

Žaidimo kortelės

BEBRAS — informatinio mąstymo ugdymas

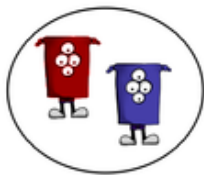
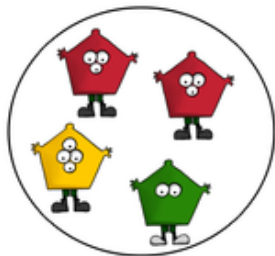
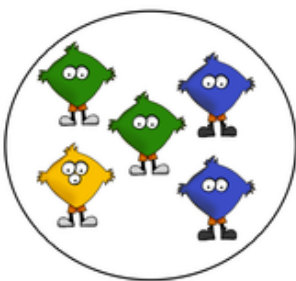


BEBRAS

Informatinio
mąstymo
ugdymas



Vaiva nupiešė keturiolika mielų padarėlių ir suskirstė juos į keturias grupes.



Pagal kurį požymį ji skirstė?

Duomenų grupavimas pagal įvairius požymius – svarbus informatikos uždavinys. Informatikas turi išmokti įžvelgti požymius, atpažinti objektus, turinčius vienokių ar kitokių požymių, skirstyti objektus į grupes pagal požymius ir pan.

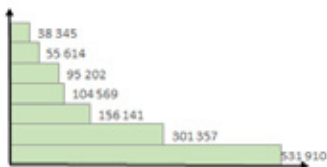


MIESTAI



Rodyklėmis parodyta, kaip aplankyti pagrindinius Lietuvos miestus. Pradedama nuo Vilniaus, turinčio daugiausia gyventojų (531910), toliau lankomi miestai gyventojų mažėjimo tvarka.

Pateiktoje diagramoje įrašytas gyventojų skaičius šiuose miestuose, bet, deja, nurodyti miestų pavadinimai.

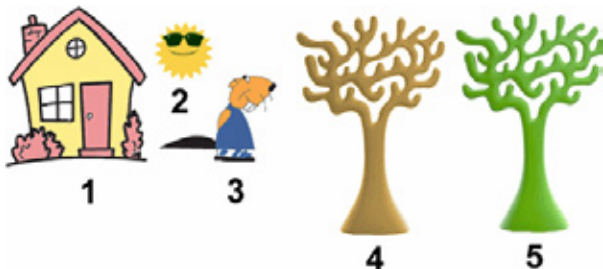


Kiek gyventojų Alytuje?

Informatikoje būtinas informacijos skaitymas ir susiejimas, ypač kai duomenys pateikiami įvairiais būdais: juostine diagrama, žemėlapiu. Svarbu pastebėti ryšį tarp surikiuotų skaičių diagramoje ir miestų rikiavimo eilę nurodančių rodyklių žemėlapyje.



Bebbras turi penkis antspaudus, pažymėtus nuo 1 iki 5:



Šiais antspaudais jis nupiešė tokį paveikslėlį:



Kokia tvarka jis naudojo antspaudus?

Programuojant dažnai tenka pateikti informaciją, atsižvelgiant į jos svarbą, prioritetą ar lygmenis. Grafikos rengyklėse objektai dėstomi sluoksniais: tolimiausias objektas dedamas pirmiausia, artimiausias objektas dedamas paskiausiai.



ŪKININKAS IR GYVATĖ



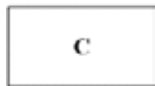
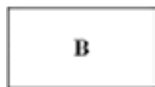
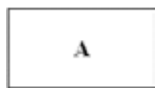
Šaltą žiemos dieną ūkininkas rado sustingusią nuo šalčio gyvatę ir parsinešė namo. Kai tik sušilo, gyvatė atgijo ir įgėlė ūkininkui. Mirdamas ūkininkas pasakė: „Gavau tai, ko nusipelniau, nes pagailėjau nedoro gyvio“.



parsinešė namo

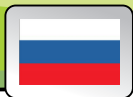


įgėlė



Kuria tvarka žodžiai (ūkininkas, gyvatė, nuodai) turi būti įrašyti stačiakampiuose A, B, C, kad pateikta diagrama vaizduotų teisingą teiginį?

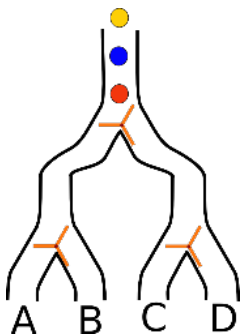
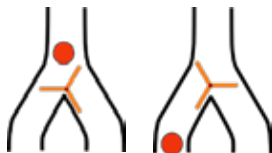
Informacijos abstrahavimas – svarbi informatinio mąstymo dalis – būtinas, apdorojant bet kokią informaciją. Abstrahuojant informaciją, svarbu išlaikyti prasmę ir struktūrinius ryšius. Tai informacijos valdymo sritis, informacijos vadybos principų taikymas.



TRYS TRIGERIAI



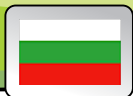
Trigeris yra įrenginys, kuris gali būti vienos iš dviejų būsenų. Kiekvieną kartą, kai gaunamas signalas, trigerio būsena pasikeičia. Sumanus bebras turi trigerį, kuris veikia taip:



Kamuolys krinta iš viršaus ir nuslysta į kairę arba į dešinę. Krisdamas jis pasuka trigerį taip, kad po jo krintantis kitas kamuolys nuslystų priešinga kryptimi. Bebras sukonstravo įrenginį iš trijų trigerių (pa-veikslėlis kairėje).

Iš kurio vamzdelio iškris paskutinis (geltonas) kamuolys?

Darbu su informacija vartojamas trigerio modelis – įtaisas, galintis nuolat būti vienos iš dviejų stabilių pusiausvyros būsenų ir pereinantis iš vienos būsenos į kitą, tik paveiktas išorinio signalo. Trigeris ypač reikalingas informacijos būsenoms saugoti.

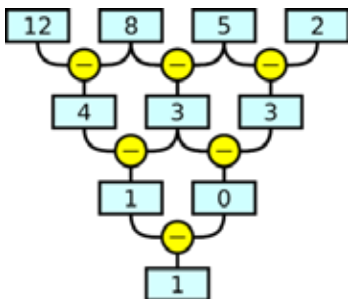


SKIRTUMŲ AUTOMATAS



Rūta suprogramavo automatą: įvedama mažėjanti 4 teigiamų sveikųjų skaičių seka. Kiekvienoje žemesnėje eilutėje skaičiuojamas viršutinės eilutės dviejų gretimų skaičių skirtumas. Suskaičiavus visus galimus dviejų skaičių skirtumus, gaunamas vienas skaičius – rezultatas.

Pavyzdžiui, įvedę skaičių seką 12 8 5 2, gauname 1.



Papildykite skaičių seką taip, kad, įrašius ją į šį automatą, rezultatas būtų lygus nuliui.

16 6

Šis skirtumų skaičiavimo automatą atlieka paprastus skaičiavimus, šiek tiek panašius į kompiuterio vykdomus mikroschemoje. Informacijos apdorojimas išskaidomas į mažesnius žingsnius ir tik tada gaunamas rezultatas.



LINKSMIEJI LANGAI



Laive langų stiklai gali būti skaidrūs arba tamsinti. Kai žiūrime per tokių dviejų langų stiklus, matome skaidrią, truputį tamsintą arba tamsią spalvą (žr. paveikslėlių).



Kapitonas Jonas savo naujoje jachtoje įrengė dalį skaidrių ir dalį tamsintų langų, kaip parodyta paveikslėliuose žemiau. Kai stovima ant kranto, iš tam tikros vietos galima žvelgti pro du atitinkamus langus į kitą jachtos pusę.

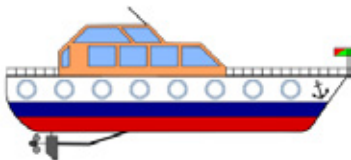
Kairioji pusė:



Dešinioji pusė:



Kokias spalvas galima matyti, žvelgiant pro du atitinkamus jachtos langus?



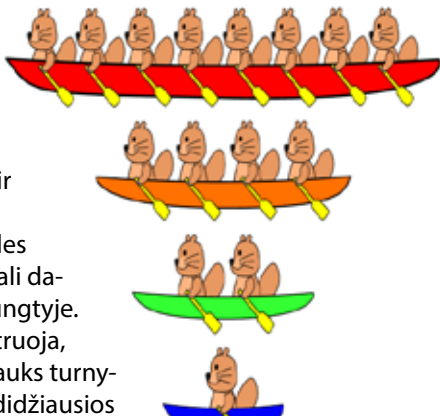
Kompiuteriai vartoja dvejetainę skaičiavimo sistemą, reiškiamą dviem simboliais – 0 ir 1. Šitaip galime koduoti ir jachtos langus: 0 – skaidrus, 1 – tamsintas. Su dvejetainiais skaičiais galima atlikti įvairius veiksmus: sudėtį, daugybą ir kt.



Bebrai rengiasi dalyvauti irklavimo turnyre.

Jie turi keturias valtis: aštuonvietę, keturvietę, dvivietę ir vienvietę.

Pagal turnyro taisykles kiekvienas bebras gali dalyvauti tik vienoje rungtyje. Bebrų treneris registruoja, kuriomis valtimis plauks turnyre. Jis pradeda nuo didžiausios valtės ir rašo 1, jei ji sukomplektuota, ir t. t. Pavyzdžiui, jei dalyvauja 10 bebrų, treneris užrašo 1010. Turnyre rengiasi dalyvauti 13 bebrų.



Ką užrašys treneris?

Dvejetainė sistema yra efektyvi, sprendžiant daugelį uždavinių. Naudojant du skaitmenis – 0 ir 1 – galima aprašyti visas turnyro rungtis, t. y. kiek ir kokių valčių dalyvaus. Kiekvieno skaitmens svoris dvejetainiame užrašė yra 2, pakeltas n-uoju laipsniu, kur n nusako skaitmens vietą, pradedant nuo skaičiaus galo.



Bebras eina aplink tvenkinį rodyklės kryptimi (žr. paveikslėlį).



Eidamas jis padarė keturias nuotraukas:



Kokia tvarka buvo darytos nuotraukos?

Kai reikia atlikti išsamią paveikslų analizę, kruopščiai apdoroti milijonus skaitmeninio paveikslų taškelių, kompiuteris kur kas pranašesnis už žmogų. Yra keletas informatikos sričių, kurioms svarbus paveikslų apdorojimas. Tai – skaitmeninė fotografija, vizualizavimo programos, erdvinis (3D) modeliavimas ir projektavimas, vizualizavimo ir animacijos tyrimai.

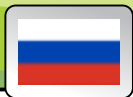


Dvejetainis laikrodis neturi rodyklių, nerodo skaitmenų. Laikas jame rodomas šviečiančiais taškais, kurių vertės reikia sumuoti. Viršutinė eilutė rodo valandas, apatinė – minutes. Laikrodis paveikslėlyje rodo 7:23.

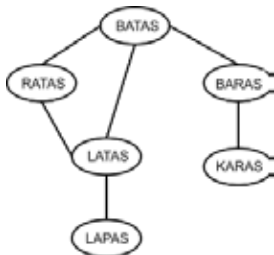


Kokį laiką rodančio laikrodžio visi taškai švies?

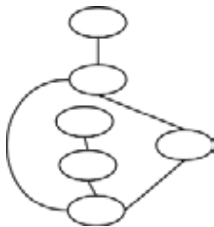
Duomenų vaizdavimas ir apdorojimas, kompiuterio naudojamų dvejetainių kodų vizualizavimas žmogui patrauklia forma – įdomios informatikos temos. Laikrodis vaizduoja dvejetainius skaičius, kurie labai svarbūs informatikoje – kompiuterių darbas, atminties įtaisai grindžiami dvejetainė skaičiavimo sistema.



Tomas turi surašyti žodžius ant kortelių ir sujungti jas gumine juosta. Mokytoja liepė sujungti bet kokius du žodžius, kurie skiriasi tik viena raide. Ką padarė Tomas, parodyta dešinėje.



Pabaigęs darbą, Tomas išėjo pažaisti į kiemą. Kai grįžo, nustebo pamatęs, ką padarė jo broliukas Petriukas: ištrynė visus žodžius. Dar ir korteles visiškai sumaišė (žr. kairėje).



Petriukas labai nusiminė, bet Tomas jį nuramino: „Nesijaudink, aš žinau, kur turi būti teisingi žodžiai, o tu man irgi gali padėti.“

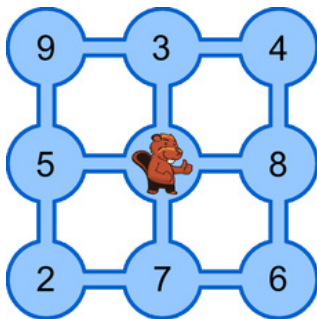
Kaip turi būti teisingai surašyti žodžiai kortelėse?

Sistemą, kurią Tomas sukonstravo iš kortelių ir guminių juostų, galima pavaizduoti grafu. Grafas yra struktūra, kurią sudaro viršūnės ir briaunos. Uždavinyje viršūnės atitinka kortelės, o briaunos yra jas jungiančios juostos.



Devyni tvenkiniai sujungti vienas su kitu kanalais. Bebras Bronius gyvena centriniame tvenkinyje, o jo draugai – kituose. Skaitmuo tvenkinyje rodo, kiek jame gyvena Broniaus draugų.

Bronius nusprendžia aplankyti draugus. Kiekvieną dieną jis gali perplaukti tik vieną kanalą, jungiantį du tvenkinius, todėl turi likti nakvoti tvenkinyje, į kurį atplaukė. Kitą dieną jis gali tęsti kelionę iš šio tvenkinio.



Bronius nori aplankyti kuo daugiau draugų.

Kiek skirtingų draugų gali aplankyti Bronius per keturias dienas, pradėdamas nuo namų? Jam nesvarbu, kur baigti kelionę.

Tvenkiniai ir kanalai – tai grafas, struktūra, kurią sudaro viršūnės (tvenkiniai) ir briaunos (kanalai). Grafe ieškoma reikalingo kelio – tokio, kad būtų galima aplankyti kuo daugiau draugų. Keliams grafuose ieškoti matematikai ir informatikai yra parengę daugybę algoritmų.

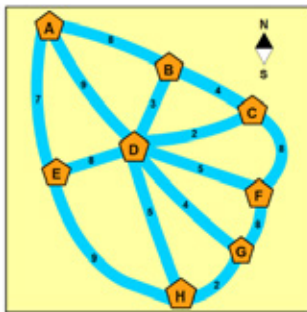


Bebrai stato užtvankas. Rąstų jie prisirenka miške. Juos plukdo upėmis pagal šias taisykles:

1. Bebrai surenka rąstus ant upės kranto ir sukrauna į plaustą.
2. Kiekvienoje upėje į plaustą galima krauti tik tam tikrą skaičių naujų rąstų.
3. Plaustai gali plaukti bet kuria upe tik pietų kryptimi.

Bebrams rūpi rasti geriausią maršrutą, kuriuo plaukdami surinktų daugiausia rąstų.

Žemėlapyje pavaizduota upių sistema ir nurodytas galimas rąstų skaičius kiekvienoje upėje. Kompasas rodo šiaurės ir pietų kryptis.

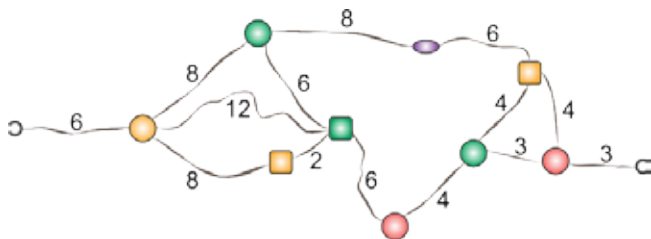


Jeį bebrai pradeda taške A ir nori pasiekti tašką H, kiek daugiausia rąstų jie gali plukdyti vienu plaustu?

Tai – grafų teorijos uždavinys, kai ieškoma ilgiausio kelio. Tuo tikslu sudaryta įvairių algoritmų. Ilgiausias kelias orientuotame grafe gali būti apskaičiuojamas, atliekant topologinį rikiavimą.



Bebriukė Elena veriasi karolius. Baigusi suabejojo, ar karoliai ne per trumpi. Skaičiai žymi siūlų ilgį tarp perlų. Susegimo žiedeliai yra kairėje ir dešinėje.



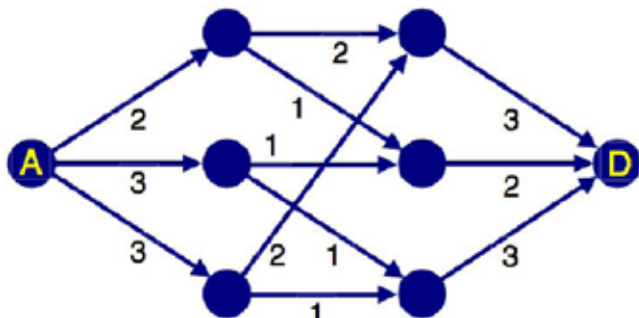
Koks bus susegtų karolių ilgis?

Trumpiausio kelio uždaviniai sprendžiami, ieškant trumpiausio kelio tarp dviejų grafo viršūnių (arba tinklo mazgų). Kelio ilgis priklauso nuo jį sudarančių atkarpų reikšmių sumos. Mažiausia kelio ilgio reikšmė ir yra trumpiausias kelias. Trumpiausiam keliui rasti sukurta daug algoritmų, vienas populiariausių – vadinamasis Dijkstros (Deikstros) algoritmas.



Bebrai plukdo rāstus kanalais pagal rodykles iš tvenkinio (A) į Debesnų pelkės ežerėlį (D).

Kiek daugiausia rāstų per minutę galima nuplukdyti kiekvienu kanalu, rodo ties juo parašytas skaičius.

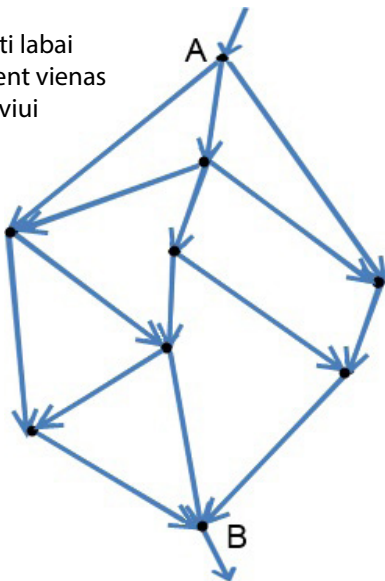


Kiek daugiausia rāstų gali būti nuplukdyta į Debesnų pelkės ežerėlį per vieną minutę?

Informatikoje ypač svarbūs algoritmų optimalumo klausimai, dažniausiai jie susiję su grafų apdorojimu. Vienas iš klasikinių informatikos optimizavimo uždavinių – maksimalaus srauto nustatymas orientuotame grafe.

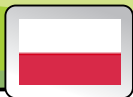


Būrys bebrų turi patikrinti labai šakotą upę: šiuo tikslu bent vienas bebras turi plaukti pasroviui kiekviena upės atšaka, pradėdamas taške A ir baigdamas taške B. Dėl stiprios tėkmės bebrai gali plaukti tik pasroviui ir įveikti tik vieną kelionę iš taško A į tašką B. Primename, kad kiekviena upės atšaka (žr. paveikslėlį) turi būti patikrinta.

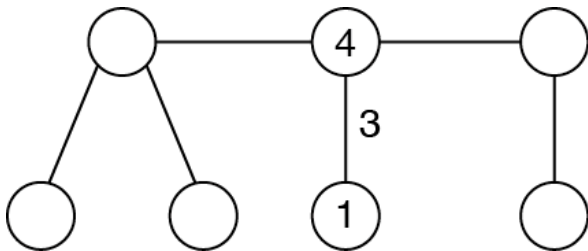


Kiek mažiausiai bebrų prireiks, kad būtų patikrinta visa upė?

Upė su atšakomis yra planariojo grafo pavyzdys. Tai yra grafas, kurio briaunos nesikerta ir turi kryptis (orientuotasis grafas). Tokiais grafais dažnai tenka analizuoti srautus. Vadina-masis daugiausia atšakų kertančios linijos paieškos algoritmas taikomas šiam srautų užda-viniui spręsti. Šios rūšies algoritmai yra svarbi informatikos studijų tema.



Pateiktą schemą sudaro apskritimai ir atkarpos. Į apskritimus reikia įrašyti skaičius 0, 1, ..., 6. Kai tik vienos atkarpos galuose įrašomi du skaičiai, atkarpa pažymima šių dviejų skaičių skirtumu. Schemoje skaičiai 1 ir 4 jau įrašyti į du apskritimus, juos jungianti atkarpa pažymėta 3.



Įrašykite likusius skaičius į tuščius apskritimus. Įsitinkite, kad visi atkarpas žymintys skaičiai yra skirtingi.



DVEJETAINIS MEDIS



Informatikoje naudojami įvairūs kodavimo būdai. Vienas iš jų – dvejetainis medis.

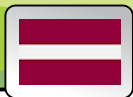
Vaikščiojimas medžiu žymimas komandomis:

- * pradžia (kamienas) – T
- * posūkis į kairę – K
- * posūkis į dešinę – D



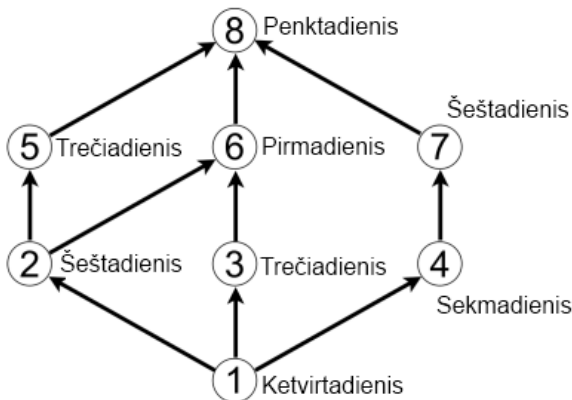
Parašykite komandų seką, vedančią prie žiedo.

Medis – tai viena efektyviausių struktūrų duomenims sistemškai sutvarkyti. Dvejetainis medis – tai visuma atkarpų, iš kurių prie kiekvienos galo galima jungti ne daugiau kaip dvi kitas atkarpas (vadinamas šakomis). Tokiu būdu galima taip sutvarkyti duomenis, kad jie būtų greitai randami. Dideliam medžiui aprašyti pakanka vos kelių ženklų.



Laukiniuose Vakaruose, kur gyvena bebrai kaubojai, pašto kompanija teikia paslaugas aštuoniose gyvenvietėse (jų numeriai – nuo 1 iki 8).

Plane prie kiekvienos gyvenvietės nurodyta pašto karietos išvykimo diena. Žinoma, kad ji išvyksta anksti rytą ir tos pačios dienos vakare pasiekia kitą gyvenvietę.



Kuriuo maršrutu siuntinys iš 1-osios gyvenvietės greičiausiai pasieks 8-ąją?

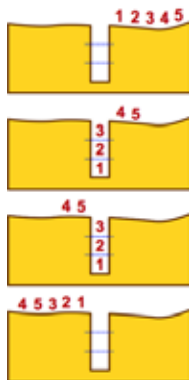
Tai grafas – struktūra, kurią sudaro viršūnės (gyvenvietės) ir briaunos (dienos). Grafe ieškoma reikalingo kelio – greičiausio maršruto iš pradinės gyvenvietės į galinę. Keliams grafuose ieškoti matematikai ir informatikai yra parengę daugybę algoritmų.



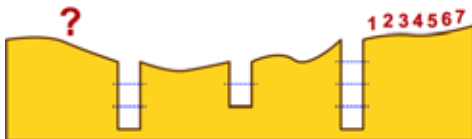
KIŠKIŲ DUOBĖ



Bebrai mėgsta vaikštinėti po mišką. Takeliai miške siauri, todėl bebrai turi eiti vorele vienas paskui kitą. Nedorėliai kiškiai išrausė duobių takeliuose. Bebrai duobę įveikia, sušokdami į ją vienas ant kito, kad likusieji galėtų pereiti. Kai likusieji bebrai pereina duobę, paskutinis ištraukia visus bebrus nuo viršutinio iki apatinio. Bebrai vėl keliauja vorele. Dešinėje parodyta, kaip penki bebrai įveikia duobę. Į duobę turėjo sušokti trys bebrai.



Galvokime apie septynių bebrų vorelę. Jie turi pereiti tris duobes. Pirmoje duobėje telpa keturi bebrai, antroje – du, o trečioje – trys bebrai.



Kaip bus išsirikiavę bebrai, kai įveiks visas tris duobes?

Apdorojant duomenis, pavyzdžiui, atliekant jų rikiavimą, svarbu pasirinkti tinkamas duomenų struktūras. Šiam uždaviniui tinkama vadinamoji dėklo duomenų struktūra (angl. *stack*). Dėklo elementai gali būti dedami tik ant viršaus ir imami tik nuo viršaus. Angliškai dėklas dar vadinamas LIFO duomenų struktūra (*last in first out*) – tai reiškia, kad paskutinis į struktūrą padėtas elementas paimamas pirmas.



Du draugai dirba kepykloje. Lina kepa riestainius, traukia iš krosnies po tris skirtingų formų ir kabina juos ant kartelės iš dešinės: pirmiausia pakabina A riestainį, tada B, paskiau O. Linas pardavinėja riestainius. Nuo kartelės jis visada ima dešiniausią riestainį. Lina kepa riestainius greičiau, negu Linas juos parduoda.

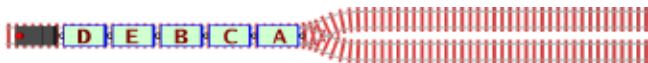


Kiek mažiausiai riestainių pardavė Linas, jei kartelė atrodo taip, kaip parodyta paveikslėlyje?

Šiuo uždaviniu iliustruojama duomenų struktūra – vadinamasis dėklas (angl. *stack*). Dėklo elementai gali būti dedami tik ant viršaus ir imami tik nuo viršaus. Angliškai dėklas dar vadinamas LIFO duomenų struktūra (*last in first out*) – tai reiškia, kad paskutinis į šią struktūrą padėtas elementas paimamas pirmas.

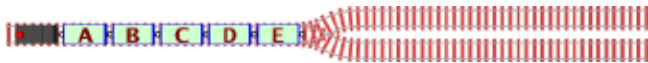


Bebrų traukinių stotyje stovintis traukinys atrodo štai taip:



Garvežys gali judėti pirmyn ir atgal. Jis taip pat gali pastumti ar patraukti ne daugiau kaip penkis prijungtus vagonus. Kiekvienas kartas, kai vagonai yra prijungiami arba atskiriami, skaičiuojamas kaip viena operacija.

Kiek mažiausiai reikia operacijų, kad traukinys atrodytų taip:



Šiuo uždaviniu vaizduojama vadinamoji dėklo duomenų struktūra (angl. stack). Dėklo elementai gali būti dedami tik ant viršaus ir imami tik nuo viršaus. Angliškai dėklas dar vadinamas LIFO duomenų struktūra (last in first out) – tai reiškia, kad paskutinis į struktūrą padėtas elementas paimamas pirmas. Dėklo pavyzdžiai gali būti kaladėlių bokšto statyba ir griovimas, knygų krovimas į siaurą dėžę (vienu bokštu) ir išėmimas iš jos.



Bebriukas turi žaislinį robotą, kuris klauso dviejų komandų:



paeina 10 cm į priekį;



pasisuka į dešinę ketvirtadalį apskritimo, t. y. 90 laipsnių.

Kaip priverstum robotą pasisukti į kairę 90 laipsnių?

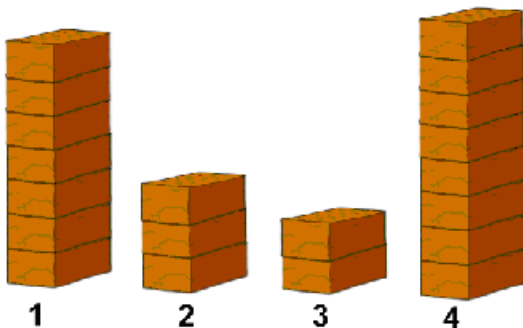
Informatikas programuotojas turi galvoti apie objekto būsenas ir galimus veiksmus. Programuojamos kompiuterinės sistemos veiksmus gali riboti techninės galimybės. Pavyzdžiui, robotas gali neatpažinti pasisukimo į kairę komandos ir pasisukti į kitą pusę. Informatikoje mus domina, kaip turimam būsenų rinkiniui parinkti tinkamus veiksmus.



PLYTŲ RIETUVĖS



Yra keturios plytų rietuvės. Robotas gali perkelti plytas iš vienos rietuvės į kitą. Viena komanda robotui yra trijų skaičių kombinacija skliaustuose. Pavyzdžiui (1, 3, 4) reiškia: iš 1-osios rietuvės paimti 3 plytas ir dėti jas ant 4-osios rietuvės. Nagrinėkime, kas parodyta paveikslėlyje.



Užrašykite komandas robotui perkelti plytas taip, kad visos rietuvės būtų vienodos.

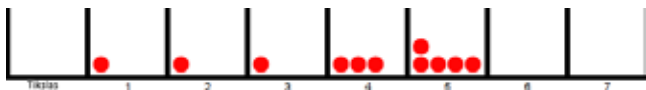
Algoritmuose ir programose svarbi komandos sąvoka. Komanda – tikslus, aiškus ir viena-reikšmis nurodymas, kurį gali atlikti kompiuteris, robotas ar automatas. Komandos vartojamos kompiuterių ar kitokių prietaisų sąsajose, algoritminėse ir programavimo kalbose.



AKMENUKŲ ŽAIDIMAS



Žaidimo tikslas – perkelti visus akmenukus į kairįjį lovelį (užrašyta **Tikslas**). Reikia laikytis šios taisyklės: lovelį su numeriu nuo 1 iki 7 galima ištuštinti, jei jo akmenukų skaičius yra lygus lovelio numeriui, tada jo akmenukai paskirstomi po vieną į visus kairiau esančius lovelius. Žaidimas laimimas, kai visi akmenukai sukeliami į lovelį **Tikslas**. Pateikiama situacija, kai žaidimas gali būti laimimas.



Kuria tvarka reikia tuštinti lovelius?

Sprendžiant šį uždavinį, lemiamą vaidmenį vaidina veiksmų seka – tai būdinga bent keliolikai informatikos problemų. Pateikiama laimėjimo strategija yra ne kas kita, kaip sąrašas algoritminių žingsnių, kuriuos atliekant žaidimas gali būti laimėtas, jei iš viso įmanoma laimėti.

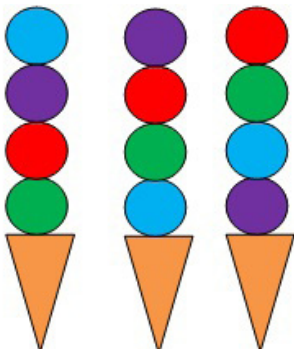


LEDŲ GAMYBOS MAŠINA



Ledų gamybos mašina tam tikru būdu deda skirtingų spalvų ledų rutuliukus. Į vieną vaflinį indelį dedami keturi ledų rutuliukai.

Paveikslėlyje pavaizduotos trys mašinos pagamintos ledų porcijos:

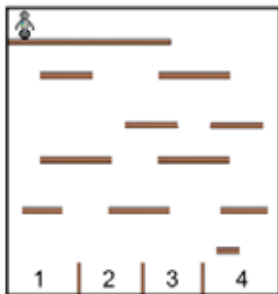
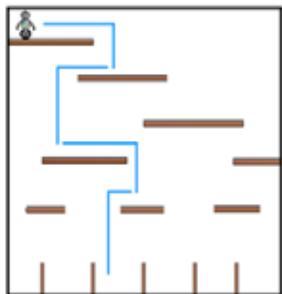


Kokia tvarka mašina sudės rutuliukus kitoje ledų porcijoje?

Siekdami kokio nors tikslo, atliekame tam tikrus veiksmus – šitaip konstruojame algoritmus. Algoritmas yra problemos (uždavinio) sprendimas žingsnis po žingsnio. Šiame uždavinyje reikia atkurti anksčiau pritaikytą algoritmą ir paskui numatyti kitą būsimą rezultatą – tai svarbu informatikoje.



Robotas krinta labirinte nuo viršutinės lentynėlės ant kitų lentynėlių, kol pasiekia vieną iš dėžučių apačioje. Lentynėlės galą, nuo kurio kris, jis pasirenka kas kartą priešingą: jei krito nuo dešiniojo galo, tai, pasiekęs kitą lentynėlę, kris nuo jos kairiojo galo ir t. t. (žr. pirmą pav.).



Į kurią dėžutę įkris robotas antrame labirinte?

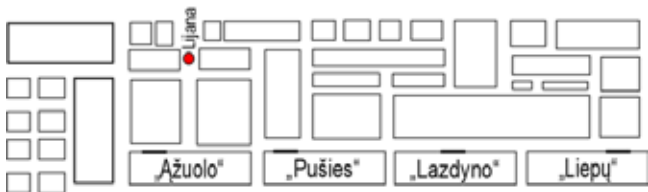
Robotas atlieka nurodytas komandas, kitaip tariant, jis taip suprogramuotas, toks jo veikimo algoritmas. Algoritmai – viena rimčiausių informatikos sričių. Roboto veikimo algoritmą galime perprasti, nagrinėdami pirmąjį paveikslėlį. Suprastas algoritmas vykdomas. Tai atliekama antrame paveikslėlyje.



Lijana apsistojo viešbutyje Beбриškių mieste. Ji norėjo pamatyti garsiąją Bebro statulą ir ją nufotografuoti, todėl vadovavosi viešbučio darbuotojų instrukcijomis:

1. Išėjusi pro viešbučio duris, iškart suk į kairę.
2. Pirmose dviuose sankryžose eik tiesiai.
3. Trečioje sankryžoje suk į dešinę.
4. Eik tiesiai. Pirmoje sankryžoje suk į kairę.
5. Eik tiesiai. Pirmoje sankryžoje suk į dešinę.

Žemėlapyje parodyti netoliese esantys viešbučiai:



Kuriame viešbutyje apsistojo Lijana?

Instrukcijų ar komandų rinkinio pateikimas ir tikslus vykdymas – pagrindinė algoritmų, sudarančių informatikos pagrindą, idėja. Algoritmas yra uždavinio (problemos) sprendimas žingsnis po žingsnio. Žingsniai turi būti kuo mažesni ir tikslesni, kad gautume sprendimą. Algoritmai užrašomi (koduojami) programavimo kalbomis ir vykdomi transliatoriais (kompilatoriais).



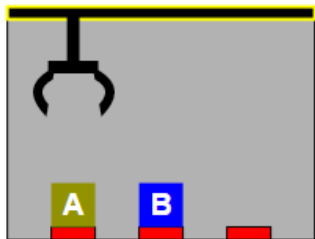
DARBAS KRANU



Bebras dirba kranininku. Aikštelėje pastatytos dvi dėžės – A ir B. Iš pradžių dėžė A stovi ant 1-osios pakylės, dėžė B – ant 2-osios.

Kranas gali vykdyti šešias komandas: VIRŠUN, ŽEMYN, DEŠINĖN, KAIRĖN, SUGRIEBTI, PALEISTI. Jos nurodomos spaudžiant komandų mygtukus.

Išrašykite komandų seką, kurią vykdydamas bebras sukeistų dėžes: A turi būti ant 2-osios pakylės, o B – ant 1-osios.



Uždaviniui išspręsti būtinas nuoseklusis algoritmas, kuriuo du objektai būtų pastumti per dvi pozicijas. Tai padaryti reikalinga trečioji pozicija. Jei turėtume daugiau kranų, kurie veiktų vienu metu ir vienas kitam netrukdytų, galėtume sudaryti lygiagrečiųjų algoritimą, kuriam nereikėtų trečiosios pozicijos.

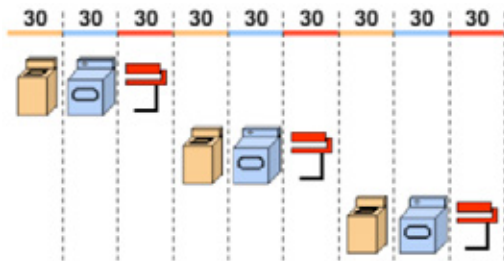


SKALBYKLA



Bebras Bronius įsteigė skalbyklą. Įsigijo tris mašinas: skalbimo, džiovinimo ir lyginimo. Kiekviena mašina jungiama į elektros tinklą atskirai ir veikia pusę valandos nepriklausomai nuo kitų. Vieno kliento visoms trimis procedūroms atlikti reikia 90 minučių. Trims klientams atliekant visas skalbyklos procedūras vienam po kito, iš viso reikėtų 270 minučių.

Kartą atvyko trys labai užimti ir skubantys beabriukai. Kiekvienas iš jų turi gana daug skalbinių, tad kiekvienas skalbimo mašiną naudoja atskirai. Jie labai norėtų baigti skalbti kiek galima greičiau.



Kiek minučių užtruks, kol visi trys beabriukai baigs visas skalbyklos procedūras?

Centrinis procesorius veikia panašiai kaip skalbykla. Kiekvienas prietaisas naudojamas, kai tik yra laisvas. Konstruojant centrinį procesorių, svarbus grandininis duomenų apdorojimas. Tai – duomenų apdorojimo būdas, kuriuo duomenys perkeliama iš vieno juos apdorojančio proceso į kitą. Kiekvienas procesas tuo pat metu apdoroja skirtingą duomenų porciją.



Robotas vabaliukas juda žaidimų aikštelės kvadrateliais pagal tokias taisykles:

- Pradžia – atsitiktinai pasirinktas langelis.
- Judama pagal rodykles, esančias langelyje, kuriame tupi robotas vabaliukas: paeinama rodoma kryptimi per tiek langelių, kiek rodyklių nupiešta (pavyzdžiui, per vieną langelį, jei yra viena rodyklė, per du langelius, jei dvi, ir t. t.).
- Judėdamas robotas ignoruoja langelių, kuriuos praeina nesustodamas, rodykles.
- Robotas vabaliukas juda tol, kol išeina už aikštelės arba pasiekia langelį be rodyklių (E stulpelis).

Iš kurių A stulpelio langelių robotas vabaliukas turėtų pradėti, kad ėjimą baigtų kuriame nors E stulpelio langelyje?

	A	B	C	D	E
1	→ → → →	→ →	↓ ↓	↓ ↓	
2	↓ ↓	→	↓ ↓ ↓	→	
3	→	↑	↓	←	
4	→	↑ ↑ ↑	→ →	→	

Uždavinyje žaidimo aikštelę galime laikyti paprasta programa: yra komandos, nusakančios veiksmus, nurodyta pradžia ir pabaiga. Programą galima vykdyti – robotui taisyklės aiškios.




VITRAŽAS



Robotas puošia langus vitražais iš spalvoto stiklo gabaliukų. Kiekviename vitraže kartojamas vienas toks pat fragmentas



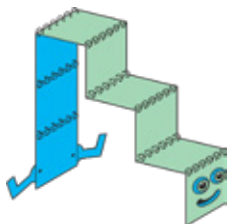
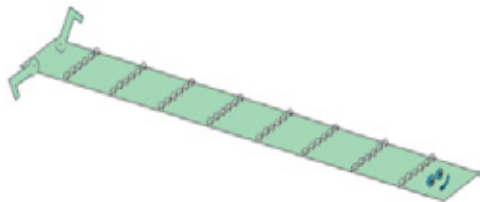
3 stulpelių lango vitražą sudaro 5 fragmentai	5 stulpelių lango vitražas atrodo šitaip:

Kiek mėlynų stiklo gabaliukų  reikia, norint sukurti 7 stulpelių lango vitražą?

Robotas veikia pagal programą. Programa – viena esminių informatikos sąvokų. Programos funkcionalumas aprašomas, pateikiant porą nedidelių pavyzdžių. Sprendėjui reikia perskaityti funkcinę uždavinio specifikaciją ir paversti ją programa, t. y. sugalvoti algoritmą, kaip galima kurti vitražą, o tada apskaičiuoti prašomą rezultatą.



Bronius konstruoja juostinį robotą iš vienodų kvadratinų plokščių. Gretimas plokštes jis jungia vyriais.



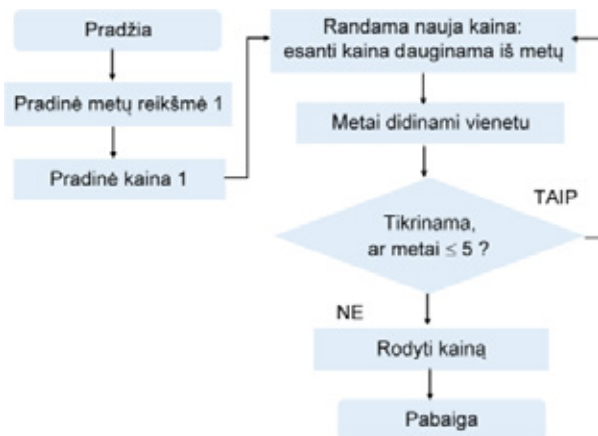
Bronius roboto formą gali keisti – lenkti jį jungimo vietose. Pavyzdžiui, juostinis robotas gali pavirsti laiptiniu robotu. Paveikslėlyje matome trijų pakopų laiptus, sukonstruotus iš 9 plokščių.

Kiek kvadratinų plokščių reikia, norint sukonstruoti 7 pakopų laiptinį robotą?

Norint sukonstruoti laiptus iš juostos, reikia sugalvoti žingsnius, kaip tai daroma, kitaip tariant, sudaryti algoritmą. Čia prireikia pasikartojančių veiksmų, jie programavime vadinami ciklais. Algoritmas turi tikti įvairiems duomenims – reikia siekti apibendrintų veiksmų.



Bebrijoje sūrio kaina priklauso nuo jo brandinimo laiko. Kainai apskaičiuoti naudojamos struktūrinės schemos. Štai ši schema skiriama 5 metus brandinto sūrio kainai rasti. Mažiausia sūrio kaina – 1 pinigais.



Kiek kainuoja 5 metus brandintas sūris?

Struktūrinė schema – supaprastintas būdas aprašyti programą, nevarojant programavimo kalbos. Schemoje nurodoma pradžia, tolesnius veiksmus žymi rodyklės, kol pasiekama pabaiga. Ciklas – veiksmų seka, kurią kompiuteris turi atlikti keletą kartų. Šiuo atveju tai – sąlyginis ciklas, t. y. ciklas veiksmų, kartojamų tol, kol tenkinama nurodyta sąlyga.



Šeši beabriukai žaidžia žaidimą. Pradžioje jie stovi apskritimuose, kaip parodyta paveikslėlyje. Kiekviename apskritime yra po balioną su skaičiumi nuo 1 iki 6, rodantį, į kurį apskritimą beabriukas turės pereiti (vadinama kryptimi, kiekvienam apskritimui ji skirtinga). Po signalo kiekvienas beabriukas eina apskritime nurodyta kryptimi. Beabriukai eina ratu tol, kol visi sustoja į savo pradinius apskritimus.



Kiek ratų bus apeita iki žaidimo pabaigos?

Ciklas – algoritmo arba programos kartojamų veiksmų dalis, kuria aprašomi veiksmai ir jų kartojimo sąlygos. Visos programavimo kalbos turi ciklų komandas ir netgi kelių rūšių.



KETURIOS VARLĖS



Robotas burtininkas juda plytelių taku pagal tam tikras komandas:



– robotas paeina į priekį vieną žingsnį (ant kitos plytelės);



– robotas padeda varlę ant plytelės priešais save.

Pasikartojančius veiksmus galima nusakyti trumpiau:



– robotas 4 kartus kartoja komandą „vienas žingsnis į priekį“, t. y. žengia keturis žingsnius į priekį;

Jei kartojama ne viena komanda, vartojami skliaustai:



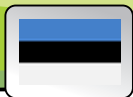
– robotas 4 kartus kartoja dvi vienodas komandas: „vienas žingsnis į priekį“ ir

„vienas žingsnis į priekį“ (iš viso paeinama 8 žingsnius į priekį).

Beje, ant vienos plytelės gali stovėti ir robotas, ir varlė, net keletas varlių.

Užrašykite komandas (programą) robotui sudėlioti keturias varles į eilę (ant keturių gretimų plytelių).

Norint išsiaiškinti, kaip veikia programa, pirmiausia reikia suprasti, kaip teisingai derinti įvairias komandas. Ypač tai svarbu, kai komandos kartojamos, tai yra sudaromas ciklas: reikia išmolti atpažinti, kurios komandos yra ciklo viduje, kurios išorėje, kada ciklas pradamas ir baigiamas. Algoritminėse ir programavimo kalbose ciklu išreiškiami kartojimo veiksmai.



Bibliotekininkas kas vakarą turi surikiuoti skaitytojų sumaišytus enciklopedijų tomus didėjimo tvarka. Jis tai norėtų atlikti per kiek galima mažiau žingsnių.

Vieną žingsnį sudaro trys veiksmai:

- * paimti iš lentynos vieną knygą,
- * pastumti kita ranka keletą knygų į kairę arba į dešinę,
- * padėti laikomą knygą į atsiradusią tuščią vietą.

Pavyzdyje parodyta, kaip vienu žingsniu galima surikiuoti penkis tomus:

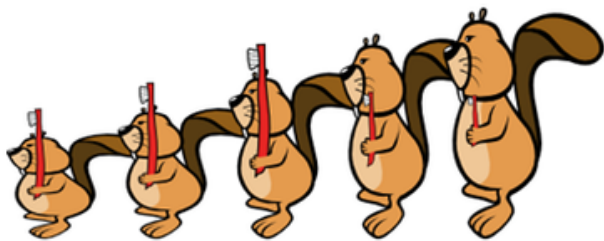


Vieną dieną bibliotekininkas rado 9 sumaišytus enciklopedijų tomus. Kiek mažiausiai žingsnių reikės, norint išrikiuoti visus šiuos tomus?

Turimus duomenis būtina tvarkyti. Sprendžiant šį uždavinį, reikia išmanyti duomenų rikiavimą ir rikiavimo taisykles: surandamas didžiausias didėjantis tomų numerių posekis ir šie tomai paliekami lentynoje. Taip pat čia esama sąlygų, kurios informatikoje dažnai vadinamos ribojimais. Ribojimai reikalingi, atliekant programas arba ieškant problemos sprendimo.



„Neskubėkite!“ – sako mama bebrė. „Eva ir Lina, apsikeiskite šepetėliais! Ana ir Lina, irgi apsikeiskite šepetėliais!“ O tada mama sutriko...



Ana Benas Lina Dainius Eva

Kuri pora dar turi apsikeisti šepetėliais, kad kiekvienas bebras turėtų savąjį?

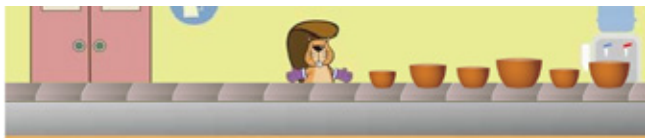
Programuojant dažnai tenka keisti du duomenis vietomis kompiuterio atmintyje. Tai viena iš pagrindinių programavimo operacijų. Pavyzdžiui, rikiuojame skaičius pagal jų dydį (kaip ir šepetėlius). Rikiavimo metodų yra daug. Vienas paprastesnių – duomenų keitimas poromis: mažiausią skaičių sukeisti su esančiu pirmoje vietoje, paskui tolesnį mažiausią skaičių sukeisti su esančiu antroje vietoje ir taip toliau, kol didžiausias skaičius tampa paskutinis.



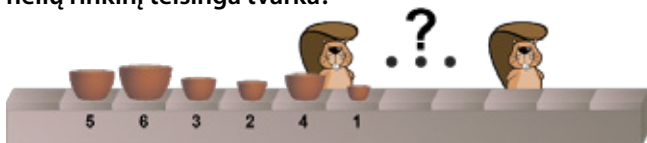
DUBENĖLIŲ KONVEJERIS



Įmonė gamina 6 skirtingų dubenėlių rinkinius. Dubenėliai slenka konvejerio juosta vienas paskui kitą iš kairės į dešinę. Visi 6 skirtingi rinkinio dubenėliai keliauja konvejeriu kartu, tačiau jie sudėti atsitiktine tvarka. Rengiantis pakuoti dubenėliai turi būti išrikiuoti nuo mažiausio iki didžiausio. Dubenėlius rikiuoja įmonės darbuotojai, sustoję palei konvejerį. Dubenėlių rinkiniui slenkant pro darbuotoją, šis apkeičia dubenėlių poras, išsidėsčiusias neteisinga tvarka.



Kiek darbuotojų reikia, norint išrikiuoti pateiktą dubenėlių rinkinį teisinga tvarka?

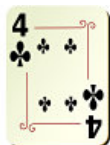


Automatinis duomenų apdorojimas vyksta daug lengviau, jei duomenys išdėstomi pagal tam tikrus kriterijus – išrikiuojami. Informatikai deda daug pastangų, ieškodami naujų, geresnių rikiavimo algoritmų. Aprašytas dubenėlių rikiavimo metodas vadinamas burbuliniu rikiavimu. Šio metodo esmė – tikrinamas objektų sąrašas ir kaskart apkeičiami gretimi objektai, išsidėstę neteisinga tvarka. Sąrašas laikomas išrikiuotu, kai, tikrinant jį visą, nereikia nieko keisti.



Bebrų mokykloje per pertraukas mokiniai žaisdami kortomis mokosi rikiavimo algoritmų. Kortos išdėstomos bet kaip, jas reikia surikiuoti didėjančiai. Leidžiama sukeisti vietomis dvi šalia esančias kortas. Kortų spalva ir rūšis nesvarbi. Kortų, išdėstytų reikiama tvarka, vietomis sukeisti negalima.

Kiek ėjimų truks žaidimas, kai kortos išdėstytos šitaip?



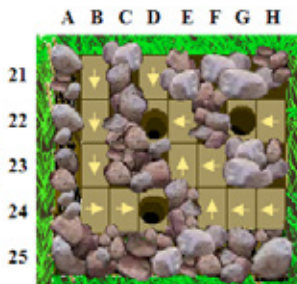
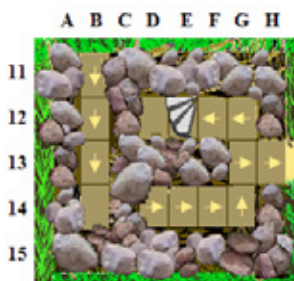
Duomenų rikiavimas yra viena iš svarbiausių informatikos temų. Surikiuotų duomenų paieška daug greitesnė. Pavyzdžiui, miesto gyventojų pavardės duomenų banke surikiuotos abėcėlės tvarka, todėl nesunku gauti informacijos apie bet kurį miesto gyventoją.



Bebras pastatė dviejų aukštų labirinto modelį. Robotė skruzdė ropinėja akmenukų sienų labirinte pagal taisykles. Kai atsistoja

- ant rodyklės – parpoja vienu langeliu rodyklės kryptimi;
- ant duobės (apskritimo) – nusileidžia į po ja esantį langelį;
- ant laiptų – kyla į antro aukšto langelį;
- ant tuščio langelio – sustoja (baigia darbą).

Pirmame aukšte yra vienas įėjimas – B11 langelis ir vienas išėjimas – H13 langelis. Antrame aukšte yra tik keturi langeliai: B21, D21, H22, H24.



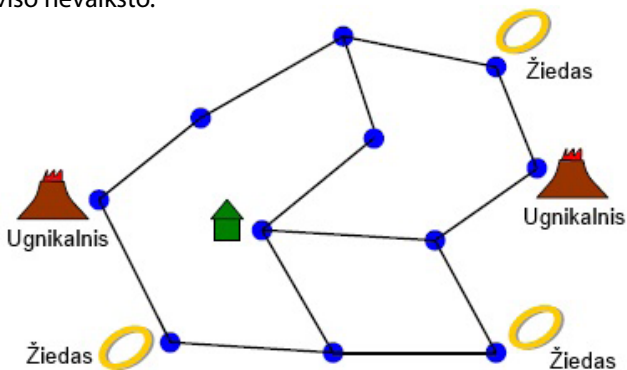
Pro kurį įėjimą įropojusi robotė skruzdė sėkmingai įveiks kelią ir išropos iš labirinto?

Čia pateikiamas veikslių algoritmas. Jį reikia įgyvendinti, atsižvelgiant į nurodytas sąlygas (reikalavimą, kad skruzdė robotė išropotų iš labirinto). Universalus metodas – visų galimų variantų paieška. Siekiama optimizuoti, atmetant akivaizdžiai neoptimalias šakas, svarbu tik nepamesti teisingų variantų. Tobulinami išsamios baigtinės paieškos įgūdžiai.



Bebras užsimanė pramogų. Išėjęs iš namų jis turi surinkti tris žiedus ir įmesti juos į ugnikalnio lavą. Tada grįžta namo.

Bebras turi žemėlapij. Kiekvienas kelias (atkarpa tarp dviejų taškų) įveikiamas per vieną dieną. Tuo pačiu keliu bebras gali eiti kad ir kelis kartus, o kai kuriais keliais iš viso nevaikšto.



Kiek mažiausiai dienų prireiks bebrui, kad surinktų tris žiedus, tada įmestų juos į vieną iš ugnikalnių ir grįžtų namo?

Universalus metodas – visų galimų variantų paieška ir fiksavimas. Siekiama optimizuoti, atmetant akivaizdžiai neoptimalius sprendinius, svarbu tik nepamesti teisingų variantų. Atliekant šiuos uždavinius, tobulinami išsamios baigtinės paieškos įgūdžiai.

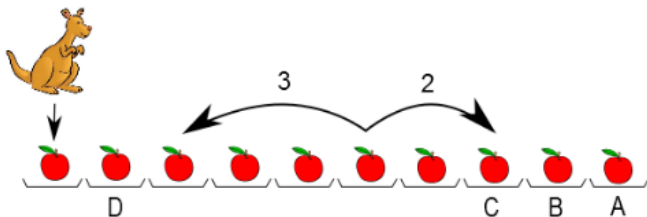


ŠOKLUS KENGŪRIUKAS



Eilėje yra 10 lėkščių. Kiekvienoje lėkštėje yra po vieną obuolį.

Kengūriukas Tomas mėgsta šokinėti. Pirmiausia jis šoka prie kairiojo krašto esančios lėkštės. Po to kiekvienu šuoliu jis gali šokti arba per dvi lėkštes pirmyn, arba per tris lėkštes atgal. (Paveikslėlyje rodyklės vaizduoja galimus šuolius nuo vienos iš lėkščių.) Tomas šoka tik prie lėkščių, kuriose yra obuolių. Prišokęs prie lėkštės, pasiima iš jos obuolį.



Jei Tomas susirinks visus obuolius, kurį paims paskutinį?

Vienas iš šio uždavinio sprendimo būdų yra visų galimų lėkščių sekų analizė ir radimas sekos, atitinkančios galimus šuolius. Kiekviena galima seka vadinama kėliniu, o tų kėlinių yra labai daug. Šis sprendimo metodas vadinamas visišku perrinkimu ir reikalauja daug laiko.



Bebriukas žaidžia miške, kirtavietėje. Jis užsilipo ant rudo kelmo ir, šokinėdamas kelmais, nori patekti ant pilkojo kelmo.

Vienu šoliu bebriukas gali peršokti ant gretimo kelmo kairėje arba apačioje (įstrižai šokti negalima).



Kiek yra skirtingų kelių, kuriais bebriukas gali pasiekti pilkąjį kelmą?

Informatikos uždaviniuose neretai tenka taikyti visiško per rinkimo metodą (angl. *brute force*). Tai labai bendras algoritminis uždavinių sprendimo metodas, kai sistemaiškai perrenkami visi galimi sprendimų variantai ir kiekvienas jų tikrinamas, ar tenkina uždavinio sąlygas. Galimas ir kombinatorikos taikymas.



IŠALKĖS BEBRAS



Išalkęs bebras nori vykti į parduotuvę riešutų. Kad būtų greičiau, bebras važiuos dviračiu. Jis pasirengė schemą ir bando išsirinkti tinkamiausią maršrutą.

Yra trys parduotuvės ir keletas kelių į jas nuvykti. Bebras žino, kad yra lengvesnių ir sunkesnių kelių atkarpų – tai jis sužymėjo skirtingos spalvos kvadratėliais. Pateikta schema ir paaiškinimai:



parduotuvė



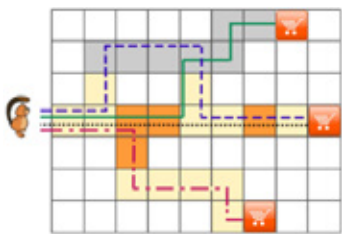
keliavimo laikas:
2 minutės



keliavimo laikas:
1 minutė



keliavimo laikas:
5 minutės



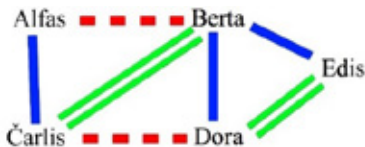
Kuriuo keliu Bebras greičiausiai pasieks parduotuvę?

- * Mėlynuoju keliu (punktyrinė linija)
- * Žalioju keliu (ištisinė linija)
- * Juodoju keliu (taškinė linija)
- * Raudonoju keliu (brūkšninė-taškinė linija)

Sisteminis galimybių perrinkimas reikalingas, sprendžiant informatikos uždavinius. Perringimą stengiamasi optimizuoti, įvertinant situaciją ir atmetant akivaizdžiai netinkamus sprendimus.



Bebrai Alfas, Berta, Čarlis, Dora ir Edis kine visada sėdi vienoje eilėje. Jų draugystės ryšius galima pavaizduoti tokia schema:



Draugystės lygmenį galima aprašyti šia „gerovės taškų“ lentele:

Jungtis	Įvertinimas	„Gerovės taškai“
Nėra jokios	„nepažįstami“	0
-----	„pažįstami“	+1
—————	„draugai“	+2
=====	„[simylėjėliai“	+3

Jeigu patys artimiausi bebrai susėda vienas šalia kito, visi jaučiasi geriau, t. y. visos grupės „gerovės taškų“ suma yra didesnė. **Kokia tvarka turi susėsti draugai, kad grupės „gerovės taškų“ suma būtų didžiausia?**

Schema aiškiai rodo ryšį su grafu: viršūnės – bebrai, briaunos – jų draugystės ryšiai. Sprendžiama optimizavimo problema. Tačiau galima taikyti ir visišką perrinkimą (angl. *brute force*) ar įžvelgti tam tikrus dėsningumus (heuristinis metodas).

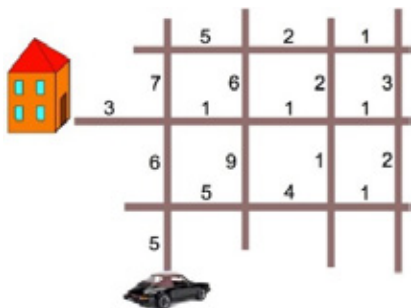


SUKTI KAIRĖN NEGALIMA!



Mieste – transporto spūstys. Mašinos rieda taip arti viena kitos, kad net negalima pasukti kairėn...

Tėtis, baigęs darbą, skuba automobiliu namo. Pateiktas gatvių planas. Skaičius šalia gatvės rodo, kiek minučių trunka važiuoti šia gatve. Greičiausio kelio namo tėtis ieško GPS įrenginiu.



Kiek mažiausiai laiko tėtis užtruks, važiuodamas namo?

Sprendimą, kuriuo keliu važiuoti, reikia priimti kiekvienoje sankryžoje, remiantis kriterijais: laiku (reikalingu atkarpai įveikti) ir kryptimi (važiuoti tiesiai ar dešinėn). Sistemingas galimybių perrinkimas reikalingas, sprendžiant informatikos uždavinius. Perrinkimą stengiamasi optimizuoti, įvertinant situaciją ir atmetant akivaizdžiai netinkamus sprendimus.



VALČIŲ KROVA



Žvejai Simas ir Rimas turi po valtį: „Lisa 1“ ir „Lisa 2“. Simas ir Rimas – dvyniai ir sveria vienodai. Kiekviena valtis gali plukdyti tik 300 kilogramų prekių, neskaitant vieno iš ta valtimi plaukiančių brolių.

Broliai gavo užsakymą plukdyti uždaras statines, pilnas geriamojo vandens. Broliai nori prikrauti į valtis kuo daugiau statinių, nes jų atlyginimas priklauso nuo krovinio svorio.



Kurias statines Simas ir Rimas pakraus į valtis?

Daugelyje gyvenimo sričių žmonėms reikia optimizuoti įvairius dalykus. Tam sukurta daug optimizavimo algoritmų. Neretai naudojamas „godusis“ algoritmas, kai kiekvienu žingsniu imamas pelningiausias (didžiausias) komponentas. Deja, daugeliu atvejų „godumo“ principas nepateikia optimalaus sprendinio ir reikia ieškoti tobulesnės sprendimo strategijos. Todėl informatikai nuolat kuria naujus algoritmus.



PAŠTAS



Paštininkas bebras turi nunešti pašto siuntas į vienos gatvės namus. Gretimi namai vienas nuo kito nutolę 1 km atstumu. Bebrui mokama už atstumą, kurį jis nueina, nešdamas siuntas į visus namus. Taigi bebras nori nueiti kuo daugiau kilometrų. Jis gali pradėti nuo bet kurio namo ir bet kuriuo namu baigti. Siuntos pristatomos į kiekvieną namą po kartą.

Kuria tvarka bebras turėtų aplankyti penkis namus, kad nueitų ilgiausią atstumą?

				
1	2	3	4	5

Optimizavimu vadinama paieška uždavinio sąlyga apibrėžtoje aibėje tokio elemento, kuriam kriterijaus reikšmė būtų minimali arba maksimali.

Namų aplankymo tvarkos, tai yra ilgiausio paštininko kelio, suradimas – tai uždavinio sprendimo optimizavimo pavyzdys. Intuityvūs šio uždavinio sprendimo metodai atitinka „godžiuosius“ algoritmus.



Voverė nuraškė du riešutus nuo lazdyno viršūnės. Ji nori surinkti kiek įmanoma daugiau riešutų, todėl šokinėja nuo šakos ant šakos nurodytomis kryptimis.

Kiek daugiausia riešutų voverė gali surinkti, šokdama du šuolius nuo viršūnės?



Universalus metodas – visų galimų variantų paieška. Siekiama optimizuoti, atmetant aki-vaizdžiai neoptimalias šakas, svarbu tik nepamesti teisingų variantų. Tobulinami išsamios baigtinės paieškos įgūdžiai. Šis uždavinys gali būti laikomas dinaminio programavimo metodo taikymo pavyzdžiu.



Senovės japonų kalendorius susideda iš 60 metų ciklo. Metai numeruojami nuo 1 iki 60 ir grupuojami poromis, kaip parodyta lentelėje. Metų poros nuspalvintos iš eilės: žaliai, raudonai, geltonai, baltai ir juodai (toliau vėl kartojama):

1	2	11	12	21						51	52
3	4	13	14							53	54
5	6	15	16							55	56
7	8	17	18							57	58
9	10	19	20							59	60

Žinoma, kad 1984 metai yra pirmieji 60 ciklo metai.
Kokios spalvos yra 2012 metų langelis?

Metai spalvinami pagal tam tikras taisykles, galima sakyti, algoritmą. Algoritmai – viena rimčiausių informatikos sričių. Metų porų spalvinimą galime perprasti, nagrinėdami paveikslėlį. Kartojamiems veiksams atlikti ir skirtas ciklas – viena svarbiausių algoritmo konstrukcijų.



svajonių suknelė



Kristina nori nusipirkti svajonių suknelę, kuri būtų:

- trumpomis rankovėmis,
- su daugiau nei trimis sagomis,
- papuošta žvaigždėmis ant rankovių.

A.



B.



C.



D.

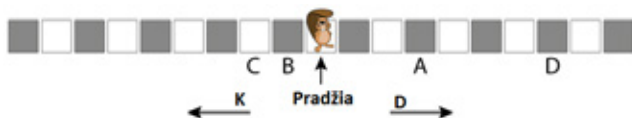


Kurioje parduotuvėje Kristina galėtų nusipirkti svajonių suknelę?

Ieškoma suknelė turi atitikti tris požymius – sąlygas. Programuojant ir sudarant algoritmus, sąlygos yra labai svarbios. Kiekvieną sąlygą atitinka skirtingi veiksmai. Sąlygos gali būti tikrinamos po vieną arba jungiamos loginiais operatoriais (AND, OR, NOT). Čia reikalingas AND loginis operatorius, kuris pateikia teisingą rezultatą, kai visos sąlygos yra tenkinamos.



Šis bebras keistai šokinėja: pradeda nuo vidurinio langelio ir šoka penkis kartus, kaskart keisdamas kryptį iš dešinės (D) į kairę (K) arba iš kairės į dešinę: pirmiausia šoka į dešinę, tada į kairę, vėl į dešinę, tada į kairę ir galų gale į dešinę.



Bebras gali pasirinkti, į kelintą langelį šokti: 1, 2, 3, 4 ar 5 – kaskart vis skirtingą skaičių. Pavyzdžiui, gali šokinėti taip: dešinėn 2, kairėn 1, dešinėn 5, kairėn 4 ir dešinėn 3 ir atsidurti $2 - 1 + 5 - 4 + 3 = 5$ langelyje dešiniau nuo langelio, kuriame pradėjo.

Kas antras langelis nuspalvintas pilkai – lengviau skaičiuoti.

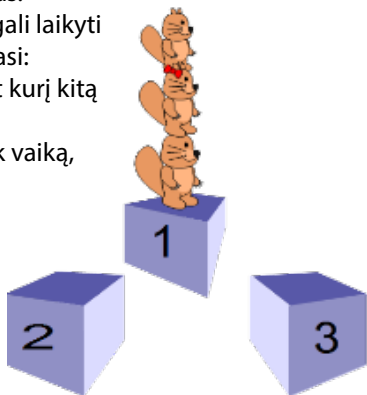
Į kurį iš pažymėtų langelių A, B, C ir D bebras niekada nepateks, šokinėdamas pagal aprašytą taisyklę?

Ėjimų sekos nustatymas – algoritmas, kuriame pirmojo žingsnio pasirinkimas lemia kitus žingsnius. Ėjimai atliekami tam tikra tvarka – būtina algoritmo sąlyga. Uždaviniui būdingas ir veiksmų abstrahavimas.



Akrobatų bebrų šeima – tėvas, mama ir vaikas – rengia akrobatinių šuolių kaskadą. Yra trys pakylės. Iš pradžių šeima stovi ant pirmos pakylės, vienas kitam ant pečių. Šokti nuo vienos pakylės ant kitos galima pagal šias dvi taisykles:

- Šoka tik viršutinis bebras.
- Bebras ant savo pečių gali laikyti tik mažesnę bebrą. Vadinasi:
 - tėvas gali laikyti bet kurį kitą (mamą ar vaiką),
 - mama gali laikyti tik vaiką, bet ne tėvą,
 - vaikas nieko negali laikyti.

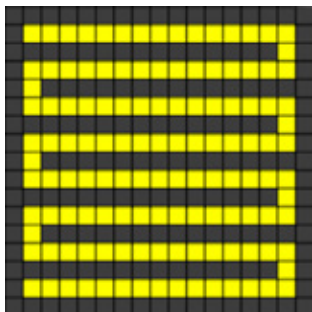


Kiek mažiausiai šuolių reikia atlikti, kad visa šeima sustotų kaskada ant 3-iosios pakylės?



Bebro pilies požemiuose gyvena pabaisa. Geltoni langeliai žymi pabaisos slapstymosi vietas. Pilki langeliai – tai sienos, per kurias pabaisa negali pereiti.

Pabaisa bus pagauta, kai liks tik vienas geltonas langelis – tada ji nebegalės niekur pasprukti.



Kiek mažiausiai spustelėjimų reikia, kad pagautume pabaisą?

Dvejetainės paieškos algoritmas yra vienas iš labiausiai žinomų ir dažnai taikomų algoritmų. Šiuo algoritmu daugelis žmonių ieškodavo žodžio žodyne arba pavardės telefonų knygoje, kai žodynai ir abėcėliniai žinynai buvo tik popieriniai.



Bebrė Alisa turi dešimties nugrauztų medžių krūvą, o bebrė Beatričė – tik vieną medį.

Bebrės pradeda graužti medžius miške vienu metu.

Kiekvieną nuverstą medį jos prideda prie savo medžių krūvos.

Alisa nugražia medį per valandą.

Beatričė graužia medžius vis sparčiau. Pirmąją valandą ji nugražia tik vieną medį, antrąją – du, trečiąją – jau tris medžius ir t. t.



Po kelių valandų Beatričė aplenks Alisą?

Šis uždavinys atkreipia dėmesį į asimptotinę algoritmų analizę. Alisa verčia medžius tiesiniu dažniu, vaizduojamu $O(n)$. Beatričė verčia medžius proporcingai laikui, pakeltam kvadratu. Tai vaizduojama $O(n^2)$. Vadinasi, po n valandų Beatričė bus nuvertusi medžių skaičių, proporcingą skaičiui n^2 .



KROSAS



Trys puikūs bėgikai bebrai susitiko krose. Bėgimo trasa parodyta paveikslėlyje: iš pradžių bėgama į kalną, tada tarp uolų, toliau bėgama nuo kalno ir vėl tarp uolų.



Violeta pradeda pirma, paskui bėga Rudis ir tada Žalė.

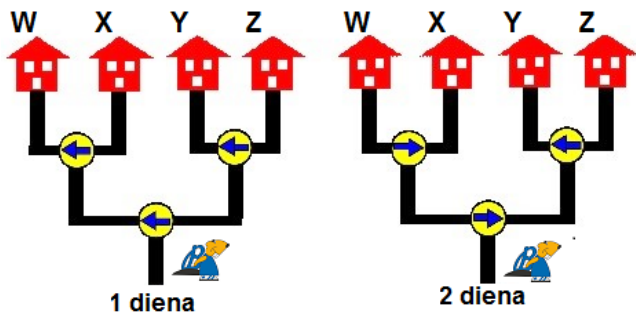
Bėgdamas į kalną, bebras Rudis pralenks tiksliai vieną bebrą.	R	
Bėgdama nuo kalno, bebrė Violeta pralenks tiksliai vieną bebrą.	V	
Bėgdama tarp uolų, bebrė Žalė pralenks tiksliai vieną bebrą.	Ž	

Kuria eilės tvarka bebrai pasieks finišą?

Tai programos testavimo uždavinys. Programuotojas turi įsitikinti, ar tinkamai veikia jo parašyta programa. Programa derinama (angl. *debugging*), t. y. stebimas jos veikimas, pa-teikiant įvairius duomenis.



Kiekvieną dieną bebras lankosi viename iš keturių kaimų pagal schemą. Iš pradžių visos rodyklės nukreipia į kairiąją gatvę. Kirtęs sankryžą, bebras rodyklę pasuka atvirkščia kryptimi.



Taigi pirmą dieną bebras pasuka kairėn pirmoje sankryžoje ir dar kartą kairėn antroje sankryžoje, taip pasiekia kaimą W. Antrą dieną jis pasuka dešinėn pirmoje sankryžoje ir kairėn antroje, taip atvyksta į kaimą Y.

Kuriame kaime bebras lankysis 30-ąją dieną?

Tai – uždavinio sprendimo (programavimo) metodas, kai uždavinys skaidomas į dalis, sudaroma viso uždavinio programa, aprašanti dalių sąveiką, po to analogiški veiksmai atliekami su dalimis, su dalių dalimis ir t. t., kol prieinama prie programavimo kalbos konstrukcijų arba jau turimų komponentų. Metodas vadinamas smulkinamuoju projektavimu ar programavimu (angl. *top-down*).



KĄ VEIKĖ BEBRAS?



Pagal oro sąlygas bebras sprendžia, ką veikti. Jis elgiasi pagal šias taisykles:

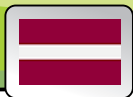
- Jei šiandien saulėta, bet vakar lijo, maudosi upėje.
- Jei šiandien saulėta, vakar taip pat buvo saulėta, žaidžia su smėliu upės pakrantėje.
- Jei šiandien lyja, bet vakar buvo saulėta, žaidžia su kaladėlėmis namie.
- Jei šiandien lyja ir vakar lijo, bebras nežaidžia.

Orai lapkričio 1–8 d. buvo tokie:

Data	Lap. 1	Lap. 2	Lap. 3	Lap. 4	Lap. 5	Lap. 6	Lap. 7	Lap. 8
Oro sąlygos								

Ką bebras veikė lapkričio 7-ąją?

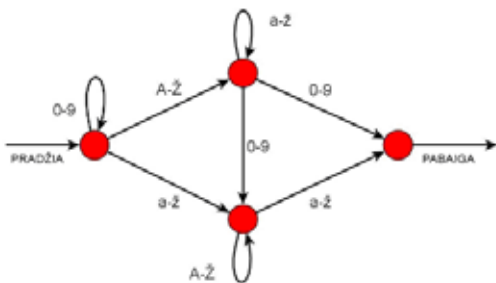
Šį uždavinį galime spręsti dvejopai: pasitelkdami logiką (atlikdami logines operacijas, pavyzdžiui, IR, ARBA) arba taikydami baigtinio automato modelį. Baigtinis automatas – tai matematinė abstrakcija, kuria modeliuojama sistemos būsenų kaita, atsižvelgiant į ankstesnę būseną ir įėjimo signalus, kai būsenų ir galimų įėjimo signalų skaičius yra baigtinis.



Mokyklos kompiuterių klasėje kuriami nauji slaptažodžiai. Leidžiama vartoti:

- Didžiąsias raides nuo A iki Ž.
- Mažąsias raides nuo a iki ž.
- Skaitmenis nuo 0 iki 9.

Kiekvienas slaptažodis sudaromas pagal taisykles, kurios nusakomos schema. Kilpa reiškia kartojimą (nuo nulio iki kiek norima kartų). Linija rodo, kad raidė arba skaitmuo turi būti vartojami tik vieną kartą. Slaptažodis sudaromas, sekant schemos linijomis nuo pradžios iki pabaigos (nebūtina pasiekti kiekvieno taško).



Kuris iš šių slaptažodžių neatitinka schemos?

- A. 123aNNa B. Petras3ABCd
C. 2010Bebras4IEVa D. beNNOZzz

Slaptažodžiui sudaryti taikomas baigtinio automato modelis. Tai modelis, kurio išėjimo simbolis priklauso nuo įėjimo simbolio ir nuo būsenos simbolio, o tolesnė būsena – nuo esamos būsenos ir įėjimo simbolio.



NE TA SKRYBĖLĖ



Keturi bebrai turi po skrybėlę, bet ne tos spalvos!

- Šiuo metu visi keturi bebrai turi netinkamos spalvos skrybėles.
- Laikoma, kad bebras užsidėjo tinkamos spalvos skrybėlę, jei jo marškinėlių spalva nesutampa su skrybėlės spalva.



Bebrė Ana



Bebras Brendonas



Bebras Deividas



Bebrė Emilija

Kuriam bebrui priklauso žalia skrybėlė?

Tradiciškai logika buvo laikoma filosofijos dalimi, bet jau du šimtmečius logika studijuojama ir kaip matematikos, o paskutiniiais dešimtmečiais – kaip informatikos mokslo dalis. Logika tiria ir klasifikuoja argumentų struktūrą, apibrėžia aprašymo schemą, nagrinėja teisingus ir klaidingus teiginius.



KAS YRA KAS?



Lukas vilki marškinėlius su mašinyte. Tomas ir Lukas yra identiški dvyniai. Gabrielės plaukai šviesūs. Eglė ant galvos ryši skarelę. Jonui patinka marškinėliai su spalvotais raštais.

Kokie šių vaikų vardai?



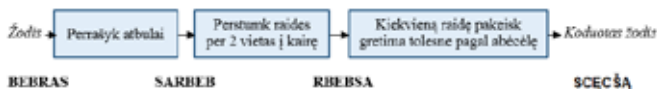
Logika yra ir matematikos, ir informatikos mokslo dalis. Logika tiria argumentų struktūrą, teisingus ir klaidingus teiginius, juos gretina ir priima išvadas. Analizuodami pateikiamus teiginius, nustatome ryšius tarp objektų ir šitaip sprendžiame problemą, t. y. nustatome vaikų vardus.



KURIS ŽODIS?



Aleksas ir Beata siunčia vienas kitam koduotas žinutes.
Kiekvieną žodį jie koduoja atskirai trimis žingsniais.



Pavyzdžiui, užkoduotas žodis BEBRAS yra SCEČŠĄ.
Beata gavo iš Alekso tokią žinutę: LAŤŠP.

Ką norėjo pasakyti Aleksas?

Paveikslėlis, tiksliau, schema rodo, kaip užšifruojamas žodis. Schemos yra vienas iš algoritmų užrašymo būdų. Šiuo algoritmu pradinis žodis pakeičiamas, kad būtų nelengva jį suprasti. Informatikoje šis procesas vadinamas šifravimu.



Įsivaizduokime, kad dabar XVIII amžius. Jūreivis Popajus Karibų jūros saloje randa lobių skrynią ir nori nusiųsti pranešimą savo draugams į žemyną. Popajus geba sukelti jūros bangas. Jo draugai žino, ką reiškia šios bangos.

Radau lobį



Laukiu saloje



Greičiau!

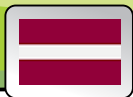


Popajus nusiųcia pranešimą, keldamas tokią bangų seką:



Ką reiškia šis pranešimas?

Mokslas apie informaciją laikomas informatikos dalimi. Norint perduoti informaciją, reikia turėti kodų sistemą – sutartus ženklus (pavyzdyje vartojami trys ženklai – trijų tipų bangos). Ženklams perduoti reikalinga terpė (informacijos kanalas) – mūsų atveju tai yra jūra. Taip pat būtinas informacijos siuntėjas ir gavėjas. Popajus (siuntėjas) saloje rankomis sukelia norimo aukščio bangas, o draugai (gavėjai) žemyne jas stebi ir įvertina bangų aukščius.



Du bebrai susirašinėja žinutėmis. Kad būtų patikimiau, žinutes koduoja savo sugalvotu būdu: paima vieną raidę, lentelėje suranda ją atitinkantį skaičių ir padaugina iš dviejų.

A 1	H 8	O 25	V 37
B 2	I 9	P 26	W 38
C 3	J 15	U 27	X 39
D 4	K 16	R 28	Y 45
E 5	L 17	S 29	Z 46
F 6	M 18	T 35	! 47
G 7	N 19	U 36	? 48

Taip žinutės tekstas pakeičiamas skaičių seka. Pavyzdžiui, BEBRAS koduojamas taip: 410456258.

Kaip jie turėtų užkoduoti žodį INFORMATIKA?

Kodavimas – veiksmas, kuriuo ženklui suteikiamas kodas (sutartinis ženklas). Koduojama dėl įvairių priežasčių: patogesnio informacijos vartojimo, techninių galimybių, informacijos apsaugos (kad neperskaitytų pašaliniai). Kai kalbama apie duomenų apsaugą, tiksliau vartoti šifravimo sąvokas (šifras, šifravimas ir pan.).





TELEFONO KLAVIATŪRA



Tadas renka trumpąsias teksto žinutes mobiliuoju telefonu, kurio klaviatūra parodyta paveikslėlyje. Raidės rašomos, spaudžiant tą patį klavišą vieną ar kelis kartus ir šiek tiek luktelint.



Pavyzdžiui,  reikia paspausti tris kartus, norint parašyti raidę C. Norint parašyti žodį ATE, reikia keturiskart paspausti klavišus.

Pirmiausia:  – vieną kartą.

Tada:  – vieną kartą.

Galų gale:  – du kartus.

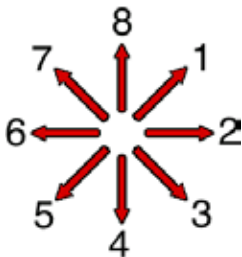
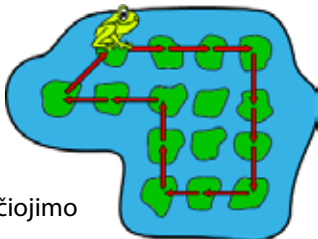
Tadas parašė draugės vardą, paspausdęs 6 kartus.
Koks draugės vardas?

Mažomis klaviatūromis, kuriose yra tik devyni klavišai, raidės rašomos, spaudžiant tam tikrą klavišą kelis kartus ir luktelint. Šitokiu specialiu būdu užkoduojamos raidės. Kurdami ribotų galimybių įrenginių (kaip ši klaviatūra su 9 klavišais) vartotojų sąsajas, informatikai turi sugalvoti įvairių kodavimo būdų. Kurti patogias, patikimas, lengvai suprantamas vartotojo sąsajas – nelengvas uždavinys informatikams.



Varlė šokčioja kūdroje nuo vieno lelijos lapo ant kito, kaip parodyta paveikslėlyje. Varlė baigia šokčioti, kai atsiduria ant to paties lapo, nuo kurio pradėjo.

Aštuonios galimos varlės šokčiojimo kryptys pažymėtos skaičiais:



Užrašykite varlės įveiktą kelių skaičių seką.

Duomenys aprašomi įvairiais kodais. Vadinamasis Freemano (Frimano) sekų kodas skirtas monochrominiams paveikslams koduoti (ir suspausti), aprašant aštuonių krypčių skaičių sekomis. Paveikslas suskaidomas į vienos spalvos gabalėlius, kiekvienas gabalėlis apeinamas (kaip kad varlė šokčioja šiame uždavinyje), naudojant krypčių skaičių sekas. Gautoji skaičių seką yra šio gabalėlio kodas.



Kartą gyveno robotas. Kaip ir kiti robotai, jis mokėjo atlikti paprastas komandas – judėti į priekį ir pasisukti kairėn arba dešinėn. Tačiau jo galvelėje komandų pavadinimai buvo keistoki – raidės susimaišiusios.

Vietoj komandos „priekį“ jis suprasdavo komandą RPEIKN, vietoj „kairėn“ – AKRINĖ.

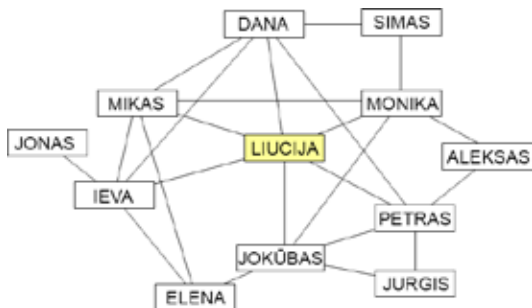
Kokią komandą turėtume pateikti išsiblaškėliui robotui, kad jis pasisuktų dešinėn?

Kodavimas – svarbi informatikos sritis. Koduojant ženklui ar kitokiam objektui suteikiamas kodas. Dažniausiai koduojama, norint pateikti duomenis kompiuteriui, taip pat norint juos suglaudinti. Kai kalbama apie duomenų apsaugą (kad jų neperskaitytų pašaliniai), vartojama šifravimo sąvoka. Yra atskiras mokslas kriptografija, tirianti užšifravimo ir iššifravimo būdus bei priemones.



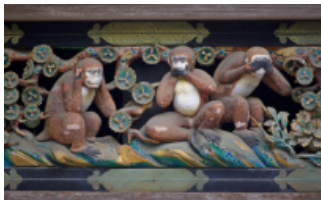
Liucija ir jos draugai užsiregistravo socialiniame tinkle. Štai Liucijos draugų ir tų draugų draugų tinklas. Atkarpa, jungianti du vardus, reiškia draugystę tarp šių žmonių.

- Jei kas nors dalijasi nuotrauka su savo draugais, tada visi tie draugai gali ją komentuoti.
- Jei kas nors komentuoja nuotrauką, tada visi jo ar jos draugai gali matyti komentarą ir nuotrauką, bet negali komentuoti, nebent jie iš pradžių galėjo tai daryti. Liucija įkėlė nuotrauką.



Su kuo Liucija gali pasidalyti šia nuotrauka, jei nenori, kad matytų Jokūbas?

Daugelis socialinių tinklų remiasi tokiais draugų ryšiais, tik tinklai sudėtingesni, ryšių labai daug, yra įvairių ribojimų. Tinkluose svarbu saugumas ir privatumas. Ne visada akivaizdu, kad tavo skelbimą draugo tinklalapyje gali matyti visiškai nepažįstami žmonės. Šio uždavinio socialinis tinklas gali būti interpretuojamas kaip mažytis interneto modelis, kuriame draugai vaizduoja svetaines, o draugų ryšiai išreiškia funkciją „susieta su“. Uždaviniui spęsti geriausia taikyti užliejimo (angl. *flood fill*) algoritmą.



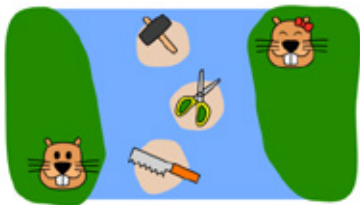
Tu stovi šalia žmogaus, kuris renka slaptažodį kompiuteriu. Kaip pasielsi šioje situacijoje?

- A. Nusigręši.
- B. Telefonu fotografuosi įvedamą slaptažodį.
- C. Pasakysi savo slaptažodį, norėdamas parodyti, kad esi draugiškas.
- D. Žiūrėsi ir stebėsi, kad slaptažodis renkamas nesislepiant.

Slaptažodžiai niekada negali būti visiškai saugūs. Slaptažodžio „nulaužimas“ priklauso nuo jo ilgio ir sudėtingumo (kiek ir kokių simbolių sudaro slaptažodį, ar yra didžiųjų ir mažųjų raidžių, skaičių, kitų leidžiamų simbolių). Kartais įsilaužti padeda tiesiog konteksto išmanymas. Pavyzdžiui, daug vartotojų slaptažodžiu pasirenka augintinių vardus arba gimimo datą. Kaip įvedate slaptažodį, gali stebėti kiti žmonės ar vaizdo kameros. Todėl, pavyzdžiui, bankomatų slaptažodžio įvedimo mygtukai yra pridengti stogeliu. Populiarija biometriniai atpažinties metodai: piršto atspaudas arba veido atpažinimas. Deja, biometriniai duomenys turi vieną trūkumą: jei abejojama, ar slaptažodis pakankamai saugus, galima lengvai jį pakeisti, tačiau tai padaryti nelengva, taikant biometrinius duomenis.



Bebrai Alius ir Bitė kuria žaislus, naudodamiesi vienu iš šių įrankių: plaktuku, žirkklėmis ar pjūkle. Įrankiai laikomi upės salelėse. Kuriam prireikia įrankio, tas jį ir pasiima. Jei reikiamo įrankio nėra salelėje, bebras laukia, kol tą įrankį grąžins kitas bebras ir turimo įrankio nepadaeda. Kartais abiem bebrams vienu metu reikia to paties įrankio. Jei taip nutinka, jie nutraukia darbą ir eina maudytis.



Kurioje iš pateiktų situacijų bebrai tikrai eis maudytis?

- A. Alius turi ir jam reikia . Bitė turi ir jai reikia .
- B. Alius turi ir jam reikia . Bitė turi ir jai reikia .
- C. Alius turi ir jam reikia . Bitė turi ir jai reikia .
- D. Alius turi ir jam reikia . Bitė turi ir jai reikia .

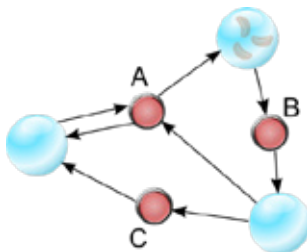
Įrankio paėmimas – tai proceso blokavimas. Situacija, kai kiekvienas sąveikaujančios procesų aibės procesas laukia įvykio, kurį gali pateikti tik tos aibės procesas, vadinama aklaviete. Kai programoms reikia bendrų išteklių, procesų blokavimas yra svarbus veiksnys, siekiant išvengti aklavietės. Bendras išteklių naudojimas ir aklavietės vengimas yra svarbios informatikos sąvokos.



Bebras Bronius apžiūri įdomų įrenginį. Jį sudaro stikliniai rutuliai, kuriuose gali būti pupelių. Rutuliai sujungti, be to, valdomi mygtukais A, B ir C. Paspaudus bet kurį mygtuką, paeiliui atliekami šie du veiksmai:

* įrenginys tikrina, ar visuose rutuliuose, iš kurių rodyklės rodo į paspaustą mygtuką, yra nors po vieną pupelę;

* jei taip, tada iš šių rutulių išimama po vieną pupelę, o į kitus rutulius, į kuriuos rodo rodyklės iš paspausto mygtuko, pridedama po vieną pupelę. Pavyzdžiui, paspaudus mygtuką B, viena pupelė išimama iš viršutinio rutulio, o apatiniame rutulyje viena pupelė atsiranda.



Kokia trumpiausia paspaudimų seka sukelia tokią įrenginio būseną, kuri nebekinta, spaudžiant bet kuriuos mygtukus?

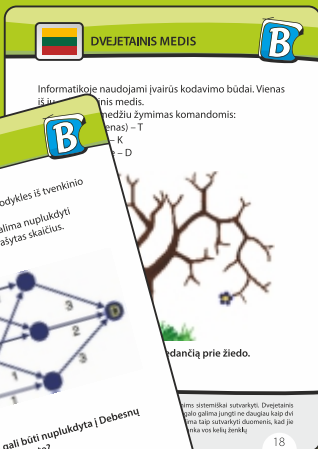
Uždavinys siejamas su tinklais, šiuo atveju – su Petri tinklu. Petri tinklas – formali sistema lygiagrečioms, sąveikaujančioms sistemoms aprašyti, sistemų elgsenai imituoti ar jų atliekamiems veiksams vaizduoti. Petri tinklai padeda geriau suprasti ir modeliuoti kompleksines, lygiagrečiai sąveikaujančias sistemas.

Žaidimo kortelės BEBRAS Informatinio mąstymo ugdymas

nuo
8
metų



Šiame rinkinyje rasite įdomių informatikos uždavinių, kuriuos sukūrė tarptautinio konkurso „Bebras“ bendruomenė.



„Bebro“ konkursas – unikalus projektas, sugalvotas ir pradėtas įgyvendinti Lietuvoje 2004 metais. Jo tikslas – atskleisti mokiniams (ir mokytojams) informatikos mokslo grožį, supažindinti su pagrindinėmis informatikos ir informatinio mąstymo sąvokomis, ugdyti mokinių kūrybiškumą, informacinę kultūrą, algoritminę ir kompiuterinę mąstyseną. Įdomūs, aiškiai formuluojami uždaviniai apima visas fundamentalias informatikos sritis, parodo jų svarbą ir dermę su kitais mokslais.

Creative Commons Attribution — ShareAlike 3.0 Unported License (CC BY-SA 3.0).

Šalia uždavinio pavadinimo pateikiama šalies, pasiūliusios uždavinio idėją, vėliava.

Parengė: V. Dagienė, G. Stupurienė, L. Vinikienė, A. Ivanauskienė (redaktorė) ir V. Kinčius (dizainas)