraliųjų skaičių aibėje.
8. **Griežtos tvarkos**
   - Antirefleksyvų, antisimetrinį ir tranzityvų sąryšį vadiname griežtos tvarkos sąryšiu.
   - *pvz*: Sąryšis “būti viršininku” žmonių aibėje
   - *pvz*: Sąryšis “būti didesniu” (>) natūraliųjų skaičių aibėje

3.2. Santykiai.

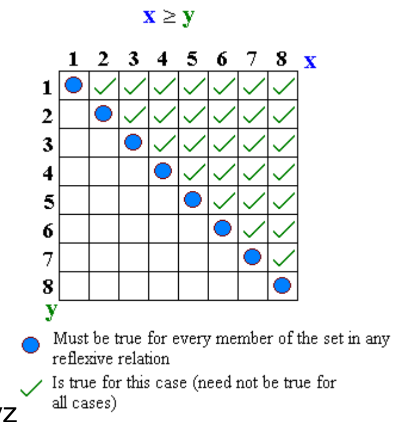
Aibių A ir B santykiu(binariniu sąryšiu) R vadinamas Dekarto sandaugos A$^{\times}$B poaibis:
$R \subseteq (A \times B)$

Santykis užrašomas:
$aRb,\ a \in A, b \in B$

Pavyzdžiui taip aprašoma „Daugiau“ natūraliųjų skaičių aibėje:
$>: R \subseteq (N \times N)$ t.y:
$R = \{(2,1),(3,1),(3,2),\dots\}$

Refleksyvus santykis
$aRa,\ \forall a \in A$
Kiekvienas elementas susietas su savimi
Tokio sąryšio grafo elementai turi turėti kilpas, gali turėti ir lankus (ryšius su kitais elementais)

Tokio sąryšio matricos įstrižainėje turi būti visi vienetai pvz

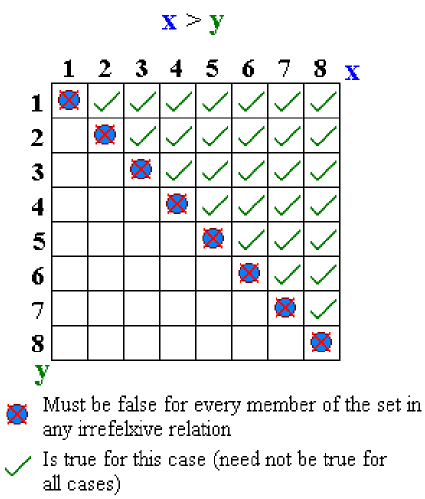


\~~~~~~

Antirefleksyvusis santykis

$\neg(aRa), \forall\, a \in A$

Nei vienas elementas nėra susijęs su savimi.
Tokio sąryšio grafas negali turėti kilpų.
Matricos įstrižainėje negali būti vienetų.

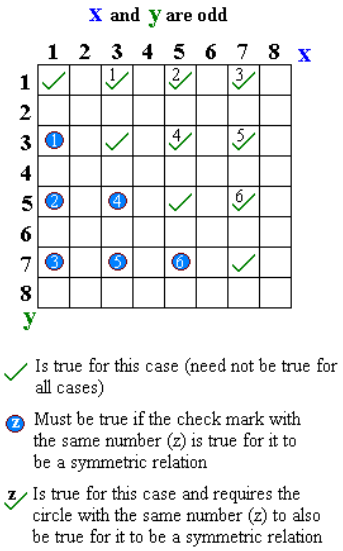


\~~~~~~

Simetriškumo santykis

$aRb \rightarrow bRa, \forall\, a,b \in A$

Jeigu kiekvienas a susietas su b, ir kiekvienas b susietas su a gauname simetriją.



~~~~~~

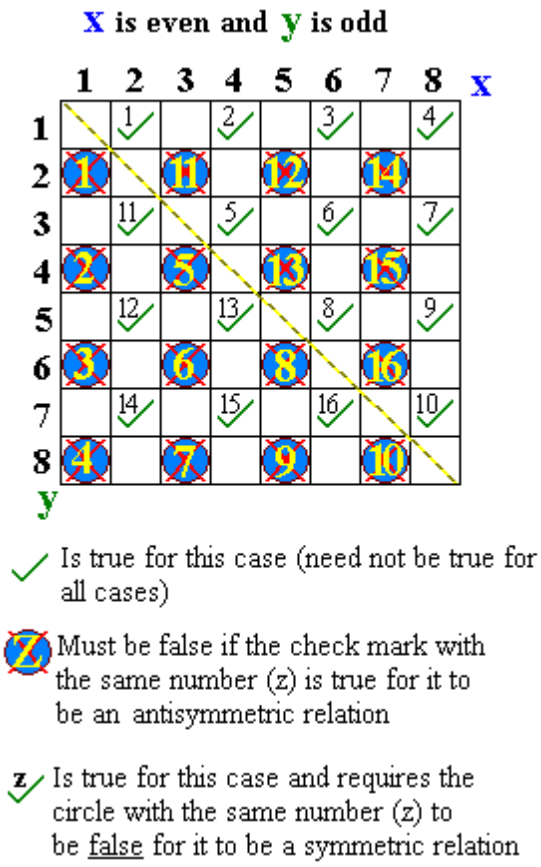
Antisimetriškumo santykis

(Antisimetriškas nėra tas pats kas asimetriškas, nes santykį galima laikyti asimetrišku jau tada, kai jis neatitinka simetriškumo sąlygų.)

$$aRb \wedge bRa \rightarrow a = b, \ \forall \ a,b \in A$$

Kai nei vienas elementas nėra susietas su savimi turime antisimetriškumo santykį.

Pavyzdžiui lyginiai ir nelyginiai skaičiai. X aibė - lyginiai skaičiai, Y - nelyginiai.
~~~~~~



~~~~~~

Pilnasis sąryšis

sąryšis $S \subset A^2$yra vadinamas pilnuoju, kai

$S \cup S^{-1} \cup I_a = U_a = A^2$, kur U_a- universalioji aibė.

Arba kitaip bet kurie du skirtingi aibės elementai a ir b turi bent po vieną ryšį.

~~~~~~

Universalusis sąryšis

U_A

Matrica sudaryta vien iš vienetų

~~~~~~

Tranzityvusis sąryšis

$aRb \wedge bRc \rightarrow aRc, \ \forall \ a,b,c \in A$

Kuomet *a* elementas susietas su *b*, o *b* su *c*. Pavyzdžiui:

kai A>B ir B > C, tuomet visada A > C

kai A$\geq$B ir B $\geq$C, tuomet visada A $\geq$ C

kai A = B ir B = C, tuomet visada A = C

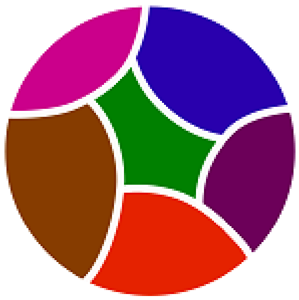
~~~~~~

Ekvivalentiškumo sąryšis

Tai sąryšis kuris yra:

* refleksyvus
* simetrinis
* tranzityvus

Ekvivalentaus sąryšio aibė yra suskirstyta į atskiras sritis ir kiekvienas aibės elementas yra kažkokios srities narys. Du elementai yra ekvivalentūs jeigu yra toje pačioje srityje. Sričių sankirta yra tuščia aibė. Sričių sąjunga yra visa aibė.



~~~~~~

Dalinės tvarkos sąryšis

Šis sąryšis yra:
* Refleksyvus
* Antisimetriškas
* Tranzityvus

~~~~~~

4. Pagrindiniai dėsniai, taikomi veiksmams su aibėmis.

1. Idempotentiškumo:
   a. $A \cap A = A$
   b. $A \cup A = A$
2. Komutatyvumo:
   a. $A \cup B = B \cup A$
   b. $A \cap B = B \cap A$
3. Asociatyvumo:
   a. $A \cup (B \cup C) = (A \cup B) \cup C$
   b. $A \cap (B \cap C) = (A \cap B) \cap C$
4. Distributyvumo:
   a. $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
   b. $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
5. Padengimo:
   a. $(A \cap B) \cup A = A$
   b. $(A \cup B) \cap A = A$
6. De Morgano:

   a. $\overline{(A \cap B)} = \overline{A} \cup \overline{B}$
   b. $\overline{(A \cup B)} = \overline{A} \cap \overline{B}$
7. Apibendrintieji De Morgano dėsniai:
   a. $A \setminus (B \cap C) = (A \setminus B) \cup (A \setminus C)$
   b. $A \setminus (B \cup C) = (A \setminus B) \cap (A \setminus C)$
8. Dvigubo neigimo (dvigubo papildymo):
   a. $\overline{\overline{A}} = A$
9. Universalioji aibė (U)
   a. $A \cup \overline{A} = U$
   b. $A \cap U = A$
   c. $A \cap \overline{A} = [?][?][?]$
10. Tuščiosios aibės:
   a. $[?][?][?] \cup A = A$
   b. $[?][?][?] \cap A = [?][?][?]$
11.
   a. $\overline{U} = [?][?][?]$
   b. $\overline{[?][?][?]} = U$

Kodavimo teorija

1. Kodavimo sistemos.
   - Pozicinės skaičių kodavimo sistemos:
     - Dvejetainė
     - Aštuntainė
     - Dešimtainė
     - Šešioliktainė
     - BCD (Binary Coded Decimal)
     - +3
   - Nepozicinės skaičių kodavimo sistemos:
     - Romėniški skaičiai
   - Simbolių kodavimo sistemos:
     - Morzės abecelė
     - Telegrafo kodai
     - ASCII
     - Unicode
     - Cezario kodas
   - Grafinių objektų kodavimo sistemos:
     - Dekarto
     - Sferinė
   - Grafinių duomenų kodavimas kompiuteryje:
     - Vektorinis būdas.
       Grafiniai objektai aprašomi juos sudarančiomis linijomis.
     - Rastrinis būdas (matricinis, pilno užpildymo)
       Šiuo būdu paveikslėlis išskaidomas taškais (pikseliais) ir kiekvienas taškas atvaizduojamas tam tikra spalva. Paveikslėlį atitinka taškų matrica.
       Judantis vaizdas yra koduojamas statinių vaizdų seka.
   - Kitos kodavimo sistemos:
     - ISBN - international standard book numbering. Kodą sudaro 9 dešimtainiai skaitmenys, taip pat dešimtasis skaitmuo arba simbolis x. Paskutinė reikšmė apskaičiuojama pagal formulę $a_{10} = (1*a_1 + 2*a_2 + \ldots + 9*a_9) \bmod 11$.
       Rašome “x”, jei liekana yra 10. Paskutinis skaitmuo naudojamas klaidų aptikimui.
     - Asmens kodai sudaromi tokia forma: $LY_1Y_2M_1M_2D_1D_2X_1X_2X_3K$. X - eilės numeris. K - kontrolinis skaitmuo. K apskaičiuojamas panašiai kaip ISBN, jei liekana 10 tada skaičiuojama pagal kitą formulę: $K = (L * 3 + Y_1 * 4 + \ldots + D_2 * 9 + X_1 * 1 + X_2 * 2 + X_3 * 3) \bmod 11$.
     - Brūkšninis kodas.
       EAN-13 - Europos brūkšninio kodo standartas
       UPC-A - Amerikos
       Pirmi 2 skaičiai reiškia šalį, sekantys 5 - gamintojas, tolesni 5 - gaminio kodas, paskutinis skaičius - kontrolinė suma.

Klaidų taisymas/aptikimas:

- Kodai, kurie aptinka klaidas, atsiradusias, dėl perdavimo metu patirtų trikdžių vadinami klaidas aptinkančiais kodais (error detecting codes). Keli kodų pavyzdžiai: kontrolinės sumos (checksum), ciklinis-perteklinis patikrinimas (cyclic redundancy check), maišos funkcijos (cryptographic hash function) etc.
- Kodai, kurie klaidas aptinka ir atstato prarastą informaciją vadinami klaidas taisančiais kodais (error detecting codes).

Kodavimo tikslai:

1. Atvaizdavimo patogumas.
2. Informacijos perdavimo efektyvesnė realizacija
    a. Galimas klaidų aptikimas arba klaidų taisymas.
3. Informacijos slaptumo užtikrinimas.
4. Duomenų kiekio sumažinimui.

2. Grėjaus kodas. Cikliniai / acikliniai kodai. Chemingo atstumas.

- **Grėjaus kodas**

  Grėjaus kodas - kodas, kurio du gretimi poaibiai skiriasi tik vienu elementu.

  Formulė:

  $$C_1 := b_1$$
  $$C_i = b_{i-1} \oplus b_i, \; i = \overline{2,n}$$

  I pvz.

  pavyzdžiui turime dvejetainį skaičių 0101, raskime jo Grėjaus kodą:

  Atsakymas: 0111

  II pvz.

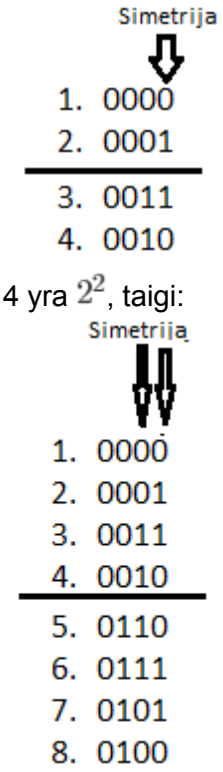
  turime dvejetainį skaičių 0001, raskime jo Grėjaus kodą:

  Atsakymas: 0001

Grėjaus kodas pasižymi „atspindžio“ savybe, t.y už kiekvieno 2^nelemento yra simetrijos ašis, kuria pasinaudojus galima rasti kitus Grėjaus kodus. Pvz:

2 yra 2^1, taigi:



4 yra 2^2, taigi:

- Cikliniai kodai - pirmas ir paskutinis elementai skiriasi viena skiltimi. Nuosekliai didėjant skaičiui, kinta tik vienos skilties turinys. Klasikinis ciklinio kodo pavyzdys – Grėjaus kodas., Didėjant skaičiui Grėjaus kode kinta simbolis tik vienoje šio kodo skiltyje – visuomet pakinta simbolis toje žemiausioje skiltyje, kurioje įvykęs pokytis sukuria naują ,iki tol nebuvusią kodinę kombinaciją.
- Acikliniai kodai - Logiškai, priešingas cikliniam kodui.
- Chemingo atstumas - vadinamas pozicijų (simbolių) skaičius , kuriuo dvi sekos (kodiniai žodžiai) skiriasi viena nuo kitos. Grėjaus kodui Chemingo atstumas lygus 1.

3. Grėjaus kodo savybės ir pritaikymai (BF minimizavimui, derinių generavimui, grafų teorija, kt.)

- Savybės
  Hemingo atstumas = 1;
  Grėjaus kodas yra ciklinis.
- Pritaikymai
  Grėjaus kodą galime panaudoti koduodami baigtinio automato padėtims. Pereinant iš vienos pozicijos į kitą reikės pakeisti vieno trigerio būseną. Kitu atveju, reikėtų pakeisti kelių trigerių būsenas. Jos pasikeičia ne vienu metu, todėl pasikeitus vieno trigerio būsenai dar atsidurtumėme kitoje būsenoje. Tai visai nereikalinga.
  Galime minimizuoti Būlio funkcijas naudojantis Karno diagramomis. Dar nagrinėjamos minimizuojamos BF mintermus galima apjungti, jei tuos mintermus atitinkantys vienetukai yra gretimuose Karno diagramos langeliuose. Karno diagrama konstruojama naudojant Grėjaus kodo kombinacijas.

Generuojant visus įmanomus objektų derinius, kiekvieną derinį galima koduoti 0 ar 1 seka, pvz. objektų a, b, c, d sudaromą derinį {a; c} atitiktų kodas 1010. Kai norime sugeneruoti visus derinius, jei generuojame juos pagal Grėjaus kodus, tai iš vieno derinio kodo į kitą pereiti mums užtenka pakeisti vieną skiltį. Tai pagerina efektyvumą.

Grėjaus kodai gali būti taikomi grafų teorijoje. Jei turime n-matį hiperkubą, galime gauti Hamiltono ciklą (kiekvieną viršūnę aplankyti po vieną kartą ir grįžti į pradžią) tinkamai sunumeravę viršūnes. Pvz, tarkime, kad turime n = 3. Pereinant briauna keičiasi viena skiltis (kinta viena koordinatė). Pereidami viršūnes pagal Grėjaus kodus galime gauti Hamiltono ciklą dėl Grėjaus kodo savybių. Grėjaus kodas yra ciklinis ir jame yra tiek skirtingų kombinacijų, kiek viršūnių hiperkube.

4. Perėjimai tarp skaičiaus atvaizdavimo skirtingais kodais.

“10”	**“2”**	**BCD**	**“16”**	**“+3”**	**Grėjaus**	**Romėn.**
0	0000	0000	0	0011	0000	-
1	0001	0001	1	0100	0001	I
2	0010	0010	2	0101	0011	II
3	0011	0011	3	0110	0010	III
4	0100	0100	4	0111	0110	IV
5	0101	0101	5	1000	0111	V
6	0110	0110	6	1001	0101	VI
7	0111	0111	7	1010	0100	VII
8	1000	1000	8	1011	1100	VIII
9	1001	1001	9	1100	1101	IX
10	1010	1 0000	A	1101	1111	X
11	1011	1 0001	B	1110	1110	XI
12	1100	1 0010	C	1111	1010	XII
13	1101	1 0011	D	10000	1011	XIII
14	1110	1 0100	E	10001	1001	XIV
15	1111	1 0101	F	10010	1000	XV

Perėjimai iš vienos skaičiavimo sistemos į kitą.

Bet kokį realų skaičių galima atvaizduoti tokia formule:

$$Z = \sum_i X_i Y^i .....$$

kur Z yra skaičiaus vertė, X_i yra argumentas, kuris gali kisti nuo 0 iki Y, o Y yra **skaičiavimo sistemos pagrindas**.

Arba tiksliau:

$$Z = ... + X_3 Y^3 + X_2\ Y^2 + X_1 Y^1 + X_0\ Y^1\ + X_{-1} Y^{-1} + ...$$

Pavyzdžiui:

Trupmeninį **dešimtainį** skaičių 445,5 kurio pagrindas Y=10 galėtume atvaizduoti pagal formulę taip:

$$Z = 4 \cdot 10^2 + 4 \cdot 10^1 + 5 \cdot 10^0 + 5 \cdot 10^{-1}$$

Dvejetainėje skaičiavimo sistemoje skaičiai turi tik dvi reikšmes (0 ir 1). Y=2

Pabandykime dvejetainį skaičių paversti į dešimtainį:

Dvejetainiam skaičiui 101010 X vertės bus tokios:

$$X_5 = 1, X_4 = 0, X_3 = 1, X_2 = 0, X_1 = 1, X_0 = 0$$

Tada:

$$Z = 1 * 2^5 + 0 * 2^4 + 1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 0 * 2^0 = 32 + 8 + 2 = 42_{10}$$

Šešioliktainėje skaičiavimo sistemoje Y = 16

paverskime 11AB į dešimtainį skaičių.

$$Z = 1 \cdot 16^3 + 1 \cdot 16^2 + 10 \cdot 16^1 + 11 \cdot 16^0 = 4523$$

Pakeitimas iš dešimtainės į dvejetainę skaičiavimo sistemą

Pakeitimui yra naudojami atimties ir dalybos metodai. Atimties metode yra tikrinama kiek dešimtainiame skaičiuje telpa 2^i skaičių. Ši procedūra kartojama tol, kol skirtumas pasiekia nulį.

Pakeiskime dešimtainį skaičių 42 į dvejetainę formą **atimties metodu:**

$$42_{10} = 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 101010_2$$

Pakeiskime dešimtainį skaičių 42 į dvejetainę formą **dalybos metodu:**

Dalybos metode iš 2 dalinama tol , kol liekana taps lygi 0. Čia dalinant pirmiausiai gaunamas jauniausias bitas (**LSB** – least significant bit):

42/2=21 liekana 0
21/2=10 liekana 1
10/2=5 liekana 0
5/2= 2 liekana 1
2/2 = 1 liekana 0
1 / 2=0 liekana 1

Atsakymas: 101010.

Pakeitimas iš dešimtainės į šešioliktainę skaičiavimo sistemą

Atimties metodu:

$$4523 = 1 \cdot 16^3 + 1 \cdot 16^2 + 10 \cdot 16^1 + 11 \cdot 16^0 = 11AB$$

Atsakymas: 11AB

Dalybos metodu:

4523/16= 282 liekana 11 (B)
282/16=17 liekana 10 (A)
17/16 = 1 liekana 1
1/16= 0 liekana 1

Atsakymas: 11AB

Kombinatorika

1. Įvadas. Kombinatorikos objektai ir uždaviniai. +

Paprastai kombinatorika suprantama kaip matematikos sritis, kurioje tiriami klausimai, kiek skirtingų kombinacijų, tenkinančių vienokias ar kitokias sąlygas, galima sudaryti iš turimų objektų.

Kombinatorikos pagrindinis objektas - diskrečiosios aibės.
Daugumoje atvejų pagrindinis dėmesys yra skiriamas dviems operacijų rūšims - tam tikrų poaibių **generavimas,** elementų **sutvarkymas** aibėse.
Skiriami trys pagrindiniai uždavinių tipai:

1. Perskaičiavimas ir išvardinimas.
2. Klasifikacija.
3. Optimizacija.

2. Duotosios aibės poaibiai. Sutvarkytos aibės.

Priminsime[3], kad aibės A *poaibiu* vadinama aibė $B \subset A$, jei visi aibės A elementai yra ir aibės B elementai. Bet kuri aibė yra jos pačios poaibis: $A \subset A$, Tuščioji aibė $\varnothing$ yra bet kurios aibės poaibis: $\varnothing \subset A$. Baigtinė aibė $|A| = n$ turi 2^n poaibių.

Įrodykime šią formulę kitu būdu. Suskaičiuokime, kiek poaibių turi baigtinė aibė $A = \{a_1, a_2, \ldots, a_n\}$.
Yra vienas poaibis, neturintis elementų – tuščioji aibė $\emptyset$.
Poaibių, turinčių po vieną elementą, yra n:

$$\{a_1\},\ \{a_2\},\ \ldots,\ \{a_n\}.$$

Poaibių, sudarytų iš dviejų elementų, yra $\dfrac{n(n-1)}{2}$:

$$\{a_1, a_2\}, \{a_1, a_3\}, \ldots, \{a_1, a_n\}, \ldots, \{a_{n-1}, a_n\}.$$

Poaibių, turinčių po k elementų, yra

$$C_n^k = \frac{n!}{(n-k)!\,k!} = \frac{(n-k+1)\cdot(n-k+2)\cdot\ \cdots\ \cdot(n-1)\cdot n}{k!}.$$

Kitas derinių skaičiaus iš n po k elementų žymėjimas yra $\binom{n}{k}$. Pastebėkime, kad

$$C_n^0 = \binom{n}{0} = C_n^n = \binom{n}{n} = 1.$$

Taigi baigtinė aibė A turi $C_n^0 + C_n^1 + C_n^2 + \cdots + C_n^n$ poaibių.
Šiam skaičiui rasti taikome gerai matematikoje žinomą Niutono[4] binomo formulę:

$$(x+y)^n = \sum_{k=0}^{n} C_n^k x^{n-k} y^k. \qquad (5.1)$$

Kai $x = y = 1$, iš čia gauname ieškomą skaičių $\sum_{k=0}^{n} C_n^k = 2^n$.

Aibė A, kurioje apibrėžtas tvarkos sąryšis R, vadinama **sutvarkyta**. Sutvarkyta aibė vadinama **visiškai sutvarkyta**, jei bet kurie du skirtingi aibės A elementai yra palyginami. Tokiu atveju sąryšis R vadinamas tiesinės tvarkos sąryšiu. Priešingu atveju aibė vadinama iš dalies sutvarkyta, o sąryšis R vadinamas dalinės tvarkos sąryšiu.

3. Junginiai (deriniai, kėliniai, gretiniai). +

Deriniai be pasikartojimų - Visi galimi k-elemenčiai junginiai iš n elementų, kai junginys nuo junginio skiriasi bent vienu elementu, vadinami deriniais ir žymimi

$$C_n^k = \frac{A_n^k}{k!} = \frac{n!}{k!(n-k)!} = \frac{n(n-1)\ldots(n-k+1)}{k!}$$

Deriniai su pasikartojimais - Turime n skirtingų rūšių daiktų. Kiek skirtingų k-elemenčių junginių iš n skirtingų rūšių daiktų galima sudaryti, jei junginys nuo junginio skiriasi bent vienu elementu ir elementai junginyje gali kartotis?

$$\overline{C_n^k} = C_{n+k-1}^k$$

Pavyzdys. Ats.: $\overline{C_4^7} = C_{7+4-1}^7 = C_{10}^7 = \frac{10\cdot 9\cdot 8}{1\cdot 2\cdot 3} = 120$

Gretiniai be pasikartojimų - turime n skirtingų daiktų. Kiek iš jų galima sudaryti k-elemenčių junginių, sudarytų iš skirtingų elementų, kai junginys nuo junginio skiriasi arba bent

vienu elementu, arba jų tvarka?

$$A_n^k = n(n-1)...(n-k+1)$$

$$A_n^k = \frac{n!}{(n-k)!}$$

Pavyzdys. Ats.: $A_5^3 = 5 \cdot 4 \cdot 3 = 60$

Gretiniai su pasikartojimais - turime n skirtingų rūšių daiktų. Kiek iš jų galima sudaryti k-elemenčių junginių, sudarytų iš skirtingų elementų, kai junginys nuo junginio skiriasi arba bent vienu elementu, arba jų tvarka ir elementai junginyje gali kartotis?

Formulė: $\overline{A_n^k} = n^k$

Pavyzdys. Ats.: $\overline{A_{10}^3} = 10^3 = 1000$

Kėliniai be pasikartojimų - Junginiai iš visų n elementų, besiskiriantys vienas nuo kito tik juose esančių elementų tvarka vadinami **kėliniais iš n elementų** arba, trumpiau, **n-elemenčiais kėliniais**. Juos žymėsime

$$P_n = n(n-1)...3 \cdot 2 \cdot 1 = n!$$

Pavyzdys. Ats.: $P_5 = 5! = 120$

Kėliniai su pasikartojimu -

4. Polinominė formulė. Niutono binomas. Binominiai koeficientai. +

Binomo formulė

Binomo formulė – dažnai dar vadinama **Niutono formule**, yra svarbi matematikos teorema, padedanti rasti dvinario, pakelto *n*-tuoju laipsniu, skleidinį. Teorema dažniausiai yra užrašoma

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{k} a^{n-k}b^k = \sum_{k=0}^{n} C_n^k a^{n-k}b^k$$

arba

$$(a+b)^n = \binom{n}{0}a^n + \binom{n}{1}a^{n-1}b + \ldots + \binom{n}{k}a^{n-k}b^k + \ldots + \binom{n}{n}b^n$$

Skaičiai $\binom{n}{k} = C_n^k = \frac{n!}{k! \cdot (n-k)!}$ yra vadinami binomo koeficientais ir yra lygūs skaičiams iš atitinkamos Paskalio trikampio eilutės.

arba

$$(a+b)^n = C_n^0 a^n b^0 + C_n^1 a^{n-1}b^1 + C_n^2 a^{n-2}b^2 + \ldots + C_n^{n-1}a^1 b^{n-1} + C_n^n a^0 b^n$$

kur C_n^k yra deriniai. Jei $(a-b)^n$, tada bus tai minusas tai pliusas, pradedant nuo minuso, pvz:

$$(a-b)^5 = C_5^0 a^5 - C_5^1 a^4 b + C_5^2 a^3 b^2 - C_5^3 a^2 b^3 + C_5^4 ab^4 - C_5^5 b^5$$

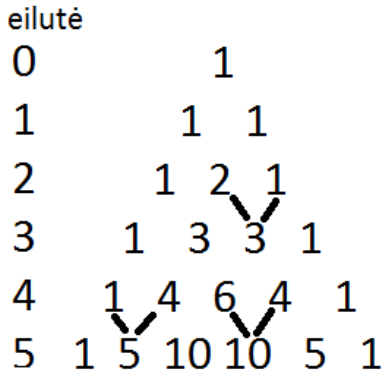
- $(a+b)^2 = a^2 + 2ab + b^2$
- $(a-b)^2 = a^2 - 2ab + b^2$
- $(a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3$
- $(a-b)^3 = a^3 - 3a^2b + 3ab^2 - b^3$

http://lt.wikipedia.org/wiki/Binomo_formul%C4%97

5. Binominės tapatybės. Paskalio trikampis.

1. $C_{n=}^{k} C_n^{n-k}$
2. $C_{n+1}^{k} = C_{n+}^{k} C_n^{k-1}$
3. $C_{n+}^{0} C_{n+}^{1} C_{n}^{2} \ldots + C_{n}^{n} = 2^n$ (iš Niutono binomo, kai a = 1, b = 1).
4. $C_{n-}^{0} C_{n+}^{1} C_{n}^{2} \ldots + (-1)^n * C_{n}^{n} = 0$ (iš Niutono binomo, kai a = 1, b = -1).

Paskalio trikampis:



Paskalio trikampio tolesnius narius galima gauti sudedant viršuje esančias Paskalio trikampio reikšmes kaip parodyta paveikslėlyje.

2 eilutė atitinka $(a+b)^2$ Niutono binomo koeficientus $a^2+2ab+b^2$, n-toji eilutė atitiktų $(a+b)^n$ Niutono binomo koeficientus.

6. Rekurentinių išraiškų metodas.

Šio metodo esmė yra ta, kad kombinatorikos uždavinio sprendimas, kai turime objektą

su n elementų, susideda į analogiško uždavinio sprendimą su mažesniu objektų skaičiumi: tai yra atliekama tam tikros išraiškos, kuri vadinama rekurentine, pagalba.

Pavyzdys: reikia rasti derinių iš n-elementinės aibės po r elementų su pasikartojimais - f_n^r.

Tarkime, kad a_1 į išraišką įeina, tokių derinių skaičius - f_{n-1}^r, jei a_1 į išraišką neįeina, tokių derinių skaičius - f_n^{r-1}. Taigi, $f_n^r = f_n^{r-1} + f_{n-1}^r = f_n^{r-1} + f_{n-1}^{r-1} + f_{n-2}^{r-1} + f_{n-3}^r$.

$f_n^1 = n$, $f_1^r = 1$.

r = 2: $f_n^2 = f_n^1 + f_{n-1}^1 + \ldots + f_1^1 = n + n - 1 + \ldots + 1 = C_{n+1}^2$

r = 3: $f_n^3 = C_{n+1}^2 + C_n^2 + \ldots + C_2^2 = C_{n+2}^3$

$f_n^r = C_{n+r-1}^r$

7. Aibių sąjungos elementų skaičiaus radimas.

$n(A_1 \cup A_2 \cup \ldots \cup A_n) = n(A_1) + n(A_2) + \ldots + n(A_n) - n(A_1 \cap A_2) - n(A_1 \cap A_3) - \ldots - n(A_{n-1} \cap A_n) + \ldots + (-1)^{n-1} n(A_1 \cap A_2 \cap \ldots \cap A_n)$

Ši formulė gali būti įrodoma matematinės indukcijos metodu:

1. Teorema galioja, kai n = 2.
2. Ji galioja, kai elementų skaičius n - 1.
3. Reikia parodyti, kad ji teisinga, kai elementų skaičius n.

8. **Priskirties / išskirties metodas. +**

Sakykime, kad kai kuriems iš N turimų daiktų būdingos savybės. Simboliu $N(\alpha i , \alpha j ,\ldots, \alpha k)$ pažymėkime skaičių daiktų, turinčių savybes (į kitas tų daiktų savybes nekreipiame dėmesio). Jei reikės pabrėžti, kad imami tik tie daiktai, kurie neturi kurios nors savybės, tai tą savybę rašysime su brūkšneliu.

Pavyzdžiui, $N(\alpha 1, \alpha 2 ,\alpha' 4)$ žymi skaičių daiktų, turinčių savybes α1 ir α2 , bet neturinčių savybės α'4 (į kitas jų savybes nekreipiame dėmesio).

$N(a_1', a_2', a_3',\ldots, a_n') = N - N(a_1) - N(a_2) - \ldots - N(a_n) + N(a_1 , a_2) + N(a_1 , a_3) + \ldots + N(a_1, a_n) + \ldots + N(a_{n-1} , a_n) - N(a_1, a_2, a_3) - \ldots - N(a_{n-2}, a_{n-1}, a_n) + \ldots + (-1)^n N(a_1, a_2, \ldots, a_n)$

Pavyzdys

9. Generuojančių funkcijų metodas. Enumeratoriai. Denumeratoriai.

Kombinatorinių uždavinių, kuriuose reikia apskaičiuoti skaičių objektų, tenkinančių nurodytas sąlygas, sprendinys dažnai būna seka $a_0, a_1, \ldots, a_k, \ldots , a_n$. a_k - ieškomų k "matavimų" objektų skaičius. Sekai gali būti priskiriama formali eilutė: $A(x) = a_0 + a_1x + \ldots + a_nx^n$. Ji vadinama šią seką **generuojančia funkcija**.

Enumeratorius: $e^*(z) = (1 + a_1z)(1 + a_2z)\ldots(1 + a_nz)$

Atsiskliaudę gauname: $e^*(z) = 1 + (a_1 + a_2 + \ldots + a_n)z + (a_1a_2 + a_1a_3 + \ldots a_{n-1}a_n)z^2 + \ldots + a_1a_2\ldots a_nz^n$. Taip išvardinami visi deriniai po vieną elementą (prie z), po du elementus (prie z^2) ir t. t.

Denumeratorius: $d^*(z) = (1 + z)^n = 1 + C_n^1z + C_n^2z^2 + \ldots + C_n^nz^n$.

Atitinkami koeficientai yra galimų derinių skaičiai.

Kriptografija

1. Įvadas. Kriptografijos tikslai, uždaviniai, pagrindinės sąvokos.

Šiuolaikinė kriptografija, tai mokslo šaka, sprendžianti elektroninės informacijos saugos problemas. Kadangi kasmet vis daugiau informacijos siunčiama elektroninėmis ryšio priemonėmis, labai svarbu užtikrinti jos saugumą, nes elektroninė informacija (toliau - informacija) dažniausiai perduodama nesaugiais kanalais, pavyzdžiui, interneto ryšiu, ir gali būti pasiekiama beveik visiems. Žodis „kriptografija“ susideda iš dviejų graikiškų žodžių: „kryptos“ reiškiančio „paslėptas“ ir „graphein“ reiškiančio „rašyti“.

Kriptografija - mokslas, susijęs su principais ir metodais, kurie yra skirti įprastinio teksto transformavimui į neperskaitomą tekstą, o po to transformuojant užšifruotą tekstą į paprastą.
Tekstograma - paprastas tekstas.
Šifrograma - užkoduotas tekstas.
Šifras - algoritmas tekstogramos transformavimui į šifrogramą.
Raktas - tam tikra svarbi informacija, kurią naudoja šifras, ir kurią žino tik siuntėjas ir gavėjas.
Užšifravimas, užkodavimas - pradinio teksto transformavimo į šifruojamą procesas naudojant šifrą ir raktą.
Dešifravimas, dekodavimas - atvirkščias procesas užšifravimui.
Kriptoanalizė - įprastinio teksto transformavimo į užkoduotą tekstą ir atvirkštinio proceso tyrimas nežinant šifro ir rakto. Kodo nulaužimas.
Kriptologija - kriptografija, kriptoanalizė kartu.
Steganografija - teksto fizinio nuslėpimo metodai: nematomas rašalas, miniatiūrinės skylutės popieriuje.

2. Kriptografinių sistemų saugos analizė. Informacijos konfidencialumas. Informacijos vientisumas. Informacijos šaltinio autentifikavimas.

Nėra visiškai saugių kriptografinių sistemų. Bet kurią kriptografinę sistemą galima nulaužti. Tačiau kartais nulaužimas kainuoja daugiau resursų nei gaunama naudos nulaužus kriptografinę sistemą. Gali būti ir taip, kad informacija jau nebėra aktuali, kol sugebama nulaužti kriptografinę sistemą. Tokią sistema laikome pakankamai saugia.

Kriptografija yra naudojama užtikrinti informacijos perdavimo saugumui. Saugumas (confidenciality) reikalauja, kad informacija gali būti prieinama tik tam, kam ji yra skirta.

Informacijos konfidencialumas - siunčiamos informacijos užtikrinimas, kad ją galės perskaityti tik gavėjas.

Informacijos vientisumas - siunčiant informaciją ji nebuvo pakeista.

Subjekto autentifikavimas - turi būti galimybė įsitikinti, kad duomenų siuntėjas yra tikrai tas, kas ir turi būti. Vienas iš būdų kaip autentifikuoti - elektroninis parašas.

3. Įsibrovimo tipai (pasyvus / aktyvus).+

Pasyvieji. Tinkle tokie įsibrovėliai nėra aktyvūs – jie tik stebi informacijos mainus tarp

Aldonos ir Broniaus. Svarbiausias jų tikslas – pažeisti informacijos konfidencialumą. Tokio įsibrovimo metu informacija išlieka vientisa, nesugadinta ir neiškraipyta.
Aktyvieji. Tokie įsibrovėliai ne tik stebi informacijos srautą, bet ir patys gali įsiterpti į jį, suklastoti, sugadinti ar perimti duomenų paketus, apsimesti legaliais kriptografinės sistemos vartotojais. Bendrąja prasme tokie įsibrovėlių veiksmai vadinami apsimetimo ataka (angl. impersonation attack). Tokio įsibrovimo metu informacija gali būti iškraipyta.

Keturi atakų tipai:

a. Tik šifrograma.
b. Žinoma tekstograma.
c. Pasirinkta tekstograma.
d. Pasirinkta šifrograma.

4. Klasikinė kriptografija. Monoalfabetiniai šifrai. Cezario šifras. +

Klasikinė kriptografija paprastai suprantama kaip vieno slapto rakto sistema.
Cezario šifras yra klasikinis monoalfabetinis šifrafivo pavyzdys.
Monoalfabetinis šifras, tai toks šifras, kai yra vienas šifro alfabetas, tiksliau, kai vieną ir tą patį tekstogramos simbolį visada atitinka tas pats šifrogramos simbolis. Postūmio ir afininiai šifrai yra monoalfabetiniai.
Cezario šifras – tai šifras, kuris visas užšifruojamo teksto raides keičia raidėmis, kurios abėcėlėje yra per “kažkiek” vietų toliau. Tas “kažkiek” yra šifro raktas. Jeigu imsime lotynišką abėcėlę maksimalus raktų skaičius 25. Lotyniškoje abėcėlėje yra 26 simboliai tačiau jei perstumsime per 26 pozicijas gausime pradinį tekstą. Labai elementarus, paprastas ir visiškai nesaugus šifravimo būdas.

5. Šifravimo būdai. Kriptoanalizė. Polialfabetiniai šifravimo būdai.

Pagrindiniai šifravimo būdai: pakeitimas (vienų alfabeto raidžių pakeitimas kitomis), perstatymas (sukeitimas vietomis).

Perstūmimas: $E(m_j) = c_j = (m_j + b) \bmod n$.

$D(c_j) = m_j = (c_j - b) \bmod n$.

m - tekstogramos alfabeto skaičius.
b - raktas
E - šifravimo funkcija.
D - dešifravimo funkcija.

Monoalfabetiniai šifravimo būdai iš esmės nesaugūs dėl to, kad juos galima nulaužti žinant, kokia kalba parašytas tekstas, ir turint omenyje tos kalbos savybes - pavyzdžiui, raidžių dažnumus.

Šifravimo būdų savybes tiria kriptoanalizė. Kriptoanalizė - įprastinio teksto transformavimo į užkoduotą tekstą ir atvirkštinio proceso tyrimas nežinant šifro ir rakto.

Polialfabetinis šifravimas - toks šifravimas, kai negalime nustatyti vienareikšmės atitikties: vieno teksto simbolio su vienu šifrogramos simboliu. Vienas teksto simbolis gali būti koduojamas skirtingai įvairiose pozicijose.

Polialfabetinio šifravimo pavyzdys - Playfair cipher. Čia naudojama rakto matrica 5x5. Joje įrašomas žodis iš nepasikartojančių raidžių. Likusios vietos užpildomos iš eilės einančiomis dar nepaimtomis raidėmis.

Žinutės užkodavimui, ji padalinama raidėmis po dvi: šios raidės yra atitinkami kampai matricoje. Galimi 4 variantai:

1. Jei abi raidės sutampa arba liko tik viena raidė, pridedamas ‘X’ po pirmos raidės. Šifruojama gauta nauja pora.
2. Jei raidės yra toje pačioje eilutėje, jos paslenkamos per vieną poziciją į dešinę (jei raidė eilutėje yra paskutinė, ji tampa pirmąja raide toje pačioje eilutėje).
3. Jei raidės yra tame pačiame stulpelyje, jos paslenkamos per vieną poziciją žemyn (jei raidė yra paskutinė stulpelyje, ji tampa pirma tame stulpelyje).
4. Kitu atveju, raidės pakeičiamos kituose jų sudaromo stačiakampio kampuose esančiomis raidėmis. Pirma užkoduota raidė yra toje pačioje eilutėje kaip ir nekoduota pirmoji raidė.

Išvados: saugumas žymiai geresnis nei monoalfabetinio šifravimo. Kad surastume raktą, reikalingas daug didesnis šifrogramos tekstas. Gali būti nulaužtas, jei turime šifrogramos tekstą virš 100 raidžių.

6. Šiuolaikinės šifravimo sistemos. Srautiniai ir blokiniai šifratoriai. +

Šiuolaikinės šifravimo sistemos :

Elektroniniu parašu (e. parašu) vadinamas asimetrinis šifravimas, kai duomenys užšifruojami privačiuoju subjekto raktu, o iššifruojami – viešuoju raktu.

Viešojo rakto (asimetrinė) kriptosistema (VRK).

Apibrėžimas. Viešojo rakto (asimetrine) kriptosisema vadinama tokia sistema, kurioje kriptografinės funkcijos realizuojamos naudojant matematiškai susijusių raktų porą: privatųjį raktą – PR (angl. private key), žinoma tik jo savininkui, ir su juo susijusį viešąjį raktą – VR (angl. public key), žinoma visiems vartotojams tinkle. Ši raktų pora vadinama asimetriniais raktais.

VRK leidžia spręsti asimetrinio šifravimo, elektroninio parašo, raktų keitimosi ir kitus svarbius šiuolaikinei informacinei visuomenei uždavinius.

Šiuolaikinės šifravimo sistemos sprendžianti elektroninės informacijos saugos problemas.

Blokinis šifravimas, tai kriptosistema, kai tekstograma suskaidoma fiksuoto ilgio k eilutėmis (blokais), o po to blokai šifruojami atskirai. Šifruojant blokus yra naudojami anksčiau aptarti šifravimo būdai: pakeitimas ir perstatymas , o taip pat Feistelio šifrais. Blokiniais šifrais užšifruojami dideli fiksuoto ilgio duomenų blokai (pvz., 64, 128, 512 bitų ilgio blokai).

Apie srautinį

Tam, kad apibrėžtume srautinį šifrą, apibrėškime raktų srautą, sėklą ir generatorių.

Tarkime K yra šifro raktų aibė. Tada seka k1 k2 k3 $\in$ K yra vadinama raktų srautu.

Raktų srautas gali būti generuojamas atsitiktinai arba apskaičiuojamas pagal algoritmą, vadinamą raktų generatoriumi, kuris generuoja raktų srautą nuo pradinio mažo įėjimo parametro, vadinamo sėklą.

Srautinio šifro apibrėžimas. Tarkime *K* yra kriptosistemos šifro raktų aibė ir tegu $k_1, k_2, k_3, \cdots \in K$ yra raktu srautas. Ši kriptosistema yra vadinama srautiniu šifru tekstogramos eilutei $m_1, m_2, m_3, \cdots$, jei kiekvienas simbolis m_i šifruojamas jam atitinkančiu raktu k_i, $i = 1,2,3, \cdots$, $E_{k_i}(m_i) = c_i$, t.y., m_1 šifruojamas raktu k_1, m_2 šifruojamas raktu k_2 ir t.t.; tarkime d_i yra dešifravimo raktas (rakto k_i inversija), tada kiekvienas šifrogramos simbolis bus dekoduojamas jam atitinkančiu dešifravimo raktu $D_{d_i}(c_i) = m_i$.

Jeigu egzistuoja toks natūrinis skaičius $l \in N$, kad $k_{i+l} = k_i$, tai sakome, kad srautinis šifras yra periodinis su periodu *l*.

Aiškios ribos tarp srautinio ir blokinio šifro užbrėžti negalima: Blokinį šifrą galima traktuoti kaip periodinį srautinį šifrą.

7. Šiuolaikinių šifravimo sistemų tipai: simetrinė / asimetrinė. +

Simetrinė kodavimo sistema - tai sistema naudojanti vieną slaptąjį raktą žinomą tik siuntėjui ir gavėjui. Privalumai: greitis. Trūkumai: saugumas, raktas gali pasimesti, jį gali sužinoti treti asmenys, raktą reikia perduoti specialiais kanalais.
Asimetrinė šifravimo sistema - viešojo rakto sistema. Naudoja viešuosius gavėjo/siuntėjo raktus, juos siunčia kartu su šifruota informacija. Gavėjas/siuntėjas viską atsirakina jau su savo asmeniniu slaptu raktu. Privalumas: saugumas gana didelis, nereikia specialių kanalų slaptam raktui perduot. Trūkumai: lėtas šifravimas, dešifravimas.

8. Simetrinės šifravimo sistemos principai. +

Simetrine kriptosistema vadinama tokia sistema, kuri visoms kriptografinėms funkcijoms atlikti naudoja vieną slaptąjį raktą k, žinoma abiem (arba keliems) protokolo dalyviams.

Nors simetrinės kriptosistemos turi nemažai privalumų, tačiau turi ir viena esminį trūkumą. Tarkime, kad kriptosistemoje yra N kriptosubjektų. Norint išsaugoti kriptosubjektų tarpusavio konfidencialumą, jiems reikia paskirstyti raktų
porų. Kiekvienas jų turi akylai saugoti N-1 raktą ryšiams su kitais kriptosubjektais. Nesunku įsitikinti, kad, didėjant kriptosubjektų skaičiui, reikalingų raktų porų skaičius didėja kvadratine priklausomybe. Išvengti šio trūkumo padeda viešojo rakto kriptosistema.

Šioje kriptosistemoje dažniausiai naudojami du principai: pakeitimas (angl. substitution) ir perstatymas (angl. trasposition/permutation).

Pakeitimas: pvz. Cezario šifras.

Perstatymas, tai toks šifravimo būdas, kai tekstogramos simboliai yra keičiami kitais to paties teksto simboliais, t.y.,- perstatomi. Tokio šifravimo raktas yra atsitiktinis kėlinys, kuris ir nurodo, kurie teksto simboliai keičiami.

Pavyzdžiui, tarkime, kad tekstogramos tekstas yra:

they flung hags (Jie šėlo kaip raganos).

Tarkime, šifro raktas yra:

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13

1, 2, 3, 4, 10 , 7, 8, 9, 5, 6, 11, 12, 13,

t.y. , pirmieji keturi tekstogramos simboliai nekeičiami, „5“ simbolis keičiamas „10“ simboliu, „6“ simbolis keičiamas „7“ ir t.t. Tada šifrograma bus tokia:

THEY HUNG FLAGS (*Jie kabino vėliavas*).

Pats seniausias žinomas tokio šifravimo būdas buvo naudojamas Spartoje (Senovės Graikija). Spartiečiai naudojo perstatymo įrenginį, kurį sudarė kūgio formos strypas, ant kurio spirališkai buvo apvyniojama siaura pergamento, papiruso arba odos juostelė. Slaptas tekstas rašomas ant strypo, (t.y. ant juostelės) iš viršaus į apačią. Po to juostelė nuvyniojama ir teksto (šifrogramos) raidės bus išsibarsčiusios, todėl jo perskaityti bus neįmanoma. Norint šifrogramą perskaityti, juostelę reikia užvynioti ant tokio pat dydžio kūgio formos strypo.

9. Asimetrinės šifravimo sistemos principai. +

Viešojo rakto (asimetrine) kriptosisema vadinama tokia sistema, kurioje kriptografinės funkcijos realizuojamos naudojant matematiškai susijusių raktų porą: privatųjį raktą – PR (angl. private key), žinoma tik jo savininkui, ir su juo susijusį viešąjį raktą – VR (angl. public key), žinoma visiems vartotojams tinkle. Ši raktų pora vadinama asimetriniais raktais.

Kol kas apžvelkime *VRK* taikymo būdus duomenims šifruoti ir elektroniniam parašui realizuoti. Tarkime, *Aldona* turi asimetrinių raktų porą VR_A, PR_A , o *Bronius* – atitinkamai VR_B, PR_B. Iš *VRK* apibrėžimo seka, PR_A, PR_B yra privatieji raktai, žinomi tik jų savininkams. Tuo tarpu VR_A, VR_B yra viešieji raktai, kuriuos žino visi dalyviai.

Sakykime, kad *Bronius*, gavęs žinutę iš *Aldonos*, nori jai išsiųsti atsakymą, pasitelkęs asimetrinio šifravimo algoritmą. Jis turi užšifruoti atsakymą taip, kad jis būtų suprantamas tik *Aldonai*. Kadangi *Bronius* žino jos viešąjį raktą, jis turi galimybę užšifruoti žinutę taip:

$$E(VR_A, t)) = E(VR_A) \circ t = c. \quad (5)$$

Čia operacija ∘ simboliškai pažymėjome funkcijos *E* poveikį argumentui *t*.

Kadangi iššifruoti atsakymą galės tik *Aldona*, iššifravimo funkcija *D* turi priklausyti tik nuo *Aldonai* žinomo privataus rakto PR_A:

$$D(PR_A, c) = D(PR_A) \circ c = t. \quad (6)$$

Tam, kad užšifravimas ir iššifravimas būtų sėkmingi, turi būti tenkinama sąlyga:

$$D(PR_A) \circ c = D(PR_A) \circ E(VR_A) \circ t = t. \quad (7)$$

1-a savybė. Tam, kad galiotų ši sąlyga, PR ir VR turi būti susiję bijekcine funkcine priklausomybe; $VR_A = \varphi(PR_A)$.

Ši sąlyga yra būtina, tačiau nepakankama, nes tam tikrus reikalavimus turi tenkinti ir funkcijos *D* ir *E*.

2-a savybė. Užšifravimo ir iššifravimo funkcijos *E* ir *D* turi būti viena kitai atvirkštinės, t.y. $E(VR_A) \circ D(PR_A) = I$, čia *I* – tapačioji funkcija.

3-ia savybė. Funkcija φ, siejanti VR_A ir PR_A privalo būti vienakryptė, t.y., kad žinant viešąjį raktą VR_A, būtų labai sunku apskaičiuoti privatųjį raktą $PR_A = \varphi^{-1}(VR_A)$.

10. Dviejų pagrindinių šifravimo sistemų palyginimas.

Simetrinė šifravimo sistema yra daug greitesnė už asimetrinę šifravimo sistemą. Simetrinė šifravimo sistema naudoja vienintelį slaptą raktą, reikia sugalvoti, kaip jį saugiai perduoti. Asimetrinė šifravimo sistema naudoja du raktus. Vienas iš jų yra viešas ir visiems žinomas, kitą žino tik pats naudotojas. Viešas raktas naudojamas šifravimui, privatus - dešifravimui. Skiriasi raktų skaičiaus priklausomybės nuo žmonių skaičiaus (n). Simetrinei šifravimo sistemai: n(n - 1) / 2 (kiekvienai porai reikia rakto), asimetrinei užtenka tik 2*n raktų.

Su asimetrine šifravimo sistema galima naudoti skaitmeninius parašus.

Realiai, naudojamos abiejų šifravimo sistemų teigiamos savybės:

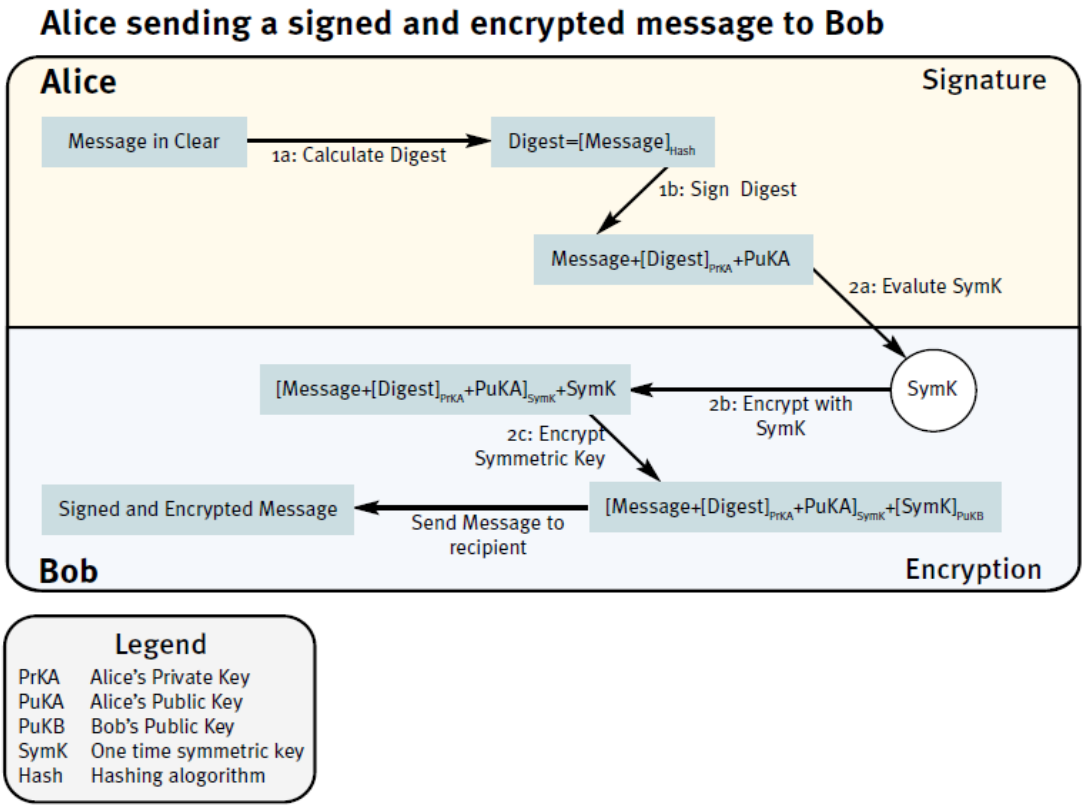


Figure 3: Signature and Encryption details with keys

11. Maišos funkcija. Reikalavimai maišos funkcijoms. Jos panaudojimo sritys.

Kriptografine maišos funkcija vadinama deterministinė procedūra, kuri paima bet kokį informacijos kiekį (paprastai šifruojamą tekstą) ir po tam tikrų operacijų kaip rezultatą duoda fiksuoto ilgio maišos funkcijos reikšmę. Maišos reikšmė (digest) yra tokia, kad bet koks atsitiktinis ar sąmoningas pradinės informacijos pakeitimas pakeis digest reikšmę (Tai vadinama griūties efektu).

Ideali kriptografinė maišos funkcija turi 4 pagrindines savybes:

1. Maišos funkcija turi būti paskaičiuojama nesudėtingai bet kokiam duotam pranešimui.
2. Neįmanoma sugeneruoti pranešimo, kuris turi duotąją maišos reikšmę.
3. Neįmanoma modifikuoti pranešimo nepakeičiant maišos rezultato reikšmės.
4. Neįmanoma rasti dviejų skirtingų pranešimų su tomis pačiomis maišos funkcijos reikšmėmis.

Visos šios savybės orientuotos į tai, kad įsibrovėliui būtų neįmanoma iš h reikšmės gauti kokią nors informaciją apie pranešimą. Kitaip tariant, h elgiasi panašiai kaip atsitiktinių skaičių generatorius - rezultatas nenuspėjamas, bet tuo pat metu maišos funkcija yra deterministinė. Rezultatas turi būti efektyviai paskaičiuojamas.

Panaudojimas: kriptografinės maišos funkcijos naudojamos pranešimo autentifikavimo koduose (MAC), skaitmeniniams parašams realizuoti, aptikti netyčinį arba sąmoningą pakeitimą kode.

12. Pranešimo autentifikavimo kodas (MAC).

Pranešimo autentifikavimas siejasi su:

1. Pranešimo vientisumo užtikrinimu.

2. Pranešimo šaltinio nustatymu.
3. Užtikrinti savybei, kad neleidžia siuntėjui išsiginti siuntus šį pranešimą.

Vienas iš būdų šioms funkcijoms užtikrinti yra MAC. Pranešimo autentifikavimo kodą sudaro: fiksuoto dydžio duomenų masyvas ir raktas. Užtikrina, kad pranešimas nepakeistas, leidžia identifikuoti siuntėją. MAC nėra skaitmeninis parašas. Jis neužtikrina “non-repudation”, nes MAC dirba su privataus rakto infrastruktūra. MAC tag’ą gali sugeneruoti bet kas turintis simetrinį raktą, bet jį turi ne vienas žmogus.

13. Skaitmeninis parašas.

Skaitmeninis parašas - mechanizmas, kurio pagalba pranešimas yra autentifikuojamas, kaip įrodymas, kad pranešimas atėjo tik iš siuntėjo. Skaitmeniniam parašui realizuoti naudojama asimetrinio rakto sistema. Jis užtikrina “non-repudiation” pricipą. Tai reiškiam, jog siuntėjas negali išsiginti siuntęs šį pranešimą. Jis užkoduoja “digest” (žinutės maišos reikšmę) su šiuo privačiu raktu, ir tai yra įrodymas, kad pranešimą siunčia būtent jis, nes tik jis turi privatų raktą.

14. Matematiniai kriptografijos elementai. Pirminiai skaičiai. Eratosteno rėtis. Merseno pirminiai skaičiai.

Pirminiu skaičiu vadiname natūralųjį skaičių, kuris turi lygiai du skirtingus daliklius: vienetą ir save patį.

Eratosteno rėtis naudojamas pirminiams skaičiams rasti. Surašomi visi skaičiai nuo 2 iki n. Einama iš eilės, randamas pirmasis neišbrauktas skaičius. Išbraukiami visi jo kartotiniai, išskyrus jį patį. Visi likę skaičiai yra pirminiai.

Pavyzdys: 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
2 3 ~~4~~ 5 ~~6~~ 7 ~~8~~ 9 ~~10~~ 11 ~~12~~ 13 ~~14~~ 15 ~~16~~ 17 ~~18~~
2 **3** ~~4~~ 5 ~~6~~ 7 ~~8~~ ~~9~~ ~~10~~ 11 ~~12~~ 13 ~~14~~ ~~15~~ ~~16~~ 17 ~~18~~
Toliau einant nėra kartotinių, kuriuos galima išbraukti, visi gauti skaičiai yra pirminiai.

Merseno pirminiais skaičiais vadinami skaičiai, kurių forma: $2^p - 1$, kur p - pirminis skaičius. Ne visi tokio pavidalo skaičiai yra pirminiai.

15. Skaičių faktorizacija. Fundamentalioji aritmetikos teorema.

Fundamentalioji aritmetikos teorema: kiekvieną natūralųjį skaičių galime išreikšti unikalia pirminių skaičių sandauga (faktorizuoti).

Pavyzdys: 288 = 2 * 2 * 2 * 2 * 2 * 3 * 3.

16. Didžiausio bendro daliklio radimas.

Didžiausiu dviejų skaičių a ir b dalikliu vadiname didžiausią skaičių c, c|a ir c|b. Skaičių galime išreikšti tokiu pavidalu a = qb + r, q - dalmuo, r - liekana.
Didžiausias bendras daliklis turi dalinti ir a, ir b be liekanos. Taigi, kadangi qb jis dalina, jis taip pat turi dalinti ir r, tam, kad dalintų a. dbd(a, b) = dbd(b, r), kai a >= b.
I pvz.

a = 24, b = 18.

24 = 18 * 1 + 6
18 = 6 * 3 + 0.

Gavome r = 0. Atsakymas.: 6.

II pvz.
a = 46, b = 32.

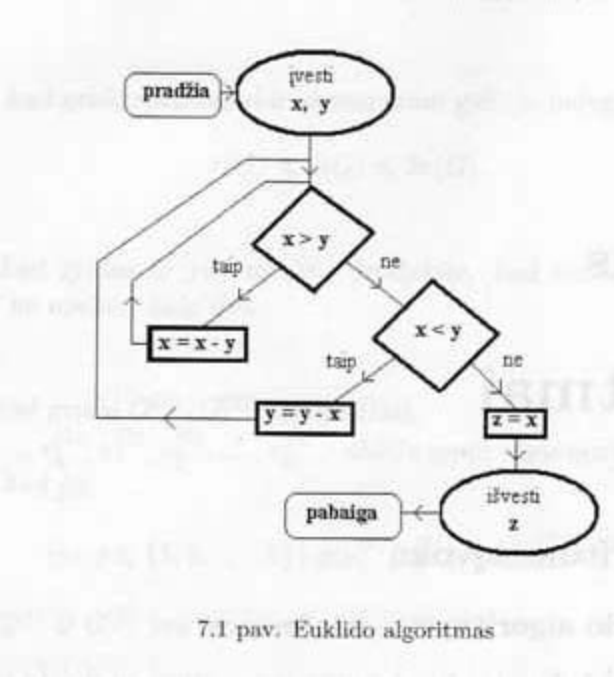
46 = 32×1 + 14;
32 = 14×2 + 4;

14 = 4×3 + **2**;
4 = **2**×2 + 0;
Atsakymas.: 2

~~~~~~

Euklido algoritmas

Didžiausias skaičius z, kuris dalina skaičius x ir y be liekanos vadinamas didžiausiu bendruoju dalikliu (DBD).
Norint rasti šį skaičių galime panaudoti Euklido algoritmą.
Algoritmo schema:



7.1 pav. Euklido algoritmas

I pvz.
Pavyzdžiui turime du skaičius x = 24, y = 18.
Pradžia:
1) x > y (24 > 18), taigi x = 24 - 18 = 6
2) x < y (6 < 18), y = 18 - 6 = 12
3) x < y (6 < 12), y = 12 - 6 = 6
4) x == y, taigi z = x
5) DBD yra 6

II pvz.
turime du skaičius x = 24, y = 17.
Pradžia:
1) x > y (24 > 17), taigi x = 24 - 17 = 7
2) x < y (7 < 17), y = 17 - 7 = 10
3) x < y (7 < 10), y = 10 - 7 = 3
4) x > y (7 > 3), x = 7 - 3 = 4
5) x > y (4 > 3), x = 4 - 3 = 1
6) x < y (1 < 3), y = 3 - 1 = 2
7) x < y (1 < 2), y = 2 - 1 = 1
~~~~~~

8) x == y, taigi z = x
9) DBD yra 1