

ALGORITMAI

14. Algoritmo sąvoka ir savybės

Dirbdami kasdieninius darbus dažniausiai nesusimąstome, kokius veiksmus ir kokia tvarka atliekame. Apie tai pagalvojame, kai norime kokį nors darbą pavesti kitam. Tuomet tenka nurodyti reikiamų atlikti veiksmų sąrašą – suformuluoti algoritmą. Algoritmas – tai nurodymų seka tam, kas turės atlikti užduotį. Mūsų noras – kuo daugiau darbų pavesti kompiuteriui. O tam ir reikia susipažinti su algoritmais, aptarti jų būdingiausias savybes.

Tikriausiai ne kartą teko girdėti žodį „algoritmas“. Ar žinote, ką jis reiškia?

Paprasčiausias algoritmo pavyzdys būtų jūsų kasrytinė ruošą, pavyzdžiui, šitokia: 1. Sučirškus laikrodžiui, keltis. 2. Daryti mankštą. 3. Praustis. 4. Rengtis. 5. Valgyti pusryčius. 6. Išsivalyti dantis. 7. Eiti į mokyklą.

Bet jūs tikriausiai imsite prieštarauti: anoks čia algoritmas, tai darome nesukdami galvos ir neįvardydami...

Ir jūs teisūs. Pavyzdyje pateiktus veiksmus vadiname paprastai – rytmetine ruošą. Tačiau toks nuoseklus veiksmų rikiavimas iš esmės yra algoritmas. Plačiausia prasme algoritmo sąvoka nusako numatomų veiksmų seką, norint padaryti kokį nors darbą ar atlikti užduotį.

1 pav. Šitokie paveikslėliai yra ant taksofonų aparatų. Jais vaizdžiai parodoma, kokius veiksmus reikia atlikti, norint skambinti.

Algoritmu galėtume vadinti ir bet kurį valgių gaminimo receptą iš kulinarijos knygos, ir instrukciją, kaip paskambinti taksofonu, ir gydytojo išrašytą vaistų receptą, ir paaiškinimą, kaip nuvykti iš vienos vietos į kitą. Tai mus supantys kasdieniniai algoritmai, kuriuos atliekame net nesusimąstydami. Vargu ar kam nors iš jūsų tenka ilgai galvoti, kokius veiksmus reikia atlikti, pavyzdžiui, kai norite išsivirti kavos arba arbatos. Tie veiksmai yra tokie įprasti, kad juos atliekate automatiškai. Kitaip yra, kai kokio nors darbo nemokame, o norime jį padaryti, arba kai norime ką nors išmokyti padaryti kokį darbą. Tuomet tenka

apgalvoti, kurie veiksmai ir kokia eilės tvarka turi būti atliekami, kaip juos aiškiau suformuluoti. Tada ir praverčia pažintis su algoritmo sąvoka.

Algoritmas – tai aiškūs ir tikslūs nurodymai, kaip ir kokių veiksmų seką reikia atlikti norint pasiekti užsibrėžtą tikslą arba išspręsti suformuluotą uždavinį.

1 pavyzdys. Vilkas, ožka ir kopūstas. Tai labai senas uždavinys, aprašytas dar VIII amžiuje.

Žmogui prireikė laivelio perkelti per upę vilką, ožką ir kopūstą. Tačiau mažame laivelyje galėjo tilpti tik žmogus ir su juo – arba vilkas, arba ožka, arba kopūstas. Tačiau palikus vienus vilką su ožka, vilkas suėstų ožką, palikus ožką su kopūstu, ožka sugraužtų kopūstą, o žmogaus akivaizdoje niekas nieko neliečia. Žmogus sėkmingai perkėlė savo krovinį per upę.

Šio uždavinio sprendimo algoritmą pateiksime piešiniais (2 pav.). Argi ne vaizdu?

2 pav. Vilko, ožkos ir kopūsto perkėlimo per upę algoritmas.

Iš algoritmų istorijos

Algoritmo sąvoka atsirado labai seniai, daugiau kaip prieš tūkstantį metų. Pats žodis „algoritmas“ kilo iš IX a. arabų matematiko ir astronomo Mohamedo ibn Musos al Chorezmio lotyniškais rašmenimis parašyto vardo *Algorithmi*. Šis mokslininkas sudarė keturių aritmetikos veiksmų su dešimtainiais skaičiais taisyklės, kurias Europoje imta vadinti „algorizmais“. Vėliau šis žodis buvo pakeistas „algoritmu“ ir jam suteikta platesnė prasmė – juo pradėta vadinti įvairių skaičiavimų (ne tik aritmetinių) taisyklės.

Ilgą laiką algoritmo sąvoką vartojo tik matematikai, todėl dažnai ši sąvoka siejama su matematinių uždavinių sprendimu. Tai patvirtina ir lietuviškojoje enciklopedijoje pateikta šitokia algoritmo apibrėžtis: „Algoritmas – pagal griežtas taisyklės atliekamų skaičiavimų (operacijų) seka, kuri leidžia išspręsti matematikos ar logikos kurios nors klasės uždavinius“. Galima suformuluoti dar bendriau: „Algoritmas – tai aiškiai suformuluotas taisyklių rinkinys kuriam nors tikslui pasiekti“.

3 pav. Mohamedas ibn Musa al Chorezmis.

Taigi matematikai ilgus šimtmečius vartojo algoritmo sąvoką ir tik XX a. pradžioje ėmė tirti pačius algoritmus – atsirado nauja matematikos šaka: algoritmų teorija. Tai buvo

didelis įvykis mokslo pasaulyje, toks pat kaip radijo, kino, televizijos, kibernetikos ar genetikos atradimai.

2 pavyzdys. Eratosteno rėtis. Įdomų metodą visiems pirminiams skaičiams nuo 2 iki n rasti pasiūlė graikų matematikas Eratostenas (III–II a. pr. Kr.). Priminsime, kad **pirminiai skaičiai** – tai didesni už vienetą natūralieji skaičiai, turintys tik du daliklius: vienetą ir save patį. Šios sąlygos neatitinkantys skaičiai vadinami **sudėtiniais**. Eratostenas surašė visus natūraliuosius skaičius nuo 2 iki 1000 ant papiruso ir ištempė jį ant rėmelio. Sudėtinius skaičius pradūrė, ir papirusas tapo tarsi rėtis, kuriuo galima „išsijoti“ sudėtinius skaičius, pirminius paliekant (4 pav.). Tokiam pirminių skaičių radimo būdui prigijo „Eratosteno rėčio“ vardas.

4 pav. Eratosteno rėtis.

Paaiškinsime Eratosteno rėčio algoritmą pavyzdžiu. Surašykime dalį natūraliųjų skaičių pradedant 2:

	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	...						

Pirmasis skaičius (2) yra pirminis. Pradedame „sijoti“ nuo jo – perbraukiame visus skaičius, kurie dalūs iš 2, t. y. kas antrą skaičių (paties dvejeta nebraukiame):

	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	...					

Toliau imamas paeiliui kitas neperbrauktas skaičius (3) ir perbraukiami visi dar neperbraukti jo (trejeto) kartotiniai:

	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	...					

Analogiškai braukant, išbraukiami visi sudėtiniai skaičiai, ir lieka tik pirminiai.

Sukūrus kompiuterį, algoritmai tapo dar svarbesni. Su jais susiduriama automatizuojant bet kurį darbą. Tarkime, jei kokį nors veiksmą pavyksta griežtai išreikšti algoritmu, tai jį galima pavesti atlikti mašinai, pavyzdžiui, robotui.

Algoritmai informatikoje

Itin svarbi algoritmo sąvoka informatikoje. Prisiminkime, kad informatika – tai mokslas apie informacijos perdavimą, saugojimą ir, svarbiausia, apdorojimą. Kaip apdoroti informaciją, t. y. kaip atlikti veiksmus su informacija (duomenimis)? Tam kuriamos taisyklės, vadinamos algoritmais. Dabar galime pateikti tikslesnį algoritmo apibrėžimą.

*Informacijos (duomenų) apdorojimo taisyklės vadinamos **algoritmais**. Kitaip sakant, algoritmas – tai taisyklių rinkinys, kuriuo nurodoma, kaip iš vieno duomenų gauti kitus.*

Algoritmo sąvoka informatikoje – fundamentali, pirminė sąvoka. Jos prasmė atsiskleidžia apmąstant pavyzdžius, nagrinėjant algoritmų savybes, sprendžiant uždavinius.

Algoritmai sudaromi tam, kad juos kas nors vykdytų – žmogus ar kompiuteris, ar kitoks mechanizmas. Samprotaujant apie algoritmus, visuomet reikia turėti galvoje vykdytoją: algoritmas sudaromas tam, kad jį kas nors atliktų ir gautų rezultatus. Mes dažniausiai kalbėsime apie tuos algoritmus, kurių vykdytojas – kompiuteris. Toliau panagrinėsime būdingąsias algoritmų savybes: diskretumą, aiškumą, baigtumą, universalumą, rezultatyvumą.

Diskretumas

Bet kuriame algoritmo užrašo pavyzdyje pirmiausia krinta į akis veiksmų suskaidymas į dalis – žingsnius. Prisiminkime kad ir galvosūkio „Vilkas, ožka ir kopūstas“ algoritmą – jį sudarė septyni žingsniai. Kuo išsamiau norime nurodyti (aprašyti) veiksmus, tuo daugiau turi būti žingsnių.

*Galimybė išskaidyti algoritmo veiksmus žingsniais vadinama **algoritmo diskretumu**.*

Taigi bet kuris algoritmas pasižymi diskretumo savybe – jis išskaidomas į atskirus žingsnius, kuriuos atlieka vykdytojas.

Aiškumas

Sudarant algoritmą, visuomet turi būti atsižvelgiama į tai, kas jį vykdys. Nuo to, kiek žino vykdytojas, kokius veiksmus jis supranta ir gali atlikti, priklauso algoritmo užrašymo būdas.

Algoritmo aiškumas – tai jo pateikimas vykdytojui suprantama ir aiškia kalba.

Pavyzdžiui, jei norėsime sudaryti lygties $x^2 - 9x + 8 = 0$ sprendimo algoritmą devintokui, kuris moka spręsti kvadratinės lygtis, tai pakaks trumpo nurodymo.

1. Pagal formulę $D^2 = b^2 - 4ac$ apskaičiuokite diskriminantą.

2. Raskite šaknis: $x_{1,2} = \frac{-b \pm \sqrt{D}}{2a}$

Mokiniui, kuris nežino kvadratinės lygties sprendimo metodų, teks parašyti išsamesnį algoritmą.

- Apskaičiuokite reiškinio reikšmę: $9 \cdot 9 - 4 \cdot 1 \cdot 8$.
- Iš gautojo skaičiaus ištraukite šaknį ir rezultatą pažymėkite raide p .
- Apskaičiuokite reiškinio $(9 + p) / 2$ reikšmę. Tai bus pirmoji šaknis x_1 .
- Apskaičiuokite reiškinio $(9 - p) / 2$ reikšmę. Tai bus antroji šaknis x_2 .

Nemokantiems traukti kvadratinės šaknies tektų dar daugiau aiškinti.

Taigi algoritmo veiksmai turi būti užrašomi aiškiai, griežtai, vienareikšmiškai, – taip, kad būtų suprantami tiems, kurie atliks algoritmą.

Baigtinumas

Visi mūsų iki šiol nagrinėti algoritmų pavyzdžiai turėdavo nedaug žingsnių. Aišku, kad tokius algoritmus vykdytojas visuomet užbaigs. Bet juk gali būti algoritmų, turinčių labai daug žingsnių...

Norint gauti rezultatą, atliekamų veiksmų skaičius visuomet turi būti baigtinis. Ši algoritmo savybė vadinama **baigtinumu**.

3 pavyzdys. Susipažinkime su šaknies traukimo algoritmu, kurį aprašė senovės graikų mokslininkas Heronas maždaug prieš 2000 metų.

1. Pasirenkame „iš akies“ pirmąją apytikslę šaknies reikšmę.
2. Pošaknio skaičių dalijame iš pirmosios apytikslės šaknies reikšmės.
3. Jeigu gautojo dalmens ir pasirinktos apytikslės reikšmės skirtumas neviršija sutartos paklaidos, tai sakome, kad šaknis ištraukta. Priešingu atveju apskaičiuojame dalinio ir dalmens aritmetinį vidurkį. Šis vidurkis bus tikslesnė šaknies reikšmė – antroji apytikslė reikšmė.

4. Antrąją apytikslę reikšmę patikriname taip pat kaip ir pirmąją, t. y. pošaknio skaičių dalijame iš antrosios reikšmės. Jeigu rezultato tikslumas nepakankamas, ieškome trečiosios apytikslės reikšmės tokiu pat būdu kaip ir skaičiuodami antrąją. Ir taip toliau.

Šaknies traukimo algoritmas gana sudėtingas. Atliekant jį gali tekti daug kartų kartoti analogiškus veiksmus (2-ąjį ir 3-įjį žingsnius). Tačiau vis dėlto kada nors gausime tokią šaknies reikšmę, kurios tikslumas mus tenkins, ir algoritmo veiksmai bus užbaigti.

Universalumas

Jeigu atidžiai įsigilintume į šaknies traukimo algoritmą, tai galėtume pasakyti, kad jis tinka šakniai traukti iš bet kurio neneigiamo skaičiaus. Taigi algoritmą galime kartoti su vis kitais pradiniais duomenimis.

*Algoritmo tinkamumas įvairioms pradinų duomenų reikšmėms vadinamas **universalumu**.*

Tai viena svarbiausių algoritmo savybių, ir nuo jos priklauso algoritmų praktinė vertė. Pavyzdžiui, sudaryti atskirą šaknies traukimo algoritmą kiekvienam skaičiui nebūtų prasmės. Tačiau yra ir tokių algoritmų, kurie skirti vienam duomeniui, pavyzdžiui, taksofonai, kurie priima tik vienos rūšies monetas. Argi šie algoritmai neturės universalumo savybės? Pamąstykime.

Jeigu tartume, kad neturės, tuomet iškiltų klausimas – kaipgi bus su algoritmais, kurie skirti tik dviem konkrečioms duomenims ir pan. Patektume į aklavietę. Todėl universalumo sąvoką reikia patikslinti. Reikia laikyti, kad kiekvienas algoritmas yra skirtas kuriai nors pradinų duomenų klasei. Pavyzdžiui, šaknies traukimo algoritmui tokią klasę sudaro visi neneigiami skaičiai, taksofonui – nurodytos rūšies monetos. Algoritmo universalumas – tai algoritmo tinkamumas nurodytai pradinų duomenų klasei.

Rezultatyvumas

Paminėsime dar vieną gana akivaizdžią algoritmų savybę, kuri reiškia, kad atlikę kurį nors algoritmą, turime gauti kokį nors rezultatą.

***Algoritmo rezultatyvumas:** algoritmas turi duoti kokį nors konkretų rezultatą.*

Rezultatas suprantamas plačiąja prasme. Tai gali būti, pavyzdžiui, ne tik duotos lygties sprendiniai, bet ir pranešimas, kad duotoji lygtis sprendinių neturi arba kad blogai parinkti pradiniai duomenys (pavyzdžiui, šaknies traukimo algoritmo atveju – neigiamas skaičius), todėl šio uždavinio sprendimas netenka prasmės.

Pabaigai pateiksime vieną, galbūt jums žinomą, žaidimą. Pabandykite patys pažaisti pagal pateiktą algoritmą, panagrinėkite ir aptarkite jo savybes.

4 pavyzdys. Žaidimas su pagaliukais. Žaidėjai paeiliui ima iš krūvelės pagaliukus (9 pav.). Vienu ėjimu galima paimti vieną arba du pagaliukus. Pralaimi tas, kuris paima paskutinį pagaliuką.

9 pav. Žaidimo su pagaliukais strategija – algoritmo atlikimas.

Prieš sudarydami algoritmą aptarsime ir pagrįsime jo sprendimą. Šio algoritmo pradinis duomuo – pagaliukų skaičius krūvelėje, rezultatas – pranešimas, kuris iš žaidėjų laimi.

Pirmiausia nustatome laimėjimo ir pralaimėjimo situacijas. Kai krūvelėje lieka vienas pagaliukas, susidaro pralaimėjimo situacija. Kai krūvelėje yra du pagaliukai, susidaro laimėjimo situacija, nes, paėmę vieną pagaliuką, priešininkui sudarysime pralaimėjimo situaciją. Tolesnė situacija (trys pagaliukai krūvelėje) – taip pat laimėjimo, nes, paėmę du pagaliukus, priešininkui paliksime vieną. Pralaimėjimo situacija susidarys, kai krūvelėje bus keturi pagaliukai, nes, paėmę vieną arba du pagaliukus, priešininkui „padovanosime“ laimėjimo situaciją. Toliau analogiškai samprotaudami, gausime šitokių situacijų seką (laimėjimo situacija pažymėta pliuso ženklu, pralaimėjimo – minuso):

1	2	3	4	5	6	7	8	9	10	11	...
–	+	+	–	+	+	–	+	+	–	+	

Taigi žaidimą laimi tas, kuris sugeba palikti priešininkui krūveles, sudarytas iš šitiek pagaliukų (skaičiuojant nuo „pabaigos“): 1, 4, 7, 10, 13, 16 ir t. t. Dėsningumą nesunku pastebėti, tad belieka suformuluoti algoritmą, kaip turi žaisti pirmasis žaidėjas, kad jis laimėtų, jeigu tik įmanoma. Taigi veiksmų seka šitokia.

1. Patikrinti, ar krūvelėje esančių pagaliukų skaičius dalus iš 3.
2. Jeigu taip, imti du pagaliukus, jeigu ne – vieną pagaliuką.
3. Pirmą ir antrą algoritmo žingsnį kartoti tol, kol bus paimtas paskutinis pagaliukas.

Žaidėjas, kuriam susidaro pralaimėjimo situacija, ima po vieną pagaliuką: jei žinai, kad praloši, tai nors stenkis žaisti ilgiau – gal varžovas apsiriks... algoritmo vykdytojas

algoritmas
algoritmo diskretumas
algoritmo aiškumas
algoritmo baigtinumas
algoritmo masiškumas
algoritmo rezultatyvumas

PRATIMAI IR UŽDUOTYS

1. Pateikiame ištrauką iš lietuvių liaudies pasakos „Gulbė karaliaus pati“:
– Pasižiūrėk, pro kurį langą jinai įskrenda, ir ant palangės papilk deguto, tai jos kojos ir sparnai prilips, tada kaire ranka griebk, dešinèn mesk – kaip buvo karalienė, taip ir bus.
Ar įžiūrite čia algoritmą? Jei taip, suformuluokite jį griežčiau.
2. Išrinkite algoritmo sąvokos aiškinimus iš keleto knygų, žodynų, enciklopedijų, vadovėlių. Aptarkite juos klasėje. Paaiškinkite skirtumus. Kurie iš jų jums atrodo taikliausi, kurie – nevykę ar tinka tik tam tikromis situacijomis? Savo požiūrį argumentuokite.
3. Galvosūkio „Vilkas, ožka ir kopūstas“ algoritmą (1 pavyzdys) suformuluokite žodžiais. Kaip, jūsų nuomone, aiškiau?
4. Pateikiame Mikės Pūkuotuko sudarytą įdomų algoritmą, paimtą iš A. Milno knygos „Pūkuotuko pasaulis“.

KOKIA TVARKA KO IEŠKOTI?

- | | |
|-------------------|------------------------------|
| 1. Speciali vieta | (Rasti Paršelį) |
| 2. Paršelis | (Sužinoti, kas yra Mažas) |
| 3. Mažas | (Rasti Mažą) |
| 4. Triušis | (Pasakyti, kad radau Mažą) |
| 5. Vėl Mažas | (Pasakyti, kad radau Triušį) |

Paieškokite analogiškų algoritmų kitose knygose.

5. Suformuluokite pynutės nėrimo vąšeliu algoritmą. Koks būdas labiau tinka šiam algoritmui aprašyti: žodinis ar vaizdinis?
6. Suformuluokite vyriško kaklaraiščio mazgo surišimo algoritmą.
7. Reikia iškepti 6 kotletus per 15 min. Keptuvėje telpa tik 4 kotletai. Viena kotletu pusė iškepa per 5 min. Užrašykite algoritmą kotletams iškepti.
8. Įsivaizduokite, kad siaurame koridoriuje, kuriame per ankšta prasilenkti ir kurio sienoje yra viena niša, susitiko keturi žmonės su kitais keturiais, einančiais priešinga kryptimi (6 pav.). Kaip jiems prasilenkti?

6 pav. Dviejų būrių prasilenkimas.

Suformuluokite algoritmą. Geriausia būtų jį pavaizduoti grafiškai, piešiniais.

9. Naudodamiesi Eratosteno rėčio algoritmu raskite visus pirminius skaičius iki 1000. Uždutį atlikite ant atskiro lapo, sudėtinis skaičius „pradurkite“. Ar graų paveikslą gavote? Kiek laiko sugaišote ŗiai uųduočiai atlikti?

10. Suformuluokite Eratosteno rėčio metodą ŗodžiais, kai norima rasti visus pirminius skaičius nuo 2 iki n .

11. Uųrašykite algoritmą kampo pusiaukampinei nubrėŗti, kai naudojamas tik skriestuvu ir liniuote. Taisykles kaip brėŗti galite rasti geometrijos vadovėliuose.

12. Suformuluokite natūralių skaių dalumo iš 3, 4, 5, 6, 9 ir 11 nustatymo taisykles – algoritmus.

13. Ŗtai senovinis uųdavinys iš maždaug XIII ŗimtmečio.

Vaikŗtinėjo trys mergaitės, kiekviena su savo tėčiu. Visi ŗeŗi atėjo prie upės ir panoro persikelti į kitą krantą. Ten tebuvo vienas laivelis, kuriuo, be irkluotojo, galėjo saugiai persikelti tik du ŗmonės. Mergaitės buvo ne tokios jau mažos, kad negalėtų savarankiŗkai irkluoti. Tačiau jos uųsispyrė ir pareiŗkė nė viena nesutinkanti keltis laiveliu arba likti ant kranto su kuriuo nors iš svetimų tėčių, jei ten nebus jos tėčio.

Tokiu būdu atsirado nenumatytos papildomos persikėlimo ŗalygos. Vis dėlto pramogautojai nutarė pabandyti jas įvykdyti. Kaip jie turėtų elgtis?

Suformuluokite persikėlimo per upę algoritmą.

14. Išrikiuokite į vieną eilę ŗeŗias ŗaŗkes pakaitomis – juoda, balta, vėl juoda, vėl balta ir t. t. (7 pav.). Iš kairės (arba iš deŗinės) palikite laisvos vietos ne daugiau kaip keturioms ŗaŗkėms.

7 pav. Punktyru paųymėtos laisvos vietos ŗaŗkėms.

Reikia perkilnoti ŗaŗkes taip, kad kairėje atsidurtų visos baltos, o deŗinėje – visos juodos. Į laisvą vietą galima perkelti tik po dvi gretimas ŗaŗkes ir negalima keisti tvarkos, kuria jos sudėliotos. Uųdavinį galima išspręsti trimis ėjimais (triskart perkėlus ŗaŗkes).

Suformuluokite ŗio uųdavinio sprendimo algoritmą.

Pagalvokite, kiek reikėtų perkėlimų, jei padidintume ŗaŗkių skaičių: išdėliotume iš viso aŗtuonias – 4 baltas ir 4 juodas ŗaŗkes. Suformuluokite algoritmą.

Analogiŗkai pabandykite su 5 baltomis ir 5 juodomis ŗaŗkėmis.

15. Penkiolika degtukų išdėlioti eilute (8 pav.). Reikia surinkti juos į penkias krūveles po tris degtukus. Galima kelti tik po vieną degtuką, kaskart perŗokant per tris gretimus (nesvarbu koku atstumu).

8 pav. Penkiolikos degtukų eilutė.

Išspręskite šį uždavinį dešimčia ėjimų. Kad būtų vaizdesnis algoritmas, degtukus sunumeruokite, o rašydami nurodykite perkeliama degtuko ir to, prie kurio perkelsite kitus degtukus, numerius.

16. Parinkite keletą valgių gaminimo receptų. Pateikite pasiūlymą, kaip reikėtų parengti jų griežtesnius, lakoniškesnius, patogesnius naudoti aprašus.

17. Atlikite 3 pavyzdžio šaknies traukimo algoritmą su įvairiomis pradinių duomenų reikšmėmis, pavyzdžiui: a) 3, b) 5, c) 8, d) 20, e) 100. Tikslumą pasirinkite patys.

18. Du berniukai žaidžia tokį žaidimą. Pirmasis pasako skaičių 1, kitas prie jo prideda bet kurį natūralųjį skaičių, ne didesnį kaip 9. Pirmasis prie tos sumos irgi prideda pasirinktą, bet ne didesnį kaip 9, natūralųjį skaičių ir t. t. Berniukai tol prideda skaičius, kol gauna 101. Tas, kuris pirmasis pasako šį skaičių, laimi.

Sudarykite algoritmą, kaip turėtų žaisti pirmasis žaidėjas, kad laimėtų. Aptarkite ir išnagrinėkite žaidimo algoritmo savybes.

19. Ant kūdikiams skirta maisto pakelių ar dėžučių būna instrukcijos, kaip pagaminti maistėlį. Keletas skirtingų algoritmų aprašymo būdų pateikta 1 lentelėje. Aptarkite kiekvieną jų. Kuris jums atrodo aiškiausias? Jei būtumėte gamintojas, kokią instrukciją ir kaip pateiktumėte ant dėžutės?

1 lentelė. Instrukcijos kūdikių maistui paruošti

Išsamus atliekamų veiksmų paaiškinimas	Paaiškinimai paveikslėliais	Trumpi nurodymai
Paimkite arbatinuką, gerai jį išplaukite, pripilkite vandens ir užkaiskite. Vandeniui užvirus, pavirinkite dar keletą minučių. Tada ataušinkite jį iki maždaug 60°C.		1. Užvirinkite vandenį ir ataušinkite jį iki 60°C.
Paimkite gerai išplautą buteliuką, įpilkite į jį nedaug vandens. Specialiu šaukšteliu, kurį radote miltelių dėžutėje, pasemkite miltelių. Kaupą nubraukite peiliu. Suberkite miltelius į buteliuką. 1 šaukštelis skirtas 30 mg vandens. Vieno mėnesio kūdikis išgeria maždaug 120 mg mišinėlio, t. y. reikės 4 šaukštelių miltelių.		2. Į buteliuką įpilkite truputį vandens. 3. Įberkite miltelių. 1 šaukštelis (be kaupo) skirtas 30 mg vandens.
Buteliuką gerai užsukite ir stipriai papurtykite, kad mišinys susiplaktų. Tada atsukite ir pripildykite vandens iki reikiamo kiekio.		4. Papurtykite buteliuką. 5. Pripilkite dar vandens iki reikiamo kiekio.

Užsukite buteliuką su žinduku. Patikrinkite mišinėlio temperatūrą: jautriausia vieta – riešas, ant jo užlašinę kelis lašus, nustatysite, ar ne per karštas.

6. Patikrinkite mišinio temperatūrą.

20. Įsivaizduokite, kad jūs įkūrėte firmą, kuri organizuoja išvykas į žinomų žmonių gyvenimo vietas. Kokius miestelius bei kaimelius pasiūlytumėte aplankyti besidomintiems Vincu Mykolaičiu-Putinu? Suplanuokite maršrutą, paskirstykite laiką. Tarkime, kad bus keliaujama savo transportu.

Analogiškus maršrutus sudarykite keliaujantiems į vietas, susijusias su šiais įžymiais žmonėmis: a) Mikalojaus Konstantino Čiurlionio; b) Antano Baranausko; c) Simono Daukanto; d) Jono Basanavičiaus.

21. Sudarykite paprastųjų trupmenų sudėties algoritmą, aptarkite ir išnagrinėkite jo savybes.

22. Temos referatams: a) „Senovės mokslininko Mohamedo ibn Musos al Chorezmio gyvenimas ir veikla“; b) „Algoritmai mūsų gyvenime“; c) „Man patinkantys algoritmai matematikoje“.