



UGDYMO
PLĖTOTĖS
CENTRAS



www.bebras.lt

Informatikos
ir informatinio
mąstymo konkursas

·b·e·b·r·a·sTM

© Bebras LT



Informatika:
Bebras 2018

Informatika: Bebras 2018

XV informatikos ir informatinio mąstymo konkurso 2018 metų
pirmojo etapo uždaviniai su sprendimais

Parengė: *Valentina Dagienė, Gabrielė Stupurienė, Lina Vinikienė*

Vilniaus universiteto Duomenų mokslo ir skaitmeninių technologijų institutas

Verčiant ir adaptuojant uždavinius talkino: *Daumilas Ardickas, Linas Būtėnas, Tolmantas Dagys, Viktoras Dagys, Gintautas Grigas, Eglė Jasutė, Tatjana Jevsikova, Alvida Lozdienė, Jūratė Skūpienė*

Redagavo *Renata Valotkienė*

Viršelį sukūrė *Vaidotas Kinčius*

Šią knygėlę galima laisvai parsisiųsti ir ja naudotis.

Siūlome knygėlėje teikiamą medžiagą naudoti klasėje ar užklasinėje veikloje.

Knygelė platinama pagal kūrybinių bendrijų licenciją nekomerciniais tikslais (*Creative Commons Attribution–NonCommercial–ShareAlike*). Tai reiškia, kad knygele galima laisvai dalytis: ją kopijuoti, platinti ar persiųsti. Daugiau informacijos apie šią licenciją galite rasti internete, ieškokite CC BY–NC–SA 3.0

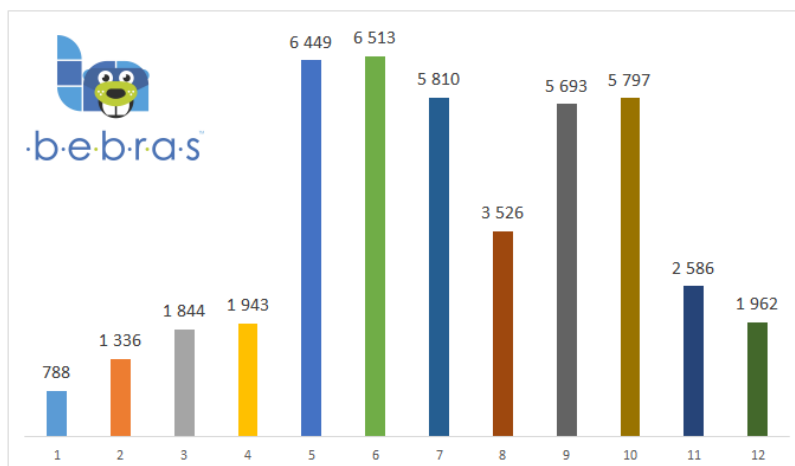


Kūrybinių bendrijų licencija, 2018

IVADAS

2018 lapkričio 5–16 d. įvyko 15-asis mokinių informatikos ir informatinio mąstymo konkursas „Bebras“. Kiekvienais metais Lietuvoje vis daugiau mokinių sprendžia šio konkurso uždavinius, tačiau šiais metais konkurse dalyvavo daugiausia – 44 247 mokiniai:

- 2 124 nykštukai (1–2 klasės),
- 3 787 mažyliai (3–4 klasės),
- 12 962 bičiuliai (5–6 klasės),
- 9 336 draugai (7–8 klasės),
- 11 490 jaunieji (9–10 klasės),
- 4 548 kolegos (11–12 klasės).



Dalyvių skaičius pagal klases

„15 metų lyg ir trumpas laikotarpis, tačiau mums džiugu, kad jau tiek metų esame drauge, kad mokiniai sprendžia uždavinius mokykloje, dalinasi idėjomis su draugais, mokytojais. Džiugu, kad turime „Bebro“ konkurso idėją, kuri mus vienija. Pirmasis konkursas prieš 15 metų įvyko tik Lietuvoje, o dabar jis vyksta daugiau nei 60 šalių. Informatinis mąstymas – tai nėra tik informatika, tai ir mokėjimas analizuoti, spręsti uždavinį, logiškai mąstyti, abstrahuoti. „Bebro“ uždaviniai trumpi, tačiau juose slypi galingos idėjos“ – sako Vilniaus universiteto profesorė Valentina Dagienė.

2018 m. konkurse dalyvavo per 2 614 000 mokinių iš 45 pasaulio šalių.

I etapo uždaviniai:

Nr.	Uždavinio pavadinimas	1-2 klasės	3-4 klasės	5-6 klasės	7-8 klasės	9-10 klasės	11-12 klasės	Informatinis mąstymas
1	Medaus puodynė	6						abstrahavimas, sisteminis vertinimas
2	Pica ir šakutė	6	6					abstrahavimas
3	Žiedai	6	6					duomenų apdorojimas, uždavinio skaidymas
4	Drabužių krūva	6	6					uždavinio skaidymas
5	Gyvutė	9						abstrahavimas
6	Spalvų kvadratas	9	6					algoritnavimas
7	Mieste	9	6	6				algoritnavimas
8	Medžių grupė	9	9	6				abstrahavimas, algoritnavimas
9	Ada ir spalvoti pieštukai	12	9	6				algoritnavimas, uždavinio skaidymas
10	Panašūs patiekalai	12	9	6				algoritnavimas, duomenų apdorojimas
11	Robotai	12						abstrahavimas, loginis pagrindimas
12	Dominuojantis karoliukas	12						algoritnavimas, uždavinio skaidymas
13	Planetos B miestai		9	9	6			algoritnavimas, uždavinio skaidymas
14	Spalvinimas		9	6				uždavinio skaidymas
15	Labirintas su rodyklėmis			9	6			algoritnavimas, sisteminis vertinimas
16	Kas su kuo apsistos?			9				algoritnavimas, duomenų apdorojimas
17	Begalinė ledų porcija			9	6			algoritnavimas, uždavinio skaidymas
18	Spynos		12	6				uždavinio skaidymas
19	Gėlės		12	9	6			loginis pagrindimas
20	Liftas		12	12	9			duomenų apdorojimas, sisteminis vertinimas
21	Mutuojantis ateivis		12			6		algoritnavimas
22	Metro linijos		12	9	6			abstrahavimas
23	Nutekėjimų paieška			12	6			algoritnavimas
24	Kapitono Bebro diena			12		6		sisteminis vertinimas, uždavinio skaidymas

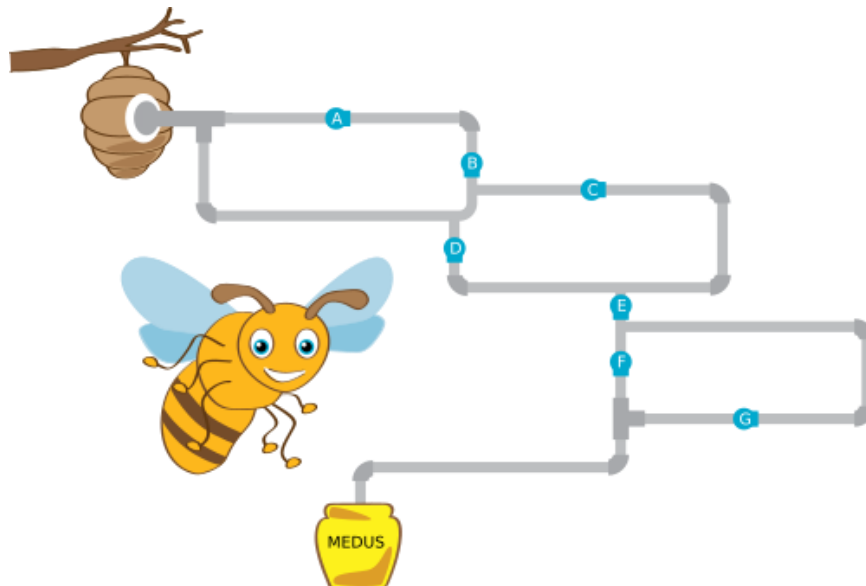
25	Melo trumpos kojos			12	9			loginis pagrindimas
26	Magija			12	9	6		algoritmavimas
27	Bebrų ežeras			12	9	6		algoritmavimas
28	Bebriška modulio operacija				9	6		algoritmavimas
29	Šuoliai lentoje				9	9	6	algoritmavimas, uždavinio skaidymas
30	Dvejtainis viešbutis				12	6		duomenų apdorojimas
31	Jungtys				12	9		loginis pagrindimas
32	Draugų lankymas				12	9	6	algoritmavimas
33	Lemputės ir jungikliai				12	9	6	uždavinio skaidymas
34	Du bebrai dirba				12	9	6	algoritmavimas
35	Kompiuterinis žaidimas				12	9	6	algoritmavimas, loginis pagrindimas
36	Dovanos					12	6	abstrahavimas
37	Kortelių vartymas					12	9	algoritmavimas, duomenų apdorojimas
38	Stulpeliai ir eilutės					12	9	abstrahavimas
39	Knygų rinkinio rikiavimas					12	9	algoritmavimas
40	Soundex algoritmas					12		algoritmavimas
41	Trys draugai					12		algoritmavimas, uždavinio skaidymas
42	Surask sklandytuvą						9	abstrahavimas, algoritmavimas
43	Ilgiausia žodžių grandinė						9	algoritmavimas, duomenų apdorojimas
44	Dingęs automobilis						9	algoritmavimas, duomenų apdorojimas
45	Paslapčių paslaptis						12	loginis pagrindimas
46	Repeticijų planavimas						12	algoritmavimas, uždavinio skaidymas
47	Medicinos laboratorija						12	algoritmavimas
48	Ijunkite šviesas!						12	algoritmavimas, loginis pagrindimas
49	Telegrafo tinklai						12	algoritmavimas, uždavinio skaidymas
50	Salės grindys						12	algoritmavimas, uždavinio skaidymas

Informatinis mąstymas	Kaip ugdytis
Loginis pagrindimas	Sprendžiant uždavinius remtis loginiu samprotavimu, indukcija ir dedukcija, taikyti logines operacijas, dėsnius. Nuosekliai argumentuoti, pagrįsti teiginius, daryti išvadas
Duomenų apdorojimas	Įvaldyti veiksmus su duomenimis: rinkti, atrinkti, rūšiuoti, klasifikuoti, sisteminti duomenis. Atpažinti pagrindinius duomenų tipus ir struktūras. Tvarkyti duomenų rinkinius.
Abstrahavimas	Mokytis išvelgti svarbias detales reiškiniuose ar procesuose, nepaisyti perteklinių, nebūtinų komponentų. Atskleisti uždavinio esmę. Pavaizduoti procesus schemomis ar kitokiais vaizdavimo būdais.
Uždavinio skaidymas	Išvelgti problemos struktūrą, komponentus, skaidomumą. Skaidyti sprendžiamą uždavinį į dalis drauge mąstant apie šių dalių sujungimą (dedukcija). Atskirai spręsti suskaidytas uždavinio dalis.
Algoritmavimas	Spręsti uždavinius, kuriems būdingas taisyklių ir komandų taikymas. Skaityt algoritmų tekstus. Suprasti algoritmo vykdymą. Atpažinti paprastus algoritmus ir taikyti juos įvairioms problemoms spręsti.
Sisteminis vertinimas	Siekti geriausio sprendimo. Subalansuoti naudojamus išteklius. Įsitikinti, ar rezultatas atitinka iškeltą tikslą ir uždavinius. Aptarti sprendimą ir jį tobulinti.
Apibendrinimas	Identifikuoti šablonus, panašumus ir dėsningumus. Naujas problemas spręsti remiantis analogiškais išspręstomis problemomis. Gautus sprendimus apibendrinti (indukcija).

1. MEDAUS PUODYNĖ

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
6						

Bitė Berta nori kuo greičiau pripildyti medaus puodynę.



Kurios sklendės vamzdžiuose turi būti atidarytos, kad medus tekėtų trumpiausiu keliu?

PAAIŠKINIMAS

Atsakymas D, E ir F. Pasirinkus atidaryti kitas sklendžių kombinacijas, medus netekės trumpiausiu keliu.

TAI INFORMATIKA

Algoritmas – baigtinė seka aiškiai suformuluotų nurodymų, kuriuos reikia atlikti uždaviniui išspręsti (tikslui pasiekti). Algoritmas, užrašytas kompiuteriui suprantamu būdu, pavyzdžiui, programavimo kalba, vadinamas programa.

Raktiniai žodžiai: duomenys, algoritmas, kelio paieška.

Informatinis mąstymas: abstrahavimas, sisteminis vertinimas.

2. PICA IR ŠAKUTĖ

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos		
6	6						

Liucija mokosi naudotis šakute. Mama išaiškina picos valgymo taisykles:

- Picos gabaliukai su krašteliu imami ir valgomi rankomis.
- Picos gabaliukai be krašteliu valgomi šakute.

Kiek yra picos gabaliukų, kuriuos Liucija turėtų valgyti šakute?



PAAIŠKINIKMAS

Yra trys picos gabaliukai be krašteliu, kaip parodyta paveiksle.



TAI INFORMATIKA

Liucija turi patikrinti kiekvieną gabaliuką, ar jis turi kraštelį, ar ne. Sakoma, kad kiekvienam gabaliukui atliekamas testas. Kuriant kompiuterines programas, dažnai yra vietos, kuriose programa turi pasirinkti vieną iš kelių sprendimo variantų. Programavime toks pasirinkimas vadinamas šakojimusi ir nusakomas *IF* komanda.

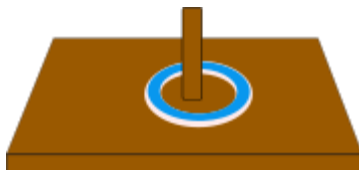
Raktiniai žodžiai: programa, šakojimas, IF komanda.

Informatinis mąstymas: abstrahavimas.

3. ŽIEDAI

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
6	6					

Sara žaidžia žiedų metimą su savo draugais. Kiekvienas žaidėjas per savo ėjimą turi užmesti penkis žiedus ant strypo:



Kiekvienas pataikytas metimas uždriba taškų:

Metimas	Taškai
Pirmas metimas	5
Antras metimas	4
Trečias metimas	3
Ketvirtas metimas	2
Penktas metimas	1

Už nepataikytą metimą taškų negaunama.

Sara numetė penkis žiedus, kaip parodyta paveiksle. Kiek taškų ji surinko?



Atsakymus surašykite į lentelę.

PAAIŠKINIMAS

Teisingas atsakymas: Sara pelnė 6 taškus.

Metimas	Žiedas	T/ N	Taškai
Pirmas metimas	geltonas	N	0
Antras metimas	mėlynas	T	4
Trečias metimas	rožinis	N	0
Ketvirtas metimas	žalias	T	2
Penktas metimas	juodas	N	0
Iš viso			6

TAI INFORMATIKA

Šiame uždavinyje naudojama duomenų struktūra – dėklas arba stekas. Iš kitos pusės, šį uždavinį galime sieti ir su sekomis. Kompiuteris analizuoja duomenis, išdėstytus seka. Duomenys kompiuteriui pateikiami nuosekliai, kad būtų galima juos apdoroti ir gauti rezultatus.

Raktiniai žodžiai: dėklas (stekas), duomenų struktūra, seka.

Informatinis mąstymas: duomenų apdorojimas, uždavinio skaidymas.

4. DRABUŽIŲ KRŪVA

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
6	6					

Mama bebrė kruopščiai išdėsto savo mažojo sūnaus Beno drabužius į krūvą ant stalo.

Marškinėliai	Apatiniai marškiniai	Kelnės	Apatiniai	Petnešos	Kojinės	Batai
						

Benas drabužius rengiasi, imdamas juos nuo krūvos viršaus iš eilės, bet nenori dėvėti petnešų po marškinėliais.

Kurias drabužių krūvas gali naudoti Benas?

A)



B)



C)



D)



PAAIŠKINIMAS

Teisingi atsakymai: B ir D.

Norint rasti teisingą atsakymą, reikia patikrinti krūvas nuo viršutinio elemento (drabužio) ir įvertinti elementų seką, ar ji teisinga, esant ribojimui (marškinėliai turi būti prieš petnešas).

Atsakymai A ir C neteisingi, nes petnešos būtų užsisegamos prieš apsivelkant marškinėlius.

TAI INFORMATIKA

Sakoma, kad norint įeiti į kambarį ir atidaryti žaislų dėžę, pirmiausia reikia atverti kambario duris. Tai gali būti ribojimų pavyzdys, kai iš pradžių reikia atverti kambario duris, o tik po to – atidaryti žaislų dėžę.

Šis uždavinys gali būti sprendžiamas patikrinus, kuri elementų seka tenkina ribojimus, kad elementas A turi būti padėtas prieš elementą B. Tai dažna problema informatikoje, kai atliekamas įvairių sąrašų tikrinimas, ar jie tenkina ribojimus.

Raktiniai žodžiai: ribojimas, elementų seka.

Informatinis mąstymas: uždavinio skaidymas.

5. GYVATĖ

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
9						

Gyvatė ruošiasi šokių konkursui.



Koks bus tolesnis jos šokio judesys?



PAAIŠKINIMAS

Teisingas atsakymas yra C.

A, B ir D atsakymai neteisingi, nes jie neatitinka taisyklių, kurios uždavinyje pavaizduotos paveikslukais. Galime jas suformuoti žodžiais štai taip:

- Kiekvienu šokio judesiu gyvatės uodega keičia poziciją, ji gali būti tiesi arba sulenкта.
- Maža juoda juostelė juda kiekviename judesyje ir atsiranda prieš ar už plačios juodos juostos (žvelgiant nuo galvos pusės).
- Kas antru šokio judesiu gyvatės galva pasisuka 90 laipsnių pagal laikrodžio rodyklę.
- Kitu šokio judesiu gyvatės galva turi pasisukti nauja kryptimi (žemyn) ir uodega turi būti tiesi, o juoda maža juostelė atsiranda prieš plačią juodą juostą.

Vienintelis atsakymas atitinka taisyklės.

TAI INFORMATIKA

Duomenys gali būti įvairių formų, pavyzdžiui, paveikslėliai, tekstas ar skaičiai. Spręsdami šį uždavinį ieškome paveikslų, kuriuose pavaizduoti šokio judesiai ir sekos.

Sekoje kiekvienas paveikslas turi atributus, kai kurie iš tų atributų bus keičiami tolesniame paveiksle, o kai kurie – tik kas antrame ar kas trečiame. Identifikuodami šiuos atributus ir nustatydami taisykles, galime prognozuoti įvykius, kurti taisykles ir spręsti bendresnes problemas.

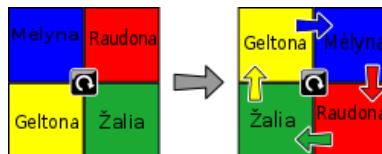
Raktiniai žodžiai: abstrakcija, algoritmas, kartojimo komanda, šablonų atpažinimas.

Informatinis mąstymas: abstrahavimas.

6. SPALVŲ KVADRATAS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
9	6					

Paspaudus mygtuką, spalvos pasisuka, kaip parodyta paveikslėliuose. Paspaudę pirmą kartą gauname:



Kaip atrodys kvadratas, jei paspausime mygtuką dar du kartus? Nutempkite spalvas į jų vietas.

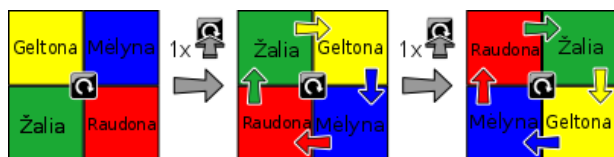


PAAIŠKINIMAS

Atsakymas:



Paaiškinimas:



TAI INFORMATIKA

Šiame uždavinyje yra žingsnių seka, kuriuos atliekant reikia sekti dabartinę būseną – arba kaip atrodo pasukimas, arba kokia viršutinio kairiojo kvadrato spalva (kurią žinant, galima nustatyti kitų kvadratų spalvas).

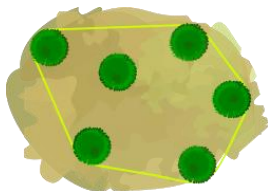
Raktiniai žodžiai: pasukimas, pasukimų seka.

Informatinis mąstymas: algorit mavimas.

8. MEDŽIŲ GRUPĖ

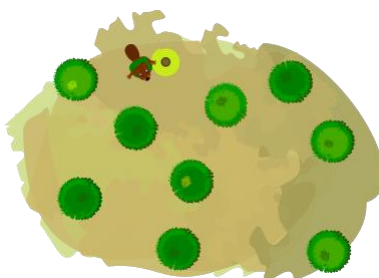
Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
9	9	6				

Bebrė žymi medžius, kuriuos nori panaudoti užtvankai statyti. Ji apjuosia medžius virve – visi medžiai apjuostoje srityje bus jos. Pavyzdžiui, jei medžiai, žiūrint iš viršaus, atrodytų taip, tai bebrė juos galėtų apjuosti:



Apjuostoje teritorijoje iš viso yra 6 medžiai, bet virvė liečia tik 5 medžius.

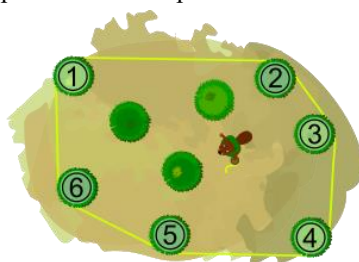
Medžiai iš viršaus atrodo štai taip:



Kelis medžius liestų virvė juos apjuosus? A. 4 B. 5 C. 5 D. 7

PAAIŠKINIMAS

Teisingas atsakymas: 6. Sprendimą galima būtų pasivaizduoti taip:



TAI INFORMATIKA

Ši problema vadinama taškų aibės iškiliojo apvalkalo (angl. *convex hull*) paieška. Vienas sprendimo būdų – mažiausio ploto, kuriame yra visi rinkinyje esantys taškai, daugiakampis. Žodis „iškilusis“ reiškia išgaubimą ir bet kurie du išgaubtos figūros taškai, nupiešti ant popieriaus, gali būti sujungti tiesia linija, esančia figūros viduje. Žodis „apvalkalas“ čia nusako „kevalą“, „lukštą“ arba „dėžę“, „įpakavimą“, sėklos lukšto ar kevalo atitikmenį. Taškų aibės iškiliojo apvalkalo paieška naudojama įvairiuose skaičiavimuose:

- atpažinimo teorija: ar paveiksluke yra veidas?
- rašytinio teksto apdorojimas: ar ranka parašytas rašmuo yra raidė B?
- geografinės informacinės sistemos: koks yra salpos ar upių sistemos dydis?
- pakavimas: koks mažiausias įpakavimo kiekis, reikalingas tvirtai įvynioti trimatį objektą?

Yra daugybė algoritmų šiai problemai spręsti, o efektyvus šios problemos sprendimas labai praktiškas ir naudingas kompiuterių algoritmų pritaikymas.

Taip pat žiūrėkite:

https://en.wikipedia.org/wiki/Convex_hull_algorithm

https://en.wikipedia.org/wiki/Convex_hull

<https://brilliant.org/wiki/convex-hull>

Raktiniai žodžiai: algoritmas, paieška.

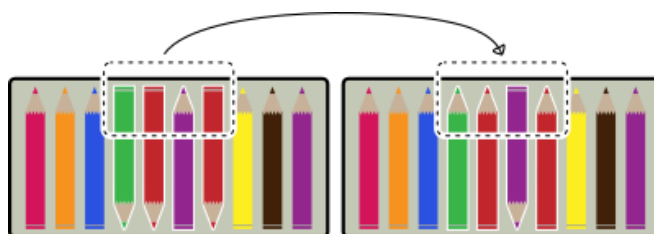
Informatinis mąstymas: abstrahavimas, algoritmavimas.

9. ADA IR SPALVOTI PIEŠTUKAI

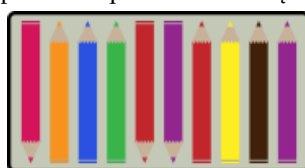
Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
12	9	6				

Ada turi 10 spalvotų pieštukų dėžutėje. Vieni pieštukai nukreipti aukštyn, kiti – žemyn. Ada norėtų, kad visi pieštukai žiūrėtų aukštyn.

Ada žaidžia žaidimą vienu žingsniu apversdama du ar daugiau gretimų pieštukų, t.y. pieštukus, kurie žiūri žemyn, nukreipia aukštyn ir atvirkščiai, kaip parodyta paveiksle.



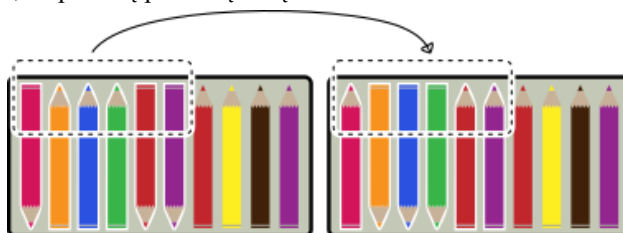
Kiek mažiausiai žingsnių reikės Adai, kad visi šio paveikslo pieštukai žiūrėtų aukštyn?



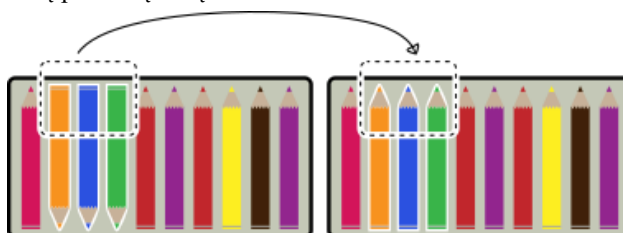
PAAIŠKINIMAS

Vienu žingsniu nepavyks išspręsti šio uždavinio, nes žemyn žiūrintys pieštukai nėra vienas šalia kito. Tačiau dviem žingsniais jau galima pasiekti, kad visi pieštukai žiūrėtų aukštyn.

Pirmas žingsnis: apversti 1, 2, 3, 4, 5, 6 spalvotų pieštukų seką.



Antras žingsnis: apversti 2, 3, 4 spalvotų pieštukų seką.



TAI INFORMATIKA

Uždavinio sprendimas su minimaliu žingsnių skaičiumi – vienas iš svarbiausių informatikos moksle ir gyvenime. Geras programuotojas visada tingus ir nori surasti geriausią uždavinio (problemos) sprendimą. Šio uždavinio atveju, geriausias sprendimas randamas atlikus du žingsnius, bet sudėtingesniais atvejais tai gali būti nelengva. Tik gera programa gali padėti rasti uždavinio sprendimą.

Raktiniai žodžiai: gretimi elementai, seka, programavimas.

Informatinis mąstymas: algorit mavimas, uždavinio skaidymas.

10. PANAŠŪS PATIEKALAI

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
12	9	6				

Virėjas ruošia du patiekalus vakarienei, bet nenori, kad jie būtų panašūs. Du patiekalai laikomi panašiais, jei bent 2 sudėtinės dalys sutampa.

Makaronai	Mišrainė	Salotos	Sriuba	Pyragas
				

Kurie patiekalai yra panašūs?

- A) Sriuba ir makaronai.
- B) Sriuba ir salotos.
- C) Sriuba ir mišrainė.
- D) Mišrainė ir pyragas.

PAAIŠKINIMAS

Teisingas atsakymas: C.

Sriubos recepte yra kiaušinis, svogūnas ir druska. Šios sudėtinės dalys taip pat reikalingos, norint pagaminti mišrainę.

Kitų patiekalų receptuose yra ne daugiau kaip viena bendra sudėtinė dalis.

Sriuba ir salotos bei salotos ir pyragas neturi bendrų sudėtinių dalių. Sriubos ir makaronų receptai turi bendrą sudėtinę dalį – svogūną.

TAI INFORMATIKA

Panašumų ir skirtumų paieška – svarbus programuotojų uždavinys. Pagalvokite apie didžiulį duomenų kiekį biologijoje (pvz., DNR, RNR), chemijoje, astronomijoje, pramonėje ir t.t.

Šiame uždavinyje pateikiami tik 5 receptai, kurių kiekvienas susideda iš 4 sudėtinių dalių. Tačiau astronomas dirba su keletu milijardų orbitų ir žvaigždžių, turinčių daugybę elementų bei savybių. Ir visus šiuos elementus ir savybes reikia palyginti.

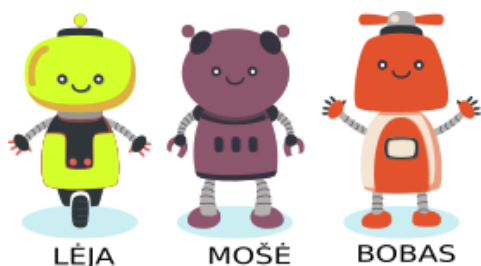
Kompiuteriai gali išspręsti šią problemą naudodami algoritmus. Labai svarbus algoritmų efektyvumas, kad kompiuteris galėtų analizuoti didelius duomenų kiekius per protingą laiką.

Raktiniai žodžiai: šakojimo algoritmas, programos testavimas.

Informatinis mąstymas: algoritmavimas, duomenų apdorojimas.

11. ROBOTAI

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
12						



Šie penki teiginiai turėtų apibūdinti tris pateiktus robotus:

1. Bobas ir Mošė šypsosi.
2. Bobas, Mošė ir Lėja turi po dvi kojas.
3. Mošės galva apvali, ir Lėja turi dvi kojas.
4. Visi trys robotai turi penkis pirštus.
5. Lėja arba Bobas iškėlę rankas.

Kurie iš šių penkių teiginių teisingi?

PAAIŠKINIMAS

Teisingas atsakymas: 1 ir 5.

Sprendimo kelias:

1. Įvertiname kiekvieną teiginį ir nusprendžiame, ar jis teisingas, ar ne. Atkreipiame dėmesį į žodžių IR ir ARBA skirtingą prasmę.
2. Nustatome teisingą atsakymą.

TAI INFORMATIKA

Informatikoje dažnai vartojami loginiai duomenys, kurie gali įgyti tik dvi reikšmes, paprastai žymima teisinga (*true*) ir neteisinga (*false*). Su jomis atliekamos loginės operacijos, dažniausios kurių yra loginė daugyba AND ir OR. Atkreipiame dėmesį, kad sakiny „Bobas ir Mošė šypsosi“ teisingas, nes abu – ir Bobas, ir Mošė – šypsosi. Taip pat pastebime, kad šis teiginys visiškai nemini Lėjos. Vienas pagrindinių informatinio mąstymo įgūdžių – gebėjimas susitelkti ties svarbiausia informacija. Šis procesas vadinamas abstrahavimu – susikoncentravimas į svarbiausias detales ir nereikalingų detalių praleidimas, siekiant lengvesnio problemų sprendimo proceso.

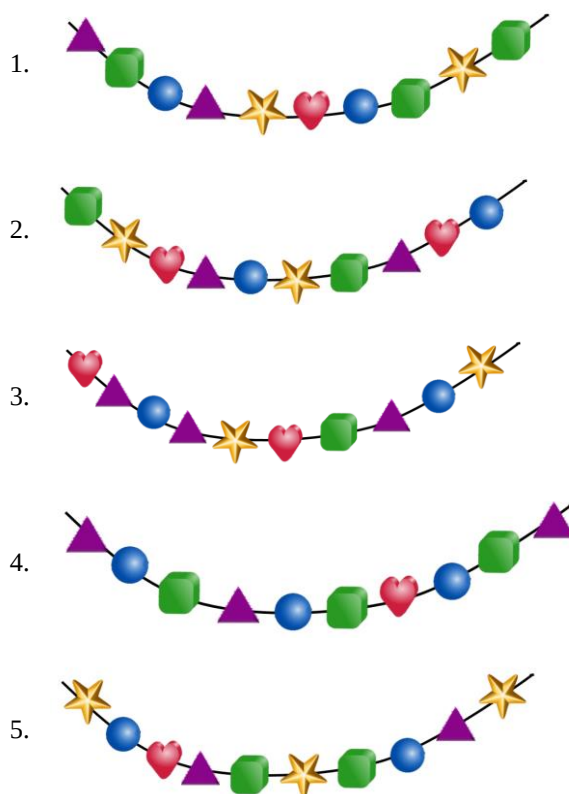
Raktiniai žodžiai: logika, loginiai duomenys, teisinga, klaidinga.

Informatinis mąstymas: abstrahavimas, loginis pagrindimas.

12. DOMINUOJANTIS KAROLIUKAS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
12						

Matome penkis vėrinius. Kai kuriuose vėriniuose vienos rūšies karoliukų daugiau negu kitų rūšių. Tokį karoliuką vadinsime dominuojančiu.



Pažymėkite visus vėrinius, kuriuose yra dominuojantis karoliukas.

PAAIŠKINIMAS

Teisingi atsakymai: 1, 3 ir 5.

1 vėrinys susideda iš 3 kubų, 2 trikampių, 2 žvaigždučių, 1 širdelės. Kubas yra dažniausiai pasikartojantis karoliukas.

2 vėrinys turi po du tos pačios formos karoliukus.

3 vėrinys susideda iš 3 trikampių, 2 širdelių, 2 žvaigždučių, 2 apskritimų, 1 kubo. Trikampis yra dažniausiai pasikartojantis karoliukas.

4 vėrinys susideda iš 3 trikampių, 3 apskritimų, 3 kubų, 1 širdelės. Nėra dažniausiai pasikartojančio karoliuko.

5 vėrinys susideda iš 3 žvaigždučių, 2 trikampių, 2 apskritimų, 1 širdelės. Žvaigždutė yra dažniausiai pasikartojantis karoliukas.

Taigi dominuojantį karoliuką turi tik 1, 3 ir 5 vėriniai.

TAI INFORMATIKA

Sprendžiant šią užduotį, reikia rasti dažniausiai pasikartojantį elementą eilutėje. Todėl galime nusipiešti kiekvieną figūrą ir užsirašyti jų skaičių eilutėse. Didžiausio skaičiaus radimui taikomas algoritmas. Kiekvienai eilutei taikomas toks pats algoritmas ir taip nustatoma dominuojanti figūros forma. Algoritmas – tai baigtinė seka aiškiai suformuluotų nurodymų, kuriuos reikia atlikti, sprendžiant tam tikrą uždavinį (siekiant tikslo).

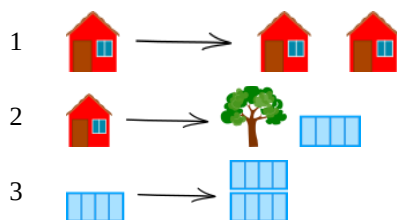
Raktiniai žodžiai: dažnai pasikartojantis elementas, eilutė, dažnio vektorius.

Informatinis mąstymas: algoritmavimas, uždavinio skaidymas.

13. PLANETOS B MIESTAI

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
	9	9	6			

B planetos gyventojai stato miestus tokiu būdu: pradeda nuo vieno namo, po to daro keitimus pagal vieną iš šių trijų taisyklių:



Pavyzdžiui, pritaikę 1-ąją taisyklę, po to 2-ąją taisyklę ir paskui du kartus 3-iąją taisyklę, gautume:



Atkreipkite dėmesį, kad nei vienas objektas nesusikeitė vietomis.

Kurio iš šių miestų nėra B planetoje?

- A.   
- B.   
- C.    
- D.    

PAAIŠKINIMAS

Teisingas atsakymas B. Pagal 2-ąją taisyklę kiekvienas medis turės bloką dešinėje. Tačiau B atveju, dešinysis medis neturi jokio bloko dešinėje pusėje. Ir nėra jokios taisyklės, kuria būtų galima panaikinti blokus.

Miestas A gali būti pastatytas, taikant 1-ą, 2-ą, 3-ią ir dar kartą 3-ią taisykles (būtinai šia nurodyta tvarka).

Miestas C gali būti pastatytas, taikant 1-ą taisyklę tris kartus.

Miestas D gali būti pastatytas, taikant 1-ą taisyklę, tada – 2-ą taisyklę (taikant kiekvienam namui) ir galiausiai kiekvienam blokui taikant 3-ią taisyklę.

TAI INFORMATIKA

Šis uždavinys supažindina su taisyklėmis, kurias naudojant vieni objektai gali būti keičiami kitais. Informatikoje tokios taisyklės naudojamos gramatikose ir aprašo programavimo kalbų sintaksę. Kalbos sintaksė – tai taisyklės, kaip teisingai sudaryti sakinius. Sintaksė skiriasi nuo semantikos. Štai sakiny „Mėnulis miega kramtoma guma“ sintaksiškai teisingas, tačiau visiškai beprasmis.

Informatikoje kalbą sudaro eilučių aibė. Pavyzdžiui, *Python* programavimo kalba yra aibė visų sintaksiškai teisingų *Python* programų. Paprastas *Python* programos pavyzdys:

```
print("Hello")
```

O jei parašytume:

```
print "Hello")
```

tai jau nebūtų *Python* programa. Jei bandytume šią programą atlikti, *Python* transliatorius (speciali programa) praneštų klaidą (sintaksės klaidą).

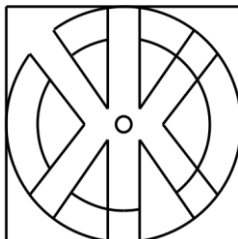
Raktiniai žodžiai: kalbos sintaksė, taisyklė.

Informatinis mąstymas: algoritmavimas, uždavinio skaidymas.

14. SPALVINIMAS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
	9	6				

Pateiktą paveikslą reikia nuspalvinti naudojant kuo mažiau spalvų. Tačiau neleidžiama ta pačia spalva spalvinti jokių dviejų bendrą kraštinę turinčių sričių. Kiek mažiausiai spalvų prireiks?



PAAIŠKINIMAS

Atsakymas: 3.

Galimi įvairūs sprendimai priklausomai nuo to, kokia spalva pradedama spalvinti ir kuri sritis pirmiausia pasirenkama spalvinti. Pateikiamas sprendimas, kai pasirenkama pirmoji spalva ir stengiamasi nuspalvinti kiek galima daugiau sričių, pradedant nuo

viršutinio kairiojo kampo: 

Antrąją spalvą spalvinama, pradedant nuo apatinio kairiojo kampo:



Paėmę trečiąją spalvą, matome, kad galime nuspalvinti visas likusias sritis, ir dvi bendrą kraštinę turinčios sritys nebus tos pačios spalvos:



Tai rodo, kad trijų spalvų pakanka. Toliau reikia įsitikinti, kad negalime nuspalvinti šio paveikslo naudodami mažiau nei tris spalvas. Imkime x pažymėtą geltoną sritį: ji turi bendrą kraštinę su dviem sritimis, taigi turime 3 sritis, ir visos jos turi būti skirtingų spalvų. Vadinasi, neįmanoma šio paveikslo nuspalvinti, naudojant mažiau nei tris spalvas.



TAI INFORMATIKA

Šis uždavinys – tai gerai žinomas matematikos uždavinys, vadinamoji keturių spalvų teorema. Ja teigiama, kad bet koks dvimatis paveikslas, padalintas į sritis, pavyzdžiui, žemėlapis, gali būti nuspalvintas ne daugiau kaip keturiomis spalvomis, laikantis taisyklės, kad jokios dvi bendrą kraštinę turinčios sritys negali būti nuspalvintos ta pačia spalva.

Keturių spalvų teorema turi daug taikymų informatikoje. Pavyzdžiui, planuojant skrydžius ir skirstant pakilimo ir tūpimo takus, kad būtų išvengta susidūrimų, parenkant mobiliųjų tinklų dažnius.

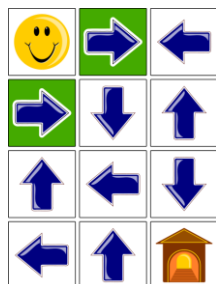
https://en.wikipedia.org/wiki/Four_color_theorem

Raktiniai žodžiai: keturių spalvų teorema, grafas.

Informatinis mąstymas: uždavinio skaidymas.

15. LABIRINTAS SU RODYKLĖMIS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
		9	6			

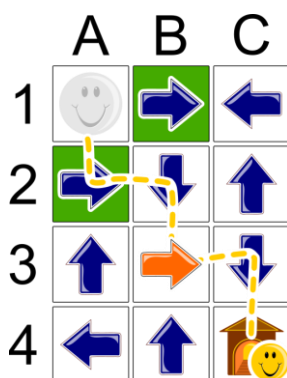


Žaidimo tikslas – padėti jaustukui  grįžti namo. Kai jaustukas yra langelyje su rodykle, jis turi judėti į gretimą langelį rodyklės rodoma kryptimi. Jaustukas gali pradėti kelią eidamas į vieną iš žalių langelių.

Tačiau judant šiame paveiksle nurodytomis kryptimis, neįmanoma pasiekti langelio su namu. Pakeiskite VIENOS rodyklės kryptį taip, kad būtų įmanoma pasiekti langelį su namu.

Norėdami pakeisti rodyklės kryptį, spustelėkite rodyklę keletą kartų, kol atsiras tinkama kryptis.

PAAIŠKINIMAS



Rodyklė, kurios kryptį reikia pakeisti, pažymėta raudonai. Jaustuko kelias paryškintas: A1 (pradžia) – A2 – B2 – B3 – C3 – C4 (tikslas).

Įrodymas, kad šis sprendinys vienintelis, pateikiamas žemiau.

Pradėsime nuo tikslo langelio C4 ir judėsime atgal. Į C3 langelį galima patekti dviem būdais: iš langelio B3 arba iš langelio C2.

C2 langelio rodyklė nerodo į tikslo langelį, todėl tam, kad sprendinys būtų teisingas, šios rodyklės kryptį reikia pakeisti. Kadangi nei viena iš kaimyninių langelių rodyklių į C2 langelį nerodo, reikia keisti dar vienos rodyklės kryptį. O pagal sąlygą galima keisti tik vienos rodyklės kryptį. Taigi tikslo langelis C4 gali būti pasiektas tik iš langelio C3.

Nėra nei vienos rodyklės, kuri rodytų į C3 langelį, todėl turime keisti rodyklės B3 arba C2 langelyje kryptį. Rodyklės, rodančios į C2 langelį taip pat nėra, patekti į C2 langelį per B3 langelį neįmanoma, nekeičiant dar vienos rodyklės krypties.

Langelis B3 pasiekiamas iš pradinio langelio, nekeičiant rodyklės krypties: A2 – B2 – B3. Taigi keliui labirinte rasti reikia pakeisti tik vieną rodyklę B3 langelyje.

TAI INFORMATIKA

Uždaviniui išspręsti reikia suprasti, kaip elgtis labirinte ir ką reiškia rodyklės.

Net jei rasti teisingą kelią paprasta, žymiai sudėtingiau įrodyti, kad nėra kito sprendinio. Šiuo atveju, grįžimų metodas efektyviai patikrina visas galimas judėjimo krypties kombinacijas. Galima įsivaizduoti, kad visi galimi labirinto keliai sudaro medį, kurio tik viena šaka veda tikslo link. Grįžimų metodu tikrinami visi galimi sprendiniai, net ir tie, kurie yra neteisingi.

Raktiniai žodžiai: labirintas, teisingo kelio paieška, grįžimų metodas.

Informatinis mąstymas: algoritmuavimas, sisteminis vertinimas.

16. KAS SU KUO APSISTOS?

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
		9				

Merginų informatikos klubo narės planuoja savaitgalio išvyką. Jos apsistos nakvynės namuose ir gyvens dideliuose šešių lovų kambariuose. Kas su kuo apsistos viename kambaryje?

Kiekviena mergina savo pageidavimus užrašo kortelėje:

1. nurodyma merginas, su kuriomis būtinai nori gyventi (+), ir
2. nurodyma merginas, su kuriomis jokių būdu nenori apsistoti (–).

Klubo prezidentė nori, kad visos liktų patenkintos išvyka. Taigi jos tikslas – taip išskirstyti merginas į kambarius, kad išsipildytų visų norai. Padėkite jai! Paskirkite merginai kambarį nutempdami kortelę su jos pageidavimais į vieną iš stačiakampių.

Alina +: Lilė –:	Ema +: –: Alina	Laura +: –: Ema	Lilė +: –: Laura	Mija +: Ema, Zina –:	Zina +: Mija –: Alina
------------------------	-----------------------	-----------------------	------------------------	----------------------------	-----------------------------

PAAIŠKINIMAS

Norint patenkinti visus pageidavimus reikia koncentruotis į teigiamus pageidavimus. Imk bet kurią nenutemptą kortelę ir:

1. Jei kambarys tuščias,
 - a. nutempk kortelę į tą tuščią kambarį;
 - b. kitu atveju, nutempk kortelę į tą kambarį, kuriame ši mergina nėra nepageidaujamų sąraše (jei tokio kambario nėra – uždavinio sprendinys NEEGZISTUOJA).
2. Nutempk visų merginų, esančių sąraše, korteles į tą patį kambarį. Jei nenutemptų kortelių neliko, uždavinys išspręstas.
3. Grįžk į pirmą žingsnį.

Taikant aprašytą procedūrą, gausime kambarių paskirstymą. Kito galimo sprendinio nėra: Alina turi būti kartu su Lile, bet Alina (taip pat ir Lilė) negali kartu apsistoti nei su Ema, nei su Zina. Taigi Alina negali būti kartu su Mija, kuri privalo apsistoti kartu su Ema ir Zina. Laura turi gyventi atskirame kambaryje, nes ji negali būti kartu nei su Lile (todėl negali ir su Alina), nei su Ema (todėl negali apsistoti nei su Mija, nei su Zina).

Kaip matote, šiam uždaviniui, sprendinys egzistuoja. Tačiau bendru atveju sprendinio gali ir nebūti. Pavyzdžiui, jei Alina norėtų būtinai gyventi kartu su Lile, bet Lilė jokių būdu nenorėtų gyventi su Alina, atsidurtume aklavietėje, ir sprendinio nebūtų.

Alina +: Lilė –:	Lilė +: –: Laura	
Mija +: Ema, Zina –:	Ema +: –: Alina	Zina +: Mija –: Alina
Laura +: –: Ema		

TAI INFORMATIKA

Norėdama patenkinti visų merginų pageidavimus, klubo prezidentė svarsto, kaip suskirstyti merginas į kambarius, atsižvelgus į jų norus. Toks suskirstymas ir yra uždavinio sprendinys.

Kaip rasti sprendinį? Paprasčiausias būdas – perrinkinėti visus galimus variantus (suskirstymus į kambarius) ir turint konkretų suskirstymą, patikrinti, ar jis tenkina duotus ribojimus, kitaip tariant, ar yra sprendinys.

Ribojimų tenkinimo uždaviniai, panašūs į šį, gerai žinomi informatikoje. Sprendžiant tokius uždavinius, reikia paskirstyti resursus, jei norima išvengti konfliktų. Pavyzdžiui, kai reikia sudaryti lėktuvų kilimo ir tūpimo grafiką, t.y. naudojimosi tais pačiais pakilimo ir tūpimo takais, grafiką. Tokius uždavinius galima efektyviai išspręsti, taikant tinkamus algoritmus.

Raktiniai žodžiai: ribojimas, ribojimų tenkinimas, konfliktas.

Informatinis mąstymas: algoritmavimas, duomenų apdorojimas.

17. BEGALINĖ LEDŲ PORCIJA

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos		
		9	6				

Du ledų aparatai gamina ledus, kuriems naudoja tų pačių keturių skonių ledus: , , , ir .

Pirmasis aparatas gamina ledus pagal šią instrukciją:

0. Pradėti tuščiu kūgiu.
1. Atsitiktinai pasirinkti skonį, įdėti du šio skonio ledų rutuliukus.
2. Įdėti vieną bet kurio kito skonio ledų rutuliuką.
3. Jei pasiekiamas prašomas ledų porcijos dydis, sustoti. Priešingu atveju pereiti prie 1 žingsnio.

Antrasis aparatas gamina nesilaikydamas jokios instrukcijos.

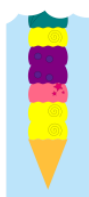
Ledų kūgiuose matote pirmuosius kelis ledų porcijos rutuliukus. Kurią ledų porciją gamina antrasis ledų aparatas?

A.

B.

C.

D.



PAAIŠKINIMAS

Teisingas atsakymas: D. Tai vienintelė ledų porcija, pagaminta nesilaikant instrukcijų. Pradedama teisingai: dedami du atsitiktinai pasirinkto skonio ledų rutuliukai  ir  ir vienas bet kurio kito skonio , bet tada dedami du skirtingų skonių ledų rutuliukai , nors turėtų būti du to paties skonio.

Atsakymų A, B ir C ledai pagaminti pagal instrukcijas, todėl juos *tikrai* gali pagaminti pirmasis aparatas, tačiau neatmetama galimybė, kad antrasis aparatas juos kažkaip pagamino netyčia – bet tikrai negalime teigti, kad jis *neabejotinai* pagamino.

TAI INFORMATIKA

Šablonai ledų kūgiuose, žodžiuose, paveikslėliuose gali būti sukurti, remiantis trumpomis instrukcijomis. Šablono atpažinimas ar jo

pastebėjimas yra kasdienis informatikos mokslininko darbas. Kartais šablonai pasikartoja, pavyzdžiui, .  ..

Lengviau pastebėti šį paprastą šabloną, kuris pasikartoja  .

Šis uždavinys sudėtingesnis, nes šablonas nepasikartoja. Tai spąstai informatikos (ar kitam) mokslininkui: kartais instrukcijos gali būti atsitiktinės. Pavyzdžiui, antrasis ledų gaminimo aparatas renkasi ledų rutuliukus atsitiktinai, nors atrodo, kad laikomasi instrukcijų. Galima atpažinti, kad instrukcijos nėra teisingos. Tačiau vien tik stebėdami niekada negalime būti tikri, jog instrukcijos vykdomos.

Raktiniai žodžiai: instrukcija, taisyklė, šablonas.

Informatinis mąstymas: algoritmavimas, uždavinio skaidymas.

18. SPYNOS

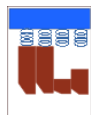
Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
	12	6				

Bebras Bronius taiso specialias spynas. Paveiksle parodyta, kaip veikia tokios spynos ir raktai.

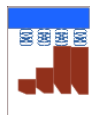


Tačiau spynos ir raktai susimaišė. Kuriai iš pateiktų spynų tinka šis  raktas?

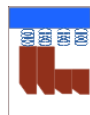
A.



B.



C.

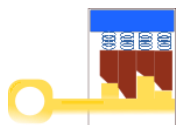


D.



PAAIŠKINIMAS

Teisingas atsakymas: D



TAI INFORMATIKA

Paprastai sudėtingą problemą reikia išskaidyti į smulkesnes dalis ir ieškoti šablonų. Šablonai – panašumų ar dėsningumų visuma, kuri būdinga visoms dalims.

Šablonų (ar vaizdų) atpažinimas – viena iš keturių pagrindinių informatikos sričių. Ši sritis tyrinėja nedidelių, suskaidytų problemų panašumus ir dėsningumus, padedančius efektyviau išspręsti kompleksines problemas.

Šablonų būsenos keičiasi pagal rakto struktūrą. Kai raktas įkišamas į spyną, tada spynos šablonas pasikeičia (veidrodžiškai atspindint). Todėl norint išspręsti uždavinį, reikia įsivaizduoti būsimą šablono būseną.

Raktiniai žodžiai: dekomponavimas, vaizdų (šablonų) atpažinimas.

Informatinis mąstymas: uždavinio skaidymas.

19. GĖLĖS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
	12	9	6			



Klara mėgsta spalvotas gėlių puokštes, tad nuėjo į gėlių parduotuvę jų nusipirkti:

			
kardelis	lėlija	tulpė	rožė

Kiekvienos rūšies gėlės yra trijų spalvų:

Balta	Geltona	Mėlyna
-------	---------	--------

Klara nori šešių gėlių puokštės, atitinkančios tokius reikalavimus:

- kiekvienos spalvos gėlių turi būti ne daugiau kaip po dvi,
- vienos rūšies gėlės negali būti tos pačios spalvos,
- vienos rūšies gėlių turi būti ne daugiau kaip dvi.

Kuri iš puokščių atitinka visus reikalavimus?

A	B	C	D
			

PAAIŠKINIMAS

Teisingas atsakymas: D. Puokštėje A yra 3 baltos gėlės, B – trys rožės, C – du geltoni kardeliai.

TAI INFORMATIKA

Uždaviniai informatikoje dažnai aprašomi ribojimų rinkiniu, ir uždavinio sprendinys turi patenkinti juos visus (arba kiek įmanoma daugiau). Galimi ir sudėtingesni uždaviniai, kur ribojimai sujungiami loginiais operatoriais konjunkcija (visi teiginiai turi būti tenkinami, kaip ir šiame uždavinyje) arba disjunkcija (bent vienas teiginys turi būti tenkinamas).

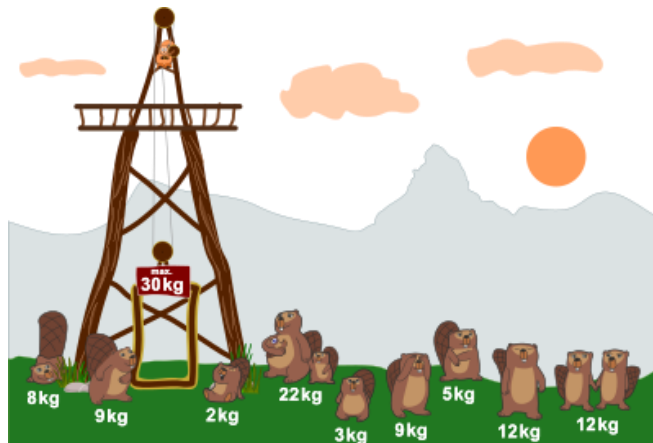
Raktiniai žodžiai: ribojimas, loginis operatorius, disjunkcija, konjunkcija.

Informatinis mąstymas: loginis pagrindimas.

20. LIFTAS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
	12	12	9			

Bebrų grupė lankosi gražioje vietovėje ir nori pakilti liftu į bokšto apžvalgos aikštelę. Jau vėlu, ir liftas pakils į aikštelę tik dukart, be to, kelti vienu metu jis gali ne daugiau nei 30 kg.

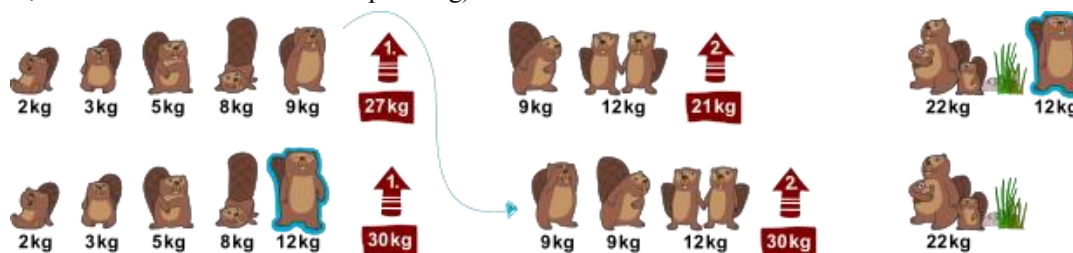


Kaip paskirstyti bebrus į lifto kėlimus taip, kad į aikštelę pakiltų kuo daugiau bebrų? Nutempkite bebrus į liftus apačioje.



PAAIŠKINIMAS

Galima bandyti į vieną liftą sulaipinti kuo daugiau bebrų, pradedant nuo lengviausio: $2 + 3 + 5 + 8 + 9 = 27$ kg, o į antrą liftą $9 + 12 = 21$ kg (8 bebrai), bet įmanoma sulaipinti į lifthus ir 9 bebrus (pakeitus 9 kg bebrą 12 kg bebru pirmame kėlime, į antrą kėlimą tilps likęs 9 kg bebras, ir abiem kėlimais bus keliama po 30 kg):



Yra ir daugiau galimų sprendimo būdų.

TAI INFORMATIKA

Šiam uždaviniui tinka labai daug paskirstymo variantų, o išnagrinėti juos visus užtruktų labai ilgai. Tokiu atveju, turime rasti kuo geresnį sprendinį, kuris ne visada optimalus.

Informatikoje neretai kalbama apie praktiškai neišsprendžiamus uždavinius. Bet šiuo atveju, vadovavomės praktiška strategija: sulaipinom kuo daugiau bebrų į pirmą liftą ir bandėm apkeisti po vieną. Toks sprendimo būdas, kurį pasitelkus randamas geras, bet ne būtinai geriausias sprendinys, vadinamas euristika.

Raktiniai žodžiai: optimizavimas, kuprinės uždavinys, euristinis sprendimo būdas.

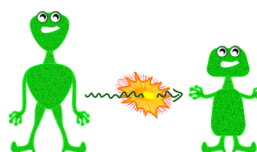
Informatinis mąstymas: duomenų apdorojimas, sisteminis vertinimas.

21. MUTUOJANTIS ATEIVIS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
	12			6		

Ateivis turi galvą, liemenį, dvi rankas ir dvi kojas. Jis gali transformuotis įvykdydamas mutacijos komandas. Ta pati kūno dalis gali mutuoti daugiau nei vieną kartą.

Mutacijų komandos		
G(A): pakeisti galvą į 	G(K): pakeisti galvą į 	G(T): pakeisti galvą į 
L(A): pakeisti liemenį į 	L(K): pakeisti liemenį į 	L(T): pakeisti liemenį į 
R(+): pakeisti rankas į ilgas 	R(-): pakeisti rankas į trumpas 	
K(+): pakeisti kojas į ilgas 	K(-): pakeisti kojas į trumpas 	



Transformacijos pavyzdys mutacijų seka: G(K), L(K), R(-), K(-):

Kaip atrodys ateivis, atlikęs tokią mutacijų seką?

G(T), K(+), L(T), R(+), G(A), R(-), L(A)

A	B	C	D
			

PAAIŠKINIMAS

Teisingas atsakymas: C.

Kiekvienai ateivio kūno daliai vėlesnė mutacija panaikina ankstesnių mutacijų rezultatus. Taigi galutinis rezultatas yra apvali galva, apvalus kūnas, trumpos rankos ir ilgos kojos. Todėl atsakymas yra C.

TAI INFORMATIKA

Vykdam programą, instrukcijų (komandų) seka vykdoma nuosekliai. Galva, liemuo, rankos ir kojos – tarsi kintamieji arba funkcijos, naudojami programoje. Kūno dalių formų parametrai: A – apskritimui, S- kvadratui ir T – trikampiui yra reikšmė, priskirta kintamajam, arba parametras, perduotas funkcijai.

Raktiniai žodžiai: seka.

Informatinis mąstymas: algorit mavimas.

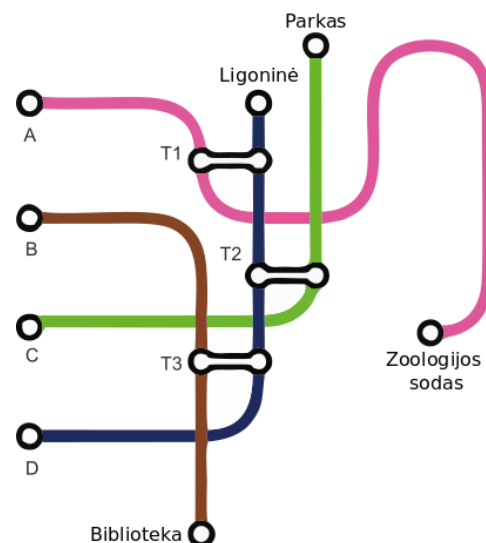
22. METRO LINIJOS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	Taivanas
	12	9	6			

Mieste veikia 4 metro linijos, prasidedančios A, B, C ir D stotyse. Yra 3 jungiamosios stotys T1, T2 ir T3, kuriose galima pereiti iš vienos metro linijos į kitą.

Jonas važiavo iki zoologijos sodo ir tik vieną kartą keitė metro liniją. Kurioje stotyje Jonas pradėjo savo kelionę?

A B C D

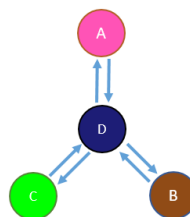


PAAIŠKINIMAS

Teisingas atsakymas: D.

Skirtingų metro linijų susijungimas vaizduojamas lentele ir grafiku:

Iš	Į		
A	D		
B	D		
C	D		
D	A	B	C



Jei Jonas pradėtų A stotyje, nereikėtų nė karto persėsti.

Jei pradėtų B stotyje, reikėtų du kartus keisti metro liniją (į T3 ir tuomet į T1).

Jei pradėtų C stotyje, taip pat reikėtų du kartus keisti metro liniją (į T2 ir tuomet į T1).

Vienas persėdimas į T1 metro liniją reikalingas, jei išvykstama iš D stoties.

TAI INFORMATIKA

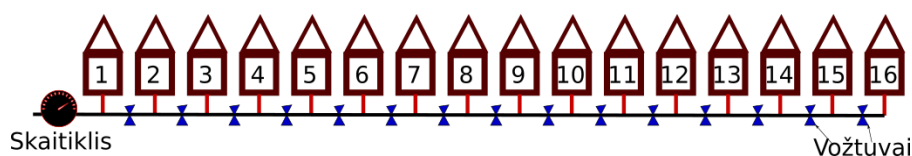
Šiuo uždaviniu supažindinama su grafais. Grafai gali būti naudojami ryšiams tarp objektų pavaizduoti. Grafas yra struktūra, kurią sudaro viršūnės ir briaunos. Uždavinyje viršūnės atitinka stotys, o briaunos – metro linijos. Metro linijų maršruto žemėlapis yra intuityviai suprantama grafiko versija. Esama daug realaus gyvenimo pavyzdžių, kuriose naudojami grafai. Tai prisijungimas prie draugų socialiniame tinkle, žemėlapių naudojimas rasti trumpiausią kelią, pateikiamos rekomendacijos apsipirkimo svetainėse ir kt. Todėl gebėjimas perkelti realias gyvenimo situacijas į grafus – svarbus įgūdis kompiuterių moksle.

Raktiniai žodžiai: duomenų vaizdavimas, grafas.

Informatinis mąstymas: abstrahavimas.

23. NUTEKĖJIMŲ PAIEŠKA

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
		12	6			



Gatvėje, kurioje 16 namų, įrengta vamzdynų magistralė. Pastebėtas vandens nutekėjimas. Pareigūnai bando surasti pažeistą vietą, todėl gyventojai išjungė vandenį visoje gatvėje.

Norėdami surasti nutekėjimą, pareigūnai užsuka vožtuvą tarp dviejų namų ir stebi, ar vamzdyje esantis skaitiklis vis tiek rodo vandens naudojimą. Jei jie pradėtų vožtuvo užsukimą tarp 8 ir 9 namo, ir skaitiklis rodytų, kad vanduo tebenaudojamas, jie žinotų, kad nutekėjimas yra tarp 1 ir 8 namo (priešingu atveju – tarp 9 ir 16 namo).

Tarkime, kad vamzdynų magistralėje yra tik vienas nutekėjimas. Kiek mažiausiai vožtuvų turi užsukti pareigūnai, kad įsitikintų, jog jie surado nuotėkį, nesvarbu, kur jis yra?

PAAIŠKINIMAS

Atsakymas: 4. Tai labai praktinis pavyzdys, kaip dvejetainės paieškos algoritmas naudojamas ne informatikoje.

Kai nutekėjimas gali būti bet kurioje vietoje, geriausia pasikliauti ne sėkme, norint jį surasti, bet kiekvieną kartą padalyti paieškos lauką į dvi lygias dalis. Tai parodys, kurioje paieškos pusėje yra nuotėkis. Kartojant šį procesą, galima surasti nuotėkį. Pavyzdžiui, pirmą kartą tikriname tarp 8 ir 9 namo, antrą kartą tarp 4 ir 5 namo, trečią kartą tarp 2 ir 3 namo ir ketvirtą kartą tarp 1 ir 2 namo.

Toks pats dvejetainis paieškos algoritmas gali būti naudojamas knygos paieškai bibliotekoje ir pan.

TAI INFORMATIKA

Tai yra klasikinis paieškos algoritmas: dvejetainė paieška, kai elemento ieškoma surikiuotame sąrašė jį dalijant pusiau. Pirmiausia palyginamas ieškomas elementas su elementu, esančiu sąrašo viduryje. Jeigu jie sutampa, paieška baigiama. Jeigu nesutampa, sąrašas padalijamas į dvi dalis ir toliau ieškoma toje dalyje, kurioje galėtų būti ieškomasis elementas. O tai nustatoma iš palyginimo rezultatų: ar ieškomasis elementas buvo didesnis, ar mažesnis už vidurinį. Algoritimą kartojant, paieškos sritis kaskart dvigubai sumažėja, kol pagaliau randamas elementas.

Raktiniai žodžiai: dvejetainė paieška.

Informatinis mąstymas: algorit mavimas.

24. KAPITONO BEBRO DIENA

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
		12		6		

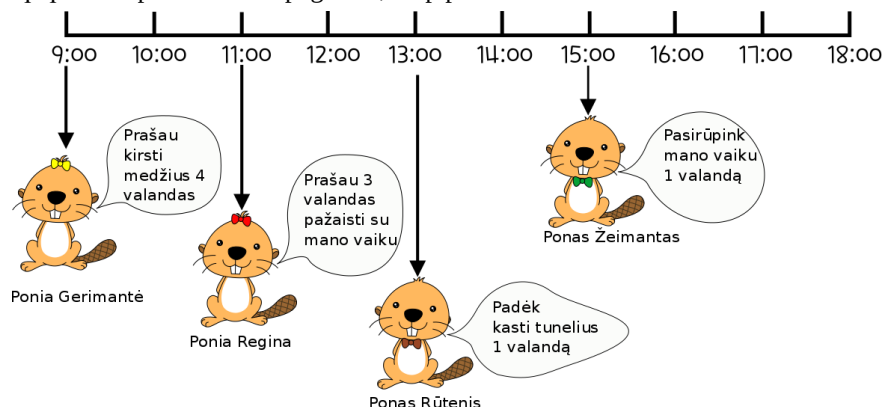
Kapitonas Bebras gerai sprendžia problemas. Kai bebrai prašo pagalbos, kapitonas visada padeda. Deja, kapitonas Bebras vienu metu gali atlikti tik vieną užduotį. Jei prašoma pagalbos tuo metu, kai kapitonas padeda kitam bebrui, jis reaguoja pagal prioriteto lentelę:

Prioritetas	Užduotis
Aukštas	Darbas su bebrūkiais
Vidutinis	Darbas su medžiais
Žemas	Kitos užduotys

Jeigu kapitonas paprašomas atlikti naują užduotį, kuri:

- mažiau svarbi, nei šiuo metu atliekama, jis tęsia pradėtą užduotį ir naująją atlieka vėliau,
- tokia pati svarbi kaip šiuo metu atliekama, jis atlieka abi užduotis pakaitomis po vieną valandą,
- svarbesnė nei šiuo metu atliekama, jis nutraukia dabar atliekamą užduotį, atlieka naują užduotį ir ją baigęs grįžta prie tos, kurią nutraukė.

Vieną dieną keturi bebrai paprašė kapitono Bebro pagalbos, kaip pavaizduota:



Kokiu metu kapitonas atliks ponios Gerimantės užduotį?

A) 13:00

B) 16:00

C) 17:00

D) 18:00

PAAIŠKINIMAS

Teisingas atsakymas: 17:00. Kapitono Bebro darbo diena pavaizduota lentelė:

	9:00 – 10:00	10:00 – 11:00	11:00 – 12:00	12:00 – 13:00	13:00 – 14:00	14:00 – 15:00	15:00 – 16:00	16:00 – 17:00	17:00 – 18:00
Ponia Gerimantė	Yellow	Yellow				Yellow		Yellow	
Ponia Reginą			Red	Red	Red				
Ponas Rūtėnis									Orange
Ponas Žeimantas							Green		

Kapitonas 9:00 valandą pradeda vidutinio prioriteto užduotį, kurią atlikti prašė ponia Gerimantė. Tada 11:00 val. sulaukęs ponios Reginos užduoties, kurios prioritetas aukštas, nutraukia ponios Gerimantės užduotį ir atlieka ponios Reginos. Nors 13:00 val. sulaukia naujos pono Rūtenio užduoties, kapitonas tęsia ponios Reginos užduotį, nes pono Rūtenio prašymas turi žemesnį prioriteta, nei tuo metu atliekama užduotis. Baigęs ponios Reginos užduotį 14:00 val., kapitonas palygina ponios Gerimantės užduotį, kuri buvo nutraukta, su pono Rūtenio. Kadangi ponios Gerimantės užduoties prioritetas aukštesnis nei pono Rūtenio, jis tęsia ponios Gerimantės užduotį. Kai 15:00 val. sulaukia pono Žeimanto užduoties, kapitonas vėl nutraukia ponios Gerimantės užduotį ir pirmą atlieka pono Žeimanto. Kai po valandos ją baigia, kapitonas vėl pratęsia ponios Gerimantės užduotį, ją užbaigia per valandą (ir dieną užbaigia atlikdamas pono Rūtenio užduotį). Taigi laikas, kada bus užbaigta ponios Gerimantės užduotis, yra 17:00 val.

TAI INFORMATIKA

Ši problema apima operacinės sistemos procesų vykdymo eiliškumą. Procesų planavimas apima įvairius būdus, leidžiančius keliems procesams efektyviai pasidalinti vieną ar ribotą kiekį procesorių, kad jie būtų kuo labiau užimti. Šiame uždavinyje kapitonas Bebras atitinka procesorių, užduočių užklausa iš bebrų atitinka procesus, o kapitono taisyklės atitinka proceso planavimo algoritmą. Operacinės sistemos procesų planavimo algoritmas įgalina procesorių (CPU) atlikti kelias užduotis vienu metu (angl. *multitasking*) ir daro įtaką per tam tikrą laiko vienetą atliktų užduočių skaičiui bei vidutiniam visų užduočių laukimo laikui. Šie pavyzdžiai dažnai pasitaiko mūsų gyvenime, pavyzdžiui, laukimas eilėse prekybos centruose. Turėtume sukurti efektyviausias strategijas, kai yra tik viena konferencijų salė, bet ja naudotis nori kelios grupės, arba kai turime ribotą laiką, bet privalome atlikti daug užduočių.

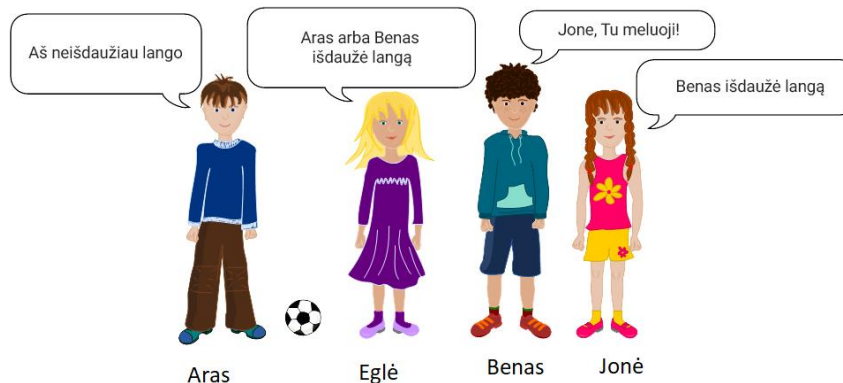
Raktiniai žodžiai: procesų planavimas.

Informatinis mąstymas: sisteminis vertinimas, uždavinio skaidymas.

25. MELO TRUMPOS KOJOS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
		12	9			

Vieną dieną Aras, Benas, Eglė ir Jonė žaidė futbolą netoli mokytojos Alvidos namų. Staiga kažkas išdaužė langą. Mokytoja norėtų žinoti, kas tai padarė. Alvida gerai pažįsta savo mokinius, žino, kad trys iš jų visada sako tiesą, tačiau ji dvejoja dėl ketvirto.



Kas išdaužė langą? A. Aras B. Benas C. Eglė D. Jonė

PAAIŠKINIMAS

Teisingas atsakymas: B. Pirmiausia pastebėjime, kad Jonė ir Beno teiginiai prieštarauja vienas kitam: abu kartu negali būti teisingi arba klaidingi. Todėl aišku, kad vienas iš jų sako tiesą, o kitas meluoja. Galimi du skirtingi, bet teisingi samprotavimų būdai. Reikėtų juos abu žinoti.

----- 1 būdas -----

(a) Jei Jonė sako tiesą, tada Benas meluoja.

(b) Jei Benas sako tiesą, tada Jonė meluoja, bet tokiu atveju turi meluoti dar kažkas kitas: arba Eglė, arba Aras. Tačiau buvo teigta, kad grupėje yra tik vienas melagis. Gauname prieštaravimą.

Vadinasi, Benas išdaužė langą.

----- 2 būdas -----

Samprotaukime šitaip:

(a) Jei Jonė meluoja sakydama, kad Benas išdaužė langą, tada visi likusieji vaikai sako tiesą. (Mes žinome, kad mokytoja Alvida pažįsta savo mokinius ir žino, kad trys iš jų sako tiesą.). Tokiu atveju, Aras sako tiesą tik teigdamas, kad jis lango neišdaužė, bet tada iš Eglės teiginio darome išvadą, kad Benas išdaužė langą. Gauname prieštaravimą Beno teiginiui. Vadinasi, Jonė negali meluoti.

b) Jei Benas meluoja, reiškia, kad visi kiti vaikai sako tiesą. Tuo atveju, Aras lango neišdaužė ir Eglės teiginys reiškia, kad langą išdaužė Benas. Šitaip teigia ir Jonė. Vadinasi, tai gali būti teisinga.

Išnagrinėjus abi (a ir b) galimybes, aišku, kad langą išdaužė Benas.

TAI INFORMATIKA

Logikos algebrą, kuri sudaro informatikos ir kompiuterių programavimo pagrindą, sukūrė Džordžas Būlis 1854 m. (George Boole, 1815–1864). Loginiai duomenys turi tik dvi reikšmes, kurias žymime vardais *true* ir *false* arba skaitmenimis 1 ir 0. Tai reikšmės teiginio, apie kurį galima pasakyti, kad jis yra teisingas arba klaidingas. Jeigu teiginys teisingas, sakoma, kad jo loginė reikšmė yra *true*, jei klaidingas – *false*. Pavyzdžiui, teiginio „Dabar lyja“ reikšmė yra *true*, jei dabar iš tikrųjų lyja, ir *false* – priešingu atveju. Teiginys „skaičius 12 yra lyginis“ visada teisingas, nes skaičius 12 iš tikrųjų lyginis. Taigi šio teiginio reikšmė yra *true*. Pagrindinis logikos algebros elementas yra reiškinys. Reiškinių sudaromi iš loginių kintamųjų, loginių reikšmių ir loginių operacijų (inversijos, konjunkcijos, disjunkcijos ir kt.), taip pat paprastųjų skliaustų. Dažniausiai loginiuose reiškiniuose būna ir santykio operacijos. Loginių reiškinų reikšmės patogiu nusakyti teisingumo lentelė, t. y. lentelė, kurioje nurodomi visi galimi loginių reiškinų deriniai ir nurodomos operacijos arba reiškinio reikšmės. Loginiai reiškiniai vartojami programavime, taip pat daugelyje taikomųjų programų, pavyzdžiui, skaičiuoklėse, saityno paieškos sistemose. Kompiuteriai sudaryti iš elektros grandinių, kurios geriausiai skiria dvi stabilias būsenas, todėl Būlio algebra (operacijos ir taisyklės) puikiai tinka skaitmeniniams kompiuterio darbams nusakyti ir analizuoti.

Raktiniai žodžiai: logika, teiginys.

Informatinis mąstymas: loginis pagrindimas.

26. MAGIJA

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
		12	9	6		

Milda sėdi ant kalno ir žavisi trimis kriokliais. Ji gali įmesti į vieną iš krioklių daiktą, kuris toliau bus plukdomas upe. Bet upėse yra tiltų su troliais, kurie magiškai pakeičia praplaukiančius po tiltais daiktus.

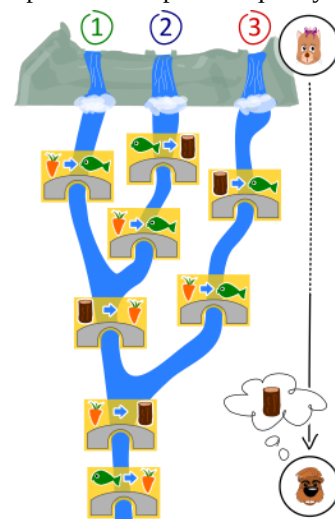
Pavyzdžiui, kai morka plaukia po tiltu, troliai pakeičia ją į žuvį.



Justinas sėdi ir laukia Mildos įmesto daikto upės gale.

Jei Justinas nori gauti rąstą, kokį daiktą ir į kurį krioklį turi įmesti Milda?

- A) Įmesti žuvį  pirmame krioklyje.
- B) Įmesti žuvį  antrame krioklyje.
- C) Įmesti morką  antrame krioklyje.
- D) Įmesti morką  trečiame krioklyje.



PAAIŠKINIMAS

Teisingas atsakymas: B.

Pažiūrėkime, kas atsitiks kiekvienu atveju:

- A) Žuvis, įmesta pirmame krioklyje, bus pakeista tik paskutiniame tilte, todėl Justinas gaus morką.
- B) Žuvis, įmesta antrame krioklyje, bus pakeista į rąstą, bet toliau plaukiant bus pakeista į morką, o po kitu tiltu morka pakeista į rąstą. Todėl Justinas gaus rąstą.
- C) Morka, įmesta antrame krioklyje, bus pakeista į žuvį, o vėliau žuvis vėl pakeista į morką. Todėl Justinas gaus morką.
- D) Morka, įmesta trečiame krioklyje, bus pakeista į žuvį, o vėliau žuvis vėl pakeista į morką, todėl Justinas gaus morką.

Kitas būdas spręsti šią užduotį – tikrinti galimus variantus nuo upės pabaigos, kur laukia Justinas. Žinoma, kad norint gauti rąstą pabaigoje, daiktas, plaukiantis po priešpaskutiniu tiltu, turi būti morka. Norint gauti morką, reikia po pirmo ir antro krioklių upių susijungimo turėti rąstą. Vienintelis būdas turėti rąstą toje vietoje – į antrą krioklį turi būti įmesta žuvis.

TAI INFORMATIKA

Galima galvoti apie kompiuterį kaip apie prietaisą, kuris nuskaito įvestį ir parašo išvestį. Kaip kompiuteris „žino“ ką reikia daryti? Atsakymas tas, kad žmonės iš anksto pasako, ką daryti!

Mes rašome programas, kurias kompiuteriai gali atlikti vėl ir vėl. Yra daug skirtingų programavimo kalbų. Vienas programavimo kalbų stilių yra funkcinis programavimas. Šis stilius panašus į patį kompiuterį, nes sukurtas iš daugybės funkcijų, nuskaitančių įvestį ir pateikiančių išvestį.

Tiltai šiame uždavinyje – tai mažos funkcijos, o visa sistema – programa, parašyta naudojant funkcinę programavimo kalbą.

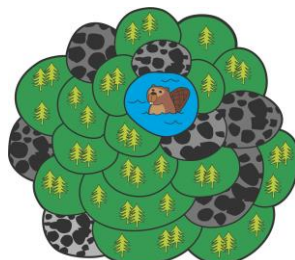
Raktiniai žodžiai: šakojimo algoritmas, programos testavimas.

Informatinis mąstymas: algoritmavimas.

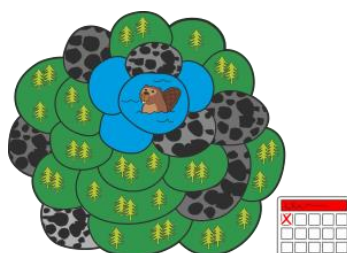
27. BEBRŲ EŽERAS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
		12	9	6		

Bebrai gyvena slėnyje, apsuptame kalnų. Tame slėnyje yra ežeras. Aplink ežerą yra miško arba uolų plotai.



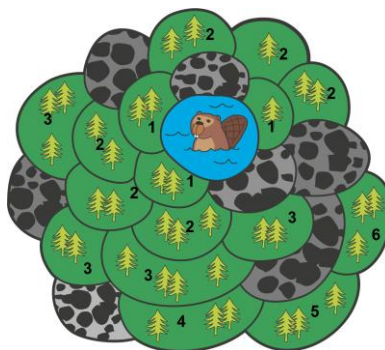
Kiekvieną dieną bebrai užtvindo visus miško plotus, esančius šalia ežero arba jau apsemtų plotų. Pavyzdžiui, po vienos dienos trys plotai bus apsemti. Uolų plotai nėra apsemiami.



Po kelių dienų visi miško plotai bus užtvindyti? Įveskite dienų skaičių.

PAAIŠKINIMAS

Teisingas atsakymas: po 6 dienų. Skaičiai paveikslėlyje nurodo miško plotus, kurie bus užlieti po kelių dienų. Miško plotai, esantys šalia ežero, pažymėti numeriu 1. tada visi dar nepažymėti miško plotai, esantys šalia pažymėtųjų, pažymimi numeriu 2. Šis procesas tęsiasi tol, kol sunumeruojami visi miško plotai. Kaip matome paveikslėlyje, paskutinis miško plotas bus užlietas po 6 dienų.



TAI INFORMATIKA

Kompiuterių mokslininkai tiria skirtingus algoritmų tipus. Algoritmas – veiksmų seka, vykdoma norint atlikti tam tikrą užduotį ar išspręsti problemą.

Šis uždavinys parodo bangos algoritmą (angl. *wavefront algorithm*), naudojamą peržiūrėti sritį, padalinti į laukelius. Ši peržiūra naudoja paieškos į plotį (angl. *breadth-first search*, BFS) metodą. BFS paieška prasideda laukelyje, vadinamame šaknimi. Kitu žingsniu applančios visos jos kaimynės, po to kaimynių kaimynės ir t.t., kol randamas ieškomas tikslas arba kol applančomi visi laukeliai. Šiame uždavinyje laukeliai atitinka plotus, o jų lankymas – miško plotų užliejimą. Kai BFS algoritmas atliekamas kruopščiai, kiekvienas laukelis applančomas tik vieną kartą. Tai atliekama organizuotai. Iš tiesų, kaip matyti iš pateikto sprendimo, laukeliai gali būti sunumeruojami, nes yra aplankyti. Šie skaičiai nurodo, kiek visi laukeliai nutolę nuo šaknies (šiuo atveju, ežero). Tai reiškia, kad BFS algoritmas gali būti naudojamas, siekiant rasti trumpiausią kelią nuo šaknies iki visų kitų laukelių.

https://lt.wikipedia.org/wiki/Paie%C5%A1ka_%C4%AF_plot%C4%AF

Raktiniai žodžiai: bangos algoritmas, paieška į plotį.

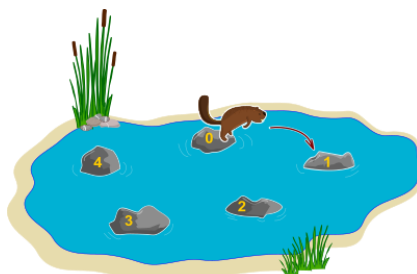
Informatinis mąstymas: algoritmavimas.

28. BEBRIŠKA MODULIO OPERACIJA

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
			9	6		

Keli bebrai dalyvavo varžybose. Pirmoji užduotis buvo šuoliuoti nuo vieno akmens iki kito laikrodžio rodyklių sukimosi kryptimi, pradedant nuo akmens, pažymėto numeriu 0. Jei bebras padarytų 8 šuolius, jis sustotų ant akmens, kurio numeris 3:

$0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3$



Vienas bebras sumanė pasirodyti prieš kitus ir padarė 129 šuolius. Ant kurio akmens jis sustojo?

PAAIŠKINIMAS

Atkreipkime dėmesį, kad jei bebras padarytų 5 šuolius, jis sustotų toje pačioje vietoje, nuo kurios ir pradėjo. Pavadinkime tai ratu. Norėdami išsiaiškinti kur bebras sustos, padaręs 129 šuolius, turime suskaičiuoti, kiek bus padaryta ratų ir kiek šuolių dar lieka. Šiuo atveju: $129 = 25 \times 5 + 4$. Vadinasi, padaręs 129 šuolius, bebras sustos toje pačioje vietoje, kaip ir būtų padaręs 4 šuolius. Bebras sustoja ant akmens, kurio numeris 4.

TAI INFORMATIKA

Uždavinyje reikia apskaičiuoti skaičiaus 129 dalybos iš 5 liekaną. Šis veiksmas labai dažnai naudojamas kompiuteriuose. Matematikoje tai vadinama dalyba su liekana (rezultatas: dalybos liekana), o informatikoje operacija vadinama moduliu ir žymima santrumpa mod. Mūsų atveju būtų: $129 \bmod 5 = 4$.

Ši operacija vartojama cikluose (visai kaip bebrui šuoliuojant) arba kai kintamųjų reikšmės išeina iš rėžių ir panašiose situacijose.

Raktiniai žodžiai: operacija moduliui.

Informatinis mąstymas: algoritmavimas.

29. ŠUOLIAI LENTOJE

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
			9	9	6	

Lentoje yra 8 langeliai, sužymėti numeriais nuo 1 iki 8. Kiekvienam langeliui priskiriama viena iš trijų judėjimo taisyklių. Kiekvienos taisyklės pavyzdys pateikiamas žemiau:

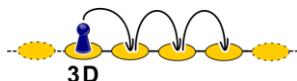
1. Judėti į kairę

Pavyzdžiui, 2K reiškia, kad reikia judėti per du langelius į kairę:



2. Judėti į dešinę.

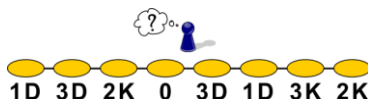
Pavyzdžiui, 3D reiškia judesį per tris langelius į dešinę:



3. Nejudėti

Taisyklė „0“ reiškia, kad iš šio langelio judėti nereikia.

Įsivaizduokite tokią lentą:



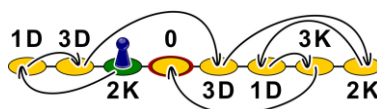
Iš kurio langelio reikia pradėti, kad judant pagal nurodytas taisykles būtų aplankytas kiekvienas langelis?

- A. 2 B. 3 C. 5 D. Neįmanoma aplankyti kiekvieno langelio.

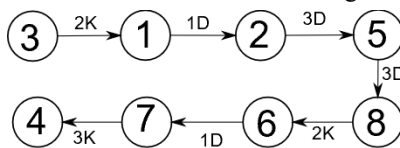
PAAIŠKINIMAS

Teisingas atsakymas: 3.

Analizuodami sprendinį nuo galinio taško (langelio su taisykle „0“), matome, kad tas taškas gali būti pasiektas iš 7 langelio, į kurį patenkama iš 6 langelio, o šis pasiekiamas iš 8 langelio, į kurį patenkama iš 5 langelio, šis pasiekiamas iš 2 langelio, į kurį ateinama iš 1 langelio, o pastarasis pasiekiamas iš 3 langelio.



Visa tai galima pavaizduoti grafu, kurio viršūnės sužymėtos langelių numeriais, o briaunos – taisyklėmis, kaip judėti tarp šių langelių. Šis grafas gali būti konstruojamas, pradedant bet kuria viršūne, ir baigiamas tada, kai surašyti visi langelių numeriai.



TAI INFORMATIKA

Apie judesį „3D“ iš 2 į 5 langelį galime galvoti kaip apie rodyklę. Tokių rodyklių rinkinyje, kuris yra orientuotas grafas, ieškoma viršūnės arba aukštesnio lygio mazgo.

Laikytis rodyklių sekos labai svarbu, pavyzdžiui, tvarkant operacinės sistemos atmintį arba įgyvendinant Javos programavimo kalbos šiukšlių surinkimo mechanizmą, kad galėtų būti atlaisvinta nebenaudojama vieta atmintyje. Daug programinės įrangos klaidų gali būti rasta analizuojant problemines komandas, kai grįžtama atgal iki jų aukštesnio lygio, pradinės priežasties.

Šiuo žaidimu gali būti modeliuojamas nukreipimo sakiny programavime. Nukreipimo sakiny nurodo, kad užuot vykdžius iš eilės einančią gretimą komandą, reikia eiti į visai kitą programos vietą ir nuo tos vietos toliau vykdyti komandas. Šiame uždavinyje „programa“ – tai nukreipimo sakinių seka su vienintele baigimo komanda 4 langelyje.

Raktiniai žodžiai: Tiuringo juosta, rodyklė, kelias.

Informatinis mąstymas: algorit mavimas, duomenų apdorojimas.

30. DVEJETAINIS VIEŠBUTIS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
			12	6		

Matas dirba viešbutyje. Jo šiandienos užduotis – ant kiekvieno namelio pakabinti naują kambario numerį. Pakabinęs 50-ąjį numerį, toliau jis kabina numerius pagal šias taisykles:

- jei naujas namelio numeris mažesnis už namelio, prie kurio stovi, numerį, eina į kairę,
- jei naujas namelio numeris didesnis už namelio, prie kurio stovi, numerį, eina į dešinę,
- jei namelis, prie kurio stovi, neturi numerio, pakabina naują numerį.

Kai kurie nameliai jau sunumeruoti:



Matas turi pakabinti 29 numerį. Remdamiesi taisyklėmis, pažymėkite namelį, ant kurio Matas turi pakabinti šį numerį.

PAAIŠKINIMAS

Teisingas atsakymas yra:



Pradedant nameliu, esančiu greta 50-ojo, 29 yra mažiau nei 50, todėl Matas eina į kairę. Kito namelio numeris 24. 29 yra daugiau už 24, todėl eina į dešinę. Kito namelio numeris 34, o 34 yra mažiau už 29, todėl eina į kairę. Šis namelis dar neturi numerio, ant jo ir pakabinamas naujas numeris.

TAI INFORMATIKA

Nameliai atitinka dvejetainio medžio viršūnes. Dvejetainis paieškos medis yra bendra duomenų struktūra informatikoje. Ji naudojama ieškomiems duomenims saugoti. Kiekviena viršūnė tenkina šias savybes:

- n-tosios viršūnės reikšmė didesnė už visas reikšmes jos kairiajame pomedyje.
- n-tosios viršūnės reikšmė mažesnė už visas reikšmes jos dešiniajame pomedyje.

Kairysis ir dešinysis pomedžiai yra dvejetainiai paieškos medžiai. Todėl, kai ieškome viršūnės, žiūrime tik į kairę arba dešinę, bet ne į abi puses.

[https://lt.wikipedia.org/wiki/Medis_\(duomen%C5%B3_strukt%C5%ABra\)#Dvejetainiai_med%C5%BEiai](https://lt.wikipedia.org/wiki/Medis_(duomen%C5%B3_strukt%C5%ABra)#Dvejetainiai_med%C5%BEiai)

Raktiniai žodžiai: dvejetainis paieškos medis, algoritmas.

Informatinis mąstymas: duomenų apdorojimas.

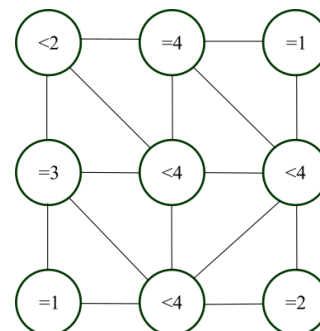
31. JUNGTYS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
			12	9		

Reikia nuspalvinti kai kuriuos apskritimus paveikslėlyje. Atkarpomis sujungtų apskritimų poras vadinsime kaimynais. Skaičiai apskritimuose (sąlygos) rodo, kiek kaimynų turi būti nuspalvinta. Pavyzdžiui, reikia nuspalvinti lygiai 3 iš 4 apskritimo (kuriame įrašyta „=3“) kaimynus ir mažiau negu 4 apskritimo (kuriame įrašyta „<4“) kaimynus.

Kiek apskritimų iš viso reikia nuspalvinti, kad visos sąlygos apskritimuose būtų tenkinamos?

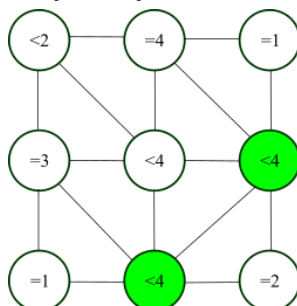
A) 4 B) 5 C) 6 D) 7



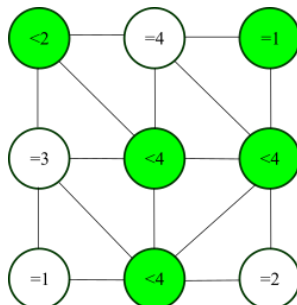
PAAIŠKINIMAS

Teisingas atsakymas: 5.

Pažiūrėję į apatinę dešinią apskritimą, matome, kad abu jo kaimynai turi būti nuspalvinti:



Taip pat visi apskritimo („=4“) kaimynai turi būti nuspalvinti:



Taip nuspalvinus, visų apskritimų sąlygos yra tenkinamos ir daugiau nieko negalima spalvinti:

- jei nuspalvintume „=1“, tai „=3“ būtų netenkinama,
- jei nuspalvintume „=2“, tai „<4“ (virš) būtų netenkinama,
- jei nuspalvintume „=3“ arba „=4“, tai „<2“ būtų netenkinama.

TAI INFORMATIKA

Šį uždavinį sprendžiame, pasitelkę logiką. Matome, kad visiško perrinkimo (arba jėgos) metodas (*brute-force*) nėra efektyvus. Jei naudotume visiško perrinkimo metodą, pastebime, kad kiekvienas apskritimas turi dvi galimybes. Tada bendras perrinkimų skaičius būtų lygus $2^9 = 512$. Taigi vienas sprendimų būtų išbandyti visos 512 skirtingų apskritimų užpildymo kombinacijų. Tačiau pasitelkus logiką, ypač dedukcinę samprotavimų seką, gerokai sumažinamas galimų užpildymų skaičius, ir uždavinys nesunkiai išsprendžiamas.

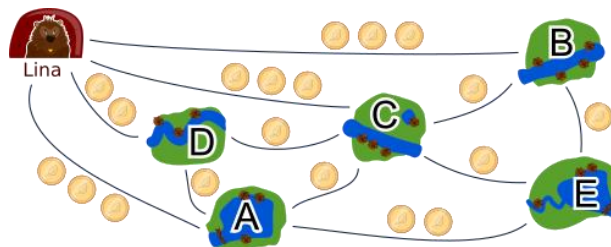
Raktiniai žodžiai: dedukcinis samprotavimas, logika, sąlyga.

Informatinis mąstymas: loginis pagrindimas.

32. DRAUGŲ LANKYMAS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
			12	9	6	

Linos draugai gyvena A, B, C, D, E kaimeliuose. Ji nori juos visus aplankyti per vieną dieną, naudodamasi viešuoju transportu. Kiekvieną kaimelį Lina aplanko tik vieną kartą ir tą pačią dieną sugrįžta namo. Nurodyta kiekvieno atstumo tarp kaimelių kaina. Didžiausia vieno atstumo kaina – 3 monetos.



Visų draugų aplankymas Linai kainuos 11 monetų, jeigu ji pasirinktų tokį maršrutą:

Namai → B → E → A → D → C → Namai.

Kokiu maršrutu keliaudama Lina išleistų mažiausiai monetų? Jei tokie maršrutai keli, pakanka surasti vieną jų.

Spustelėk kaimelių pavadinimų raides, kad pažymėtum maršrutą.

Namai → ____ → ____ → ____ → ____ → ____ → Namai

PAAIŠKINIMAS

Galimi du teisingi atsakymai:

Namai → B → E → C → A → D → Namai

Namai → D → A → C → E → B → Namai

Abu teisingi sprendimai yra atvirkštiniai vienas kitam. Daugiau teisingų atsakymų nėra, nes vienas kelionės atstumas iš namų arba namo Linai kainuoja 3 monetas arba 2 monetas, todėl jei kitiems keturiems atstumams, ir lieka po 1 monetą, vis tiek iš viso bus 9 monetos.

Kiti sprendimai brangesni:

10 monetų: Namai → A → D → C → E → B → Namai

10 monetų: Namai → A → E → B → C → D → Namai

10 monetų: Namai → B → C → E → A → D → Namai

10 monetų: Namai → B → E → A → C → D → Namai

10 monetų: Namai → B → E → C → D → A → Namai

10 monetų: Namai → C → B → E → A → D → Namai

10 monetų: Namai → D → A → E → B → C → Namai

10 monetų: Namai → D → A → E → C → B → Namai

10 monetų: Namai → D → C → A → E → B → Namai

10 monetų: Namai → D → C → B → E → A → Namai

11 monetų: Namai → B → E → A → D → C → Namai

11 monetų: Namai → C → D → A → E → B → Namai

TAI INFORMATIKA

Optimalaus sprendimo paieška – pagrindinis informatikos uždavinys. Galima vizualizuoti šio optimizavimo uždavinio aprašą grafu, kur draugų kaimeliai būtų mazgai, o atstumai tarp jų – grafo briaunos. Šis uždavinys taptų trumpiausio kelio (mažiausiai monetų) uždaviniu. Tai panašu į garsųjį Keliaujančio pirklio uždavinį. Tokius uždavinius sudėtinga spręsti kompiuteriu. Vengiant gausybės sprendimų tikrinimo, dažniausiai pasirenkamas euristinis algoritmas.

https://lt.wikipedia.org/wiki/Keliaujan%C4%8Dio_pirklio_u%C5%BEdavinys

Raktiniai žodžiai: keliaujančio pirklio uždavinys, optimizavimas.

Informatinis mąstymas: algoritmavimas.

33. LEMPUTĖS IR JUNGIKLIAI

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
			12	9	6	

Mėgėjas meistrauti Vytas sujungė 6 lemputes su 6 jungikliais. Kiekvienas jungiklis valdo tik vieną lemputę. Tačiau nežinoma, kuri jungiklio būseną įjungia ar išjungia lemputę.

Kad sužinotume, kuris jungiklis valdo kurią lemputę, atlikome eksperimentą su jungikliais. Keturių bandymų rezultatai pateikiami piešinyje. Raidėmis žymimi jungikliai, o skaičiais – lemputės.

											
A	B	C	D	E	F	1	2	3	4	5	6
											
A	B	C	D	E	F	1	2	3	4	5	6
											
A	B	C	D	E	F	1	2	3	4	5	6
											
A	B	C	D	E	F	1	2	3	4	5	6

Kuris jungiklis kurią lemputę valdo?

- A) 1 – C, 2 – F, 3 – E, 4 – A, 5 – D, 6 – B
 B) 1 – C, 2 – E, 3 – D, 4 – A, 5 – F, 6 – B
 C) 1 – C, 2 – F, 3 – D, 4 – E, 5 – A, 6 – B
 D) 1 – C, 2 – F, 3 – B, 4 – A, 5 – D, 6 – E

PAAIŠKINIMAS

Teisingas atsakymas: A.

1 – C, 2 – F, 3 – E, 4 – A, 5 – D, 6 – B

Paprasciausia pradėti nuo 6-os lemputės. Ji šviečia per visus keturis bandymus, o yra tik B mygtukas, kurio būseną taip pat nesikeičia per visus 4 bandymus.

2-a lemputė šviečia tik ketvirtame bandyme, todėl reikia ieškoti jungiklio, kurio būseną keičiasi tik paskutiniame bandyme. Tai F mygtukas.

3-a lemputė šviečia tik pirmame bandyme, taigi ieškome jungiklio, kuris tik po pirmojo bandymo keičia savo būseną. Tai E mygtukas.

4-a lemputė šviečia tik trečiajame bandyme, todėl ją atitinka tik A jungiklis.

1-a lemputė šviečia antrame ir trečiajame bandyme, o likusi, 5-a lemputė, šviečia pirmame ir antrame bandyme. Taigi šių lempučių būsenos nėra priešingos, todėl lengva nustatyti, kad 1-ą lemputę valdo jungiklis C, o 5-ą – D.

TAI INFORMATIKA

Kompiuterinėse sistemose svarbu nustatyti veiksmų seką. Tai būtina, jei jūs programuojate ir reikia programą patobulinti ar taisyti.

Derinimas (priežasties, kuri sukelia neteisingą programos veikimą, ieškojimas) svarbus, tobulinant programą. Ieškant klaidų, dažniausiai keliamos hipotezės arba atliekami maži kontroliniai eksperimentai.

<http://ims.mii.lt/EK%C5%BD/enciklo.html?word=derinimas>

Raktiniai žodžiai: seka, derinimas, atpažinimas.

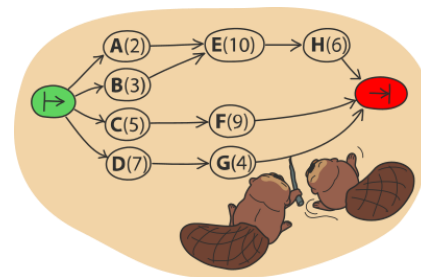
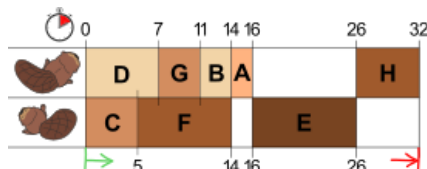
Informatinis mąstymas: uždavinio skaidymas.

34. DU BEBRAI DIRBA

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
			12	9	6	

Du bebrai stato užtvanką ir turi atlikti 8 užduotis (nukirsti medžius, pašalinti šakas, surinkti kamienus ir pan.): A(2), B(3), C(5), D(7), E(10), F(9), G(4), H(6). Skaičius skliaustuose nurodo, kiek valandų reikia užduočiai atlikti. Kai kurias užduotis galima atlikti tik užbaigus ankstesnes, tai nurodoma rodyklėmis paveiksle. Bebrai darbuojasi vienu metu, tačiau kiekvienas imasi atskiros užduoties.

Bebrai naudoja strategiją: iš galimų užduočių pasirenka ilgiausiai trunkančią. Užduočių atlikimo planas atrodytų šitaip:



Bebrai pastatytą užtvanką per 32 valandas. Taikant kitokią strategiją, užtvanką galima būtų pastatyti per trumpesnę laiką. Kiek mažiausiai valandų reikia dviem bebrams užtvankai pastatyti?

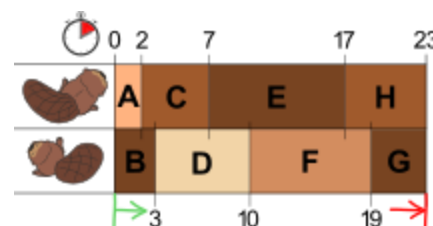
PAAIŠKINIMAS

Teisingas atsakymas: 23.

Uždavinio lentelėje parodytas dviejų bebrų užduočių atlikimo planas. Matome, kad pirmasis bebras neturi darbo gana ilgą laiką, net 8 valandas, o antrasis – 6 valandas. Geriausia būtų, jeigu jie galėtų dirbėti visą laiką. Taikysime kitą strategiją: planuosime, kad dvi ilgiausiai trunkančios užduotys E(10) ir F(9) neatitektų tam pačiam bebrui. Pateikiame vieną tokių užduočių paskirstymo planų.

Matome, kad užtvanką galima pastatyti net per 23 valandas!

Atkreipkite dėmesį, kad ši strategija leido atlikti darbą per trumpiausią laiką todėl, kad abu bebrai galėjo dirbėti be pertraukų.



TAI INFORMATIKA

Kai kuriems uždaviniams bebrų strategija (imk ilgiausiai trunkančią užduotį iš likusių) gali pateikti optimalų sprendimą – užduotys gali būti atliktos per trumpiausią laiką. Kitiems uždaviniams (kaip pateikta čia) geriau tinka ilgiausiai trunkančių užduočių paskirstymas skirtingiems atlikėjams. Deja, kiekvienai šiai strategijai galima rasti uždavinių pavyzdžių, kuriems ji netinka. Taip yra todėl, kad šioms uždaviniams nėra bendros procedūros (sprendimo) trumpiausio laiko trukmei rasti – optimalų sprendimą galima rasti tik išbandant visus galimus užduočių paskirstymo variantus! Paprastai tai nėra realu, ir net skaičiuojant kompiuteriu procesas gali trukti ilgiau, nei vienas bebras pastatytų visą užtvanką!

Šio uždavinio pavyzdys buvo kruopščiai parinktas, kad bebrų strategija (imk ilgiausiai trunkančią užduotį iš likusių) netiktų. Tai nebuvo lengva. Daugumai uždavinių ši bebrų strategija (dažnai vadinama godžiaja) tinka gana gerai, galima greitai paskirstyti užduotis, ir bebrai gali imtis dirbėti nieko nelaukdami.

Sukurti pavyzdžius, neatinkančius kurios nors sprendimo strategijos (kaip buvo daroma rengiant šį uždavinį), reikia kūrybiškumo ir gilaus strategijos suvokimo. Tačiau šie įgūdžiai reikalingi informatikui norint nustatyti, pavyzdžiui, ilgiausią programos atlikimo kompiuteriu laiką, kas vartojama algoritmų analizėje (skaičiavimų sudėtingumo teorijos srityje).

Raktiniai žodžiai: planavimas, trumpiausia laiko trukmė, užduočių paskirstymas, optimalus sprendimas.

Informatinis mąstymas: algoritmavimas.

35. KOMPIUTERINIS ŽAIDIMAS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
			12	9	6	

Audra suprogramavo mokykloje kompiuterinį žaidimą. Žaidimo taisyklės paprastos:

- Žaidimą sudaro etapai. Per vieną etapą krenta vienas lapas. Bebras bando pagauti lapą, kol jis nepasiekė žemės. Kad laimėtų, bebras turi pagauti 15 lapų. Žaidimas pralaimimas, kai nukrenta 4-asis lapas.
- Žaidimo trukmė matuojama etapų skaičiumi.

Pateiktame pavyzdyje bebras pralaimi po 6 etapų, nes nepagavo 4 lapų. Šio žaidimo trukmė – 6 etapai.

Etapai	Rezultatas	Bendras lapų skaičius	
		pagautų	nepagautų
1 etapas	pagauta	1	0
2 etapas	nepagauta	1	1
3 etapas	pagauta	2	1
4 etapas	nepagauta	2	2
5 etapas	nepagauta	2	3
6 etapas	nepagauta	2	4

Kiek ilgiausiai gali trukti šis žaidimas?

- A. 4 etapai B. 15 etapų C. 18 etapų D. 19 etapų E. 20 etapų F. Trunka neribotai

PAAIŠKINIMAS

Norėdami nustatyti galimai ilgiausią žaidimo trukmę, turime suderinti visas sąlygas, kurioms esant vyksta žaidimas. Tam apjungiame maksimalų sugautų lapų skaičių (15 etapų) ir maksimaliai leistiną nepagautų lapų skaičių (3 lapus, nes 4 lapai jau sustabdytų žaidimą). Jei žaidimas buvo užbaigtas nepagavus 4 lapų, tuomet pagautų lapų skaičius gali būti tik 14, kitaip žaidimas būtų baigtas. Todėl maksimali žaidimo trukmė yra: $15 + 3 = 14 + 4 = 18$ etapų. Teisingas atsakymas: c).

a) 4 etapai yra apskritai minimali žaidimo trukmė (kai visi keturi lapai nepagauti).

b) 15 etapų yra minimali laimėto žaidimo trukmė (kai visi lapai pagaunami).

Kiti atsakymai yra neteisingi, nes maksimalus pagautų ar nepagautų lapų skaičius pasiekiamas, o žaidimas tebevyksta.

TAI INFORMATIKA

Kai norima suprogramuoti žaidimą, jo taisyklės turi būti vienareikšmiškai nusakytos. Svarbu, kad būtų galima laimėti arba pralaimėti (yra pakankamai lapų) ir žaidimo trukmė būtų nei per trumpa, nei per ilga.

Žaidimo etapai nusako procesą. Informatikai turi mokėti modeliuoti ir aprašyti procesus. Vienas iš svarbiausių tikslų yra suprasti, kas gali nutikti esant įvairioms sąlygoms ir kiek ilgai gali trukti visas procesas.

Raktiniai žodžiai: logika, baigties sąlygos, procesas, derinimas.

Informatinis mąstymas: algorit mavimas, loginis pagrindimas.

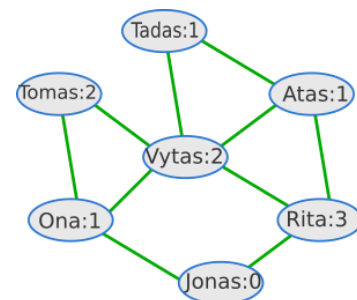
36. DOVANOS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
				12	6	

Paveikslėlis vaizduoja vaikų draugystę. Du vaikai draugauja tik tada, jei jų vardai sujungti.

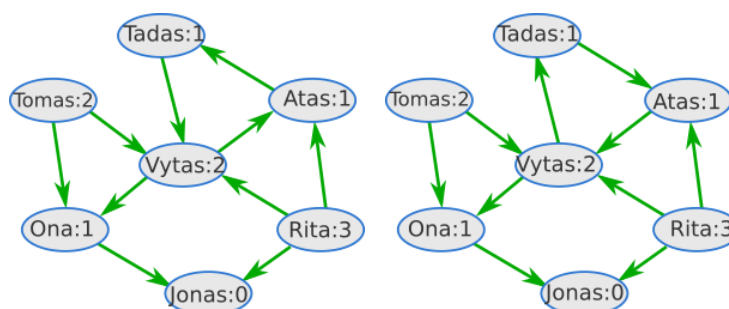
Vaikai planuoja vakarėlį. Vienas vaikas iš draugų poros perka tik vieną dovaną kitam poros draugui. Paveikslėlyje prie vardo parašyta, kiek dovanų vaikas gali pirkti. Nustatykite kryptis kiekvienai draugų porai, kuris turi pirkti ir įteikti dovaną draugui. Kiekvienas negali pirkti daugiau dovanų nei parodyta paveikslėlyje.

Spustelk draugus jungiančią liniją ir nustatyk kryptį. Jei rodyklė rodo nuo draugo A į draugą B, tai reiškia, kad A perka dovaną draugui B.



PAAIŠKINIMAS

Yra du galimi būdai, atitinkantys nurodytas taisykles.



Pradėkime nuo Jono. Jonas neturi pinigų, todėl jis gaus dovanas iš Onos ir Ritos. Ona išleis visus savo pinigus, bet dovanas gali gauti iš Vyto ir Tomo. Kito pasirinkimo nėra.

Tadas ir Atas gali pirkti tik po vieną dovaną. Reikia pasirinkti, kuris draugas dovaną perka. Jei pasirenkame Atą, jis turės gauti dovaną iš Vyto. Tada Vytas jau nebeturės pinigų ir gaus dovanas iš Tado, Tomo ir Ritos. Rita turės pirkti dovaną Atui. Tokį sprendimą vaizduoja kairysis paveikslėlis.

Jei pasirenkame Tadą, jis gaus dovaną iš Vyto. Vytas gaus dovanas iš Ato, Tomo ir Ritos. Rita dar pirks dovaną Atui. Tokį sprendimą vaizduoja dešinysis paveikslėlis.

Tadas ir Atas nulemia kitų draugų dovanų pirkimą ir dovanojimą. Taigi gaunami tik du skirtingi atvejai.

TAI INFORMATIKA

Iš vaikų sudaromas draugystės tinklas su mazgais (vaikais) ir saitais (draugyste). Tai primena gerai žinomus žmonių socialinius tinklus. Tačiau tarp šių tinklų svarbus skirtumas: kai kuriuose iš jų yra abipusė „draugystė“ (t. y. saitai be krypties), kaip ir šioje „Bebro“ užduotyje. Kituose tinkluose yra „pasekėjai“, kai saitai turi kryptį. Pavyzdžiui, Jūs galite sekti tam tikrą žymų asmenį, bet jis gali Jūsų ir nesekti.

Šiame uždavinyje tiesiami dvikrypčiai draugystės saitai. Tai skiriasi nuo sekimo, nes jei vaikai ribojami dovanų skaičiumi ir pinigais jų įsigyti. Skaičiuojame, kiek dovanų gali gauti kiekvienas vaikas. Tai primena informatikos uždavinį, kai tinkle ribojami tam tikrų taisyklių saitai, tačiau rezultatas turi turėti kuo daugiau galimų saitų. Šiame uždavinyje galias turi mazgai, o informatikoje – tinklų saitai. Nesunku šį uždavinį keisti į realaus tinklo uždavinį.

Raktiniai žodžiai: grafas, mazgas, briauna, dvikryptis tinklas.

Informatinis mąstymas: abstrahavimas.

37. KORTELIŲ VARTYMAS

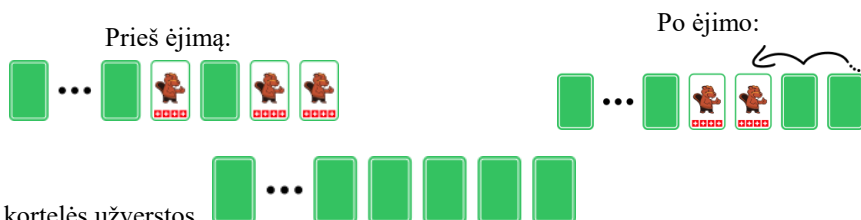
Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
				12	9	

Kortelės sudėliotos eile. Kiekviena kortelė gali būti atversta  arba užversta .

Vieną žaidimo ėjimu galima atlikti tokius veiksmus:

- Kortelės peržiūrimos iš dešinės į kairę.
- Jei peržiūrima kortelė užversta, ji atverčiama, ir ėjimas baigiamas.
- Jei peržiūrima kortelė atversta, ji užverčiama, ir pereinama prie kitos kortelės.
- Žaidimas baigiamas, kai peržiūrėtos visos kortelės.

Paveiksle vaizduojama, kaip pasikeičia kortelės, atlikus vieną ėjimą. Peržiūrima iš dešinės: dvi pirmos kortelės atverstos, todėl jas užverčiame, trečioji kortelė užversta – ją atverčiame ir ėjimas baigiamas.



Žaidimo pradžioje visos 32 kortelės užverstos.

Kiek bus atverstų kortelių, atlikus 32 žaidimo ėjimus?

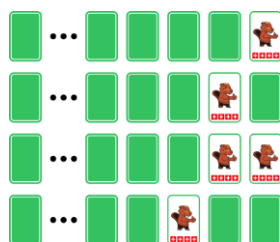
PAAIŠKINIMAS

Atsakymas: 1.

Yra kelios galimybės rasti teisingą atsakymą. Atlikus pirmuosius ėjimus, išaiškėja dėsningumas. Po pirmo, antro, ketvirto, aštunto ėjimo atversta lieka tik viena kortelė, kuri po pirmo, antro, ketvirto, aštunto ėjimo kas kartą pasislenka per vieną į kairę. Galime daryti išvadą, kad po 16-ojo ėjimo bus vėl atversta viena kortelė – penktoji iš dešinės, o po 32-ojo ėjimo bus atversta viena kortelė – šeštoji iš dešinės.

Kitas būdas. Pastebime, kad, du vienas paskui kitą ėjimai turi tokį pat efektą, kaip ir vienas ėjimas eilėje, turintis viena kortelę iš dešinės mažiau. Tada ėjimai duoda tokį patį efektą, kaip du ėjimai eilėje, kurioje yra viena kortelė mažiau. Tai atitinka vieną ėjimą eilėje, kuri trumpesnė dviem kortelėmis. Šitaip 8, 16, 32 ėjimai formuoja vieną ėjimą eilėse, kurios yra atitinkamai trumpesnėje.

Galima patyrinti pateiktą kortelių pavyzdį. Matome, kad pačios dešiniausios kortelės būsenos seka kartojasi kas dvi kortelės. Reiškia, esant lyginiam ėjimui, ji bus užversta. Antros kortelės būseną kartojasi kas keturi ėjimai, reiškia, kortelė 4 ėjimu bus užversta. Trečia – kas 8 kortelės ir t. t.



TAI INFORMATIKA

Kompiuteriuose skaičiai vaizduojami dvejetainiais skaitmenimis – nulių ir vienetų, kurie užima vieną bitą informacijos. Vienetas vaizduojamas 0...00001, du – 0...00010, trys – 0...00011, 4 – 0...00100, 0...00101, 0...00110, ir t. t. Šiuolaikiniai kompiuteriai grindžiami 32 bitų logika. Kortelių seką vaizduojame taip: nulių (0), jei kortelė užversta, ir vienetų (1), jei ji atversta. Tad būtų pastebimi tie patys dėsningumai. Žinant, kad 32 yra vaizduojami kaip 0...0100000, galima įsitikinti, kad tai ir bus uždavinio atsakymas. Kiekvienu uždavinio ėjimu vaizduojamas dvejetainis skaičius padidinamas vienetu.

Raktiniai žodžiai: seka, sekos dėsningumas, dvejetainiai skaičiai.

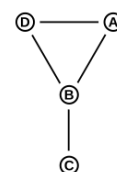
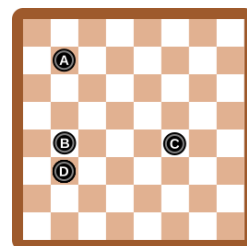
Informatinis mąstymas: algorit mavimas, duomenų apdorojimas.

38. STULPELIAI IR EILUTĖS

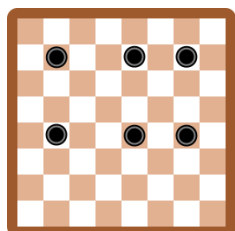
Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
				12	9	

Duota žaidimo lenta, kurioje stovi figūros. Lentą bandoma pavaizduoti diagrama taip:

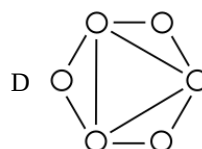
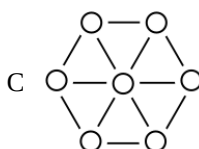
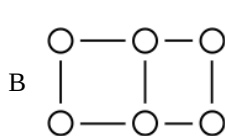
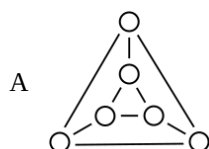
- Kiekvienai figūrai lentoje nupiešiamas apskritimas.
- Jei dvi figūros lentoje yra toje pačioje eilutėje ar tame pačiame stulpelyje, tarp jų diagramoje nubrėžiama linija (kitais atvejais linijos nebrėžiamos).



Figūros pažymėtos raidėmis tik tam, kad būtų lengviau patikrinti, ar diagrama teisinga.

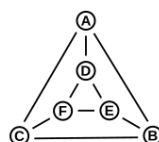
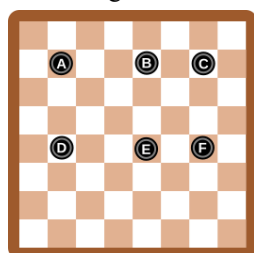


Kuri diagrama nupiešta šiai lentai?

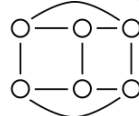


PAAIŠKINIMAS

Surašius figūrų raides, nesunku patikrinti, kad duotai lentai tinka diagrama A.



Lengviausia patikrinti, kad kitos trys diagramos netinka, pastebėjus, kad lentoje kiekviena figūra toje pačioje eilutėje sujungta su dviem kitomis figūromis ir tame pačiame stulpelyje su dar viena. Taigi, diagramoje kiekvienas apskritimas turi būti sujungtas su 3 kitais apskritimais. (Atsakyme C yra vienu apskritimu per daug.). Diagrama B atrodo panaši į teisingą, bet kairiausi ir dešiniausi apskritimai sujungti tik su dviem kitais apskritimais, todėl ji negali būti teisinga. Norint ją ištaisyti, užtenka



pridėti dvi linijas taip:

TAI INFORMATIKA

Informatikos moksle diagramos, kaip šiame uždavinyje, vaizduoja esminę uždavinio informaciją. Tokios diagramos vadinamos grafais. Diagramoje apskritimais vaizduojamos grafo viršūnės, o juos jungiančios atkarpos, – briaunomis.

Svarbu, ar grafo viršūnės yra sujungtos, ar ne. Tas pats grafas dažnai gali būti vaizduojamas skirtingais būdais, kaip pavaizduota anksčiau: atsakymas A ir paskutinio paaiškinimo paveikslėlio diagramos yra teisingos ir vaizduoja tą patį grafą.

Informatikos uždaviniuose grafai naudojami informacijos vaizdavimui ir priklauso nuo uždavinio. Pavyzdžiui, jei norite žinoti, ar balta figūra lentoje yra toje pačioje eilutėje ar stulpelyje kaip juoda, tada duomenų vaizdavimas grafu nėra geras būdas: viršūnės nevaizduojamos figūrų spalvomis. Jei norime žinoti, ar lentoje liko pakankamai figūrų, kad laimėtume žaidimą, tai grafas „perkrautas“: turi sekti, kiek liko figūrų lentoje kiekvienu momentu ir nėra prasmės įrašyti, kuri figūra yra toje pačioje eilutėje ar langelyje kaip kitos figūros. Kita vertus, grafas – geras būdas pavaizduoti atsakymus į klausimą, koks yra minimalus figūrų skaičius, kurį turi pašalinti, kad nebūtų figūros tame pačiame eilutės stulpelyje, kaip bet kurios kitos figūros.

Teisingas duomenų vaizdavimo būdas yra vienas iš iššūkių programuotojui ar informatikos mokslininkui.

Raktiniai žodžiai: grafas, diagrama, vaizdavimas.

Informatinis mąstymas: abstrahavimas.

39. KNYGŲ RINKINIO RIKIAVIMAS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	Chinese Taipei (Taiwan Province)
				12	9	

Trys bebrai turi po savo stalą. Ant kiekvieno stalo padėta po dvi knygas. Knygų tvarka sumaišyta, ir bebrai nori jas surikiuoti atlikdami apkeitimo žingsnius.

Yra du žingsnių tipai.

- Pirmojo tipo žingsnis: kiekvienas bebras gali sukeisti dvi ant stalo gulinčias knygas (A pavyzdys).
- Antrojo tipo žingsnis: bebrai gali susikeisti kaimyninėmis knygomis su šalia esančiu stalu (B pavyzdys).



Pirmu žingsniu kiekvienas bebras sukeičia knygas ant savo stalo.

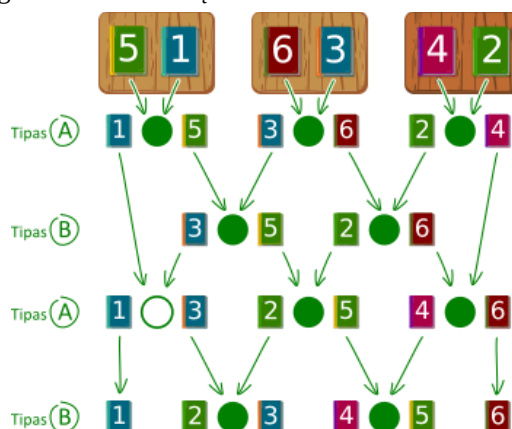
Kiek mažiausiai žingsnių reikia, kad knygos būtų išrikiuotos 1, 2, 3, 4, 5, 6?

- A. trys žingsniai B. keturi žingsniai C. penki žingsniai D. šeši žingsniai

PAAIŠKINIMAS

Teisingas atsakymas: B.

Šis uždavinys yra lygiagretaus tinklo rikiavimo algoritmo versija. Pirmiausia jie pradeda sukeisdami knygas ant kiekvieno stalo, kaip nurodyta uždavinyje. Godus algoritmas nusprendžia, ar atlikti apkeitimą, ar ne. Antras žingsnis: vienintelis galimas veiksmas – visi bebrai stengiasi sukeisti knygas tarp stalų. Tuomet vienintelis kito žingsnio galimas veiksmas – sukeisti knygas ant kiekvieno stalo ir t. t. Godus algoritmas intuityvus ir optimalus. Vykdamas algoritmą, galima lengvai rasti atsakymą. Toliau pateiktame paveikslėlyje parodyti minėtos strategijos veiksmas pagal rikiavimo tinklą.



TAI INFORMATIKA

Objektų išdėstymas eile pagal kurį nors jų parametą, skaičiumi – pagal jų dydį, žodžiais – pagal raidžių rikiavimo eilę abėcėlėje ir pan. Rikiuoti galima dvejopai: didėjančiai arba mažėjančiai. Jeigu objektai skaidomi į kelias grupes, sakoma, kad jie rūšiuojami. Rūšiavimas gali būti panaudotas kaip pagalbinis rikiavimo veiksmas. Objektai surūšiuojami į grupes (rūšis), po to grupės sujungiamos, ir gaunama surikiuotų objektų eilė. Atkreipiame dėmesį, kad anglų kalboje neskiriamos rikiavimo ir rūšiavimo sąvokos. Į tai reikia atsižvelgti, verčiant programas.

Raktiniai žodžiai: godus algoritmas, lygiagretus tinklas, rikiavimas.

Informatinis mąstymas: algoritmavimas.

40. SOUNDEX ALGORITMAS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
				12		

Soundex algoritmas skirtas atpažinti angliškus vardus pagal skambesį ir juos indeksuoti. Kiekvienam vardui generuojamas 4 skaitmenų kodas pagal šias taisykles:

1. Palikti pirmąją žodžio raidę.
2. Pašalinti visas šias raides: A, E, I, O, U, H, W, Y.
3. Likusias raides pakeisti skaitmenimis iš lentelės:

Raidės	Skaitmenys
B, F, P, V	1
C, G, J, K, Q, S, X, Z	2
D, T	3
L	4
M, N	5
R	6

4. Du ar daugiau iš eilės einančius tuos pačius skaitmenis pakeisti vienu tuo pačiu skaitmeniu.
5. Imti tik pirmuosius keturis žodžio simbolius; jei žodis trumpesnis, pabaigoje pridėti nulį.

Pavyzdys:

Žodis	Kodas
BOB	B100
BEAVER	B160
HEILBRONN	H416
ESSAY	E200

Koks kodas būtų sugeneruotas vardui HILBERT?

- A. H410
- B. B540
- C. H041
- D. H416

PAAIŠKINIMAS

Teisingas atsakymas: D.

1. Paliekama pirmoji žodžio raidė (H).
2. Hilbert → H4163 (pritaikius pirmąsias tris taisykles).
3. H416 (paliekami tik pirmieji keturi simboliai).

TAI INFORMATIKA

Algoritmas žinomas Soundex vardu.

Soundex algoritmas naudojamas vardų paieškai pritaikant tarimą anglų kalba. Algoritmas daug dėmesio skiria fonetinių klaidų korekcijai: klaidoms, atsirandančioms rašant vardų sekas, ypač kai yra panašiai tariamų žodžių. Algoritmą naudoja įvairios tarptautinės institucijos (pavyzdžiui, Interpolas), tačiau kai kurių šalių vardams netinka.

<https://en.wikipedia.org/wiki/Soundex>

Algoritmas išpopuliarėjo, kai jį aprašė Donaldas Knuthas savo knygoje „The Art Of Computer Programming, Vol. 3: Sorting And Searching“.

Raktiniai žodžiai: algoritmas, klaidos.

Informatinis mąstymas: algoritmavimas.

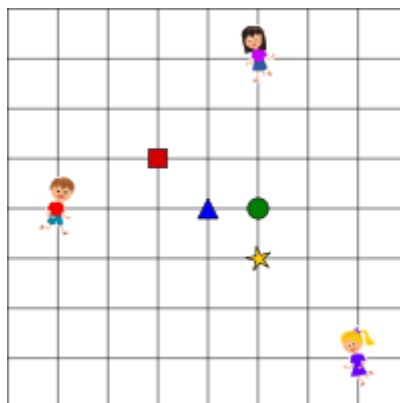
41. TRYS DRAUGAI

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
				12		



Benas, Alisa ir Jonė žaidžia skirtingose vietose. Jie nori žaisti kartu paskirdami susitikimo vietą kurioje nors iš sankirtų. Todėl išmatuoja atstumą langelių skaičiumi eidami stačiai ir gulsčiai nuo jų dabartinės padėties iki susitikimo vietos.

Pavyzdžiui, atstumas nuo Jonės iki susitikimo taško ▲ yra 6.



Kuris susitikimo taškas turi būti pasirinktas, kad visi draugai nueitų trumpiausią atstumą nuo jų dabartinės padėties? Pažymėk žemėlapyje teisingą figūrą.

PAAIŠKINIMAS



Neteisingas. Bendras atstumas nuo jų pradinės padėties iki raudono kvadrato yra $4 + 3 + 8 = 15$.



Neteisingas. Bendras atstumas nuo jų pradinės padėties iki mėlyno trikampio yra: $4 + 3 + 6 = 13$.



Teisingas. Bendras atstumas nuo jų pradinės padėties iki žalio skritulio yra: $3 + 4 + 5 = 12$.



Neteisingas. Bendras atstumas nuo jų pradinės padėties iki geltonos žvaigždutės yra: $4 + 5 + 4 = 13$.

TAI INFORMATIKA

Informatikos moksle lokali paieška gali būti naudojama, sprendžiant uždavinius, kai reikia optimizuoti sprendimą. Taikant lokalias paieškos algoritmus, lokaliai optimalus sprendinys randamas nagrinėjamo sprendinio aplinkoje.

Šiame uždavinyje geriausia susitikimo vieta randama sudėjus visus atstumus, kuriuos apriboja kiekvieno taško charakteristikos. Mažiausias iš sumų yra teisingas atsakymas. Pasitelkę šį metodą, naudojate lokaliąją paiešką.

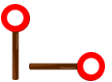
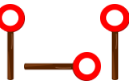
Raktiniai žodžiai: lokali paieška, algoritmas, optimizavimas.

Informatinis mąstymas: algoritmavimas, uždavinio skaidymas.

42. SURASK SKLANDYTUVĄ

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
					9	

Jonė ir Kajus žaidžia lauke su sklandytuvo modeliu. Vaikai susitarė, kad Jonė paleis sklandytuvą nuo kalvos, o Kajus paims jį, kai sklandytuvas nusileis ant žolės. Tačiau žolė nešienauta ir nutūpusį lėktuvą galima matyti tik nuo kalvos. Vaikai sugalvojo signalinių ženklų kodus, kuriais Jonė ant kalvos parodo, kur nusileido lėktuvas:

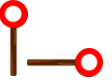
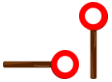
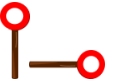
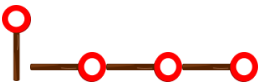
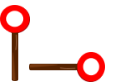
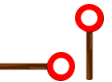
Kairėn	Dešinėn	Link kalvos	Tolyn nuo kalvos
			

Deja, šis kodas turi trūkumą: kai komandos siunčiamos be pauzės, galima skirtingai suprasti tą pačią komandų seką. Pavyzdžiui, jei rodomos šios komandos:



tai gali reikšti: Kairėn, Link kalvos, Kairėn arba: Kairėn, Dešinėn, Kairėn, Kairėn.

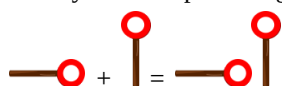
Vaikai pabandė patikslinti savo kodą. Tik vienas iš jų sugalvotų kodų leidžia vienareikšmiškai suprasti komandų seką. Kuris?

	Kairėn	Dešinėn	Link kalvos	Tolyn nuo kalvos
A.				
B.				
C.				
D.				

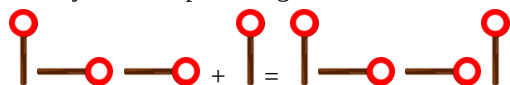
PAAIŠKINIMAS

Pirmiausia atmetami netinkami atsakymai.

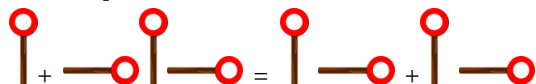
Atsakymas B nepateikia gero kodo, nes seka Kairėn, Dešinėn išreiškiama taip pat, kaip komanda Link kalvos:



Atsakymas D nepateikia gero kodo, nes seka Kairėn, Link kalvos išreiškiama taip pat, kaip komanda Tolyn nuo kalvos.



Sunkiau įžvelgti, kad atsakymas A nepateikia gero kodo. Čia komandų seka Kairėn, Link kalvos išreiškiame tokia pačia komandų seka, kaip ir Dešinėn, Dešinėn:



Kodėl atsakymas C yra teisingas? Kaip galima įsitikinti, kad nėra dviejų skirtingų signalų, išreiškiamų ta pačia seka? Pastebėkime, kad atsakymo C keturios komandos koduojamos skirtingu signalinio ženklo nuleidimų skaičiumi (nulis nuleidimų, vienas nuleidimas, du nuleidimai ir trys nuleidimai), einančių po vieno signalinio ženklo pakėlimo. Taigi kai tik Kajus pamato pakeltą signalinį ženklą, jis žino, kad pradedamas rodyti naujas kodas. Iš koduojant žinutę, jis turi suskaičiuoti nuleidimų skaičių tarp pakėlimų.

TAI INFORMATIKA

Informacija, siunčiama tarp kompiuterių laidiniu ar bevieliu tinklu, naudoja greitas signalų sekas. Kiekvienas signalas turi dvi būsenas (įjungta arba išjungta), kaip ir vaikai naudoja du signalus (pakeltas arba nuleistas ženklas). Tik kompiuteriuose šis procesas vyksta sudėtingiau.

Būdas, kuriuo pranešimai arba komandos pakeičiamos į signalus, vadinamas (dvejetainiu) kodu. Šiuo atveju yra kintamo ilgio kodas, nes vienam pranešimui ar komandai naudojamų signalų skaičius yra nevienodas.

Svarbu, kad koduoto pranešimo gavėjas galėtų išversti signalus į pradinį pranešimą be klaidų. Kitaip tariant, reikia būti atsargiam, kuriant tokį kodą. Tinkamai sukurti kodai turi būti vienareikšmiškai iš koduojami.

Vienas iš tokių kodų yra prefiksinis kodas, kuris ypatingas tuo, kad jame nėra tokio žodžio, kuris visas būtų kito to paties kodo žodžio pradžia. Pavyzdžiui:

Kairėn	Dešinėn	Link kalvos	Tolyn nuo kalvos
			

Prefiksiniai kodai gali būti labai trumpi ir lengvai iš koduojami. Jie naudojami ne tik komunikacijoje, bet ir glaudinimo algoritmuose (bei įvairiose Bebro užduotyse).

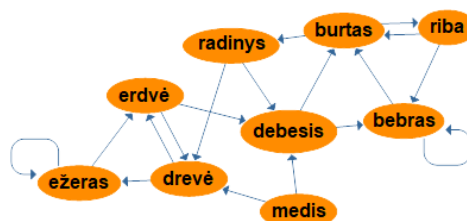
Raktiniai žodžiai: kodas, kintamo ilgio kodas, prefiksinis kodas.

Informatinis mąstymas: abstrahavimas, algoritmavimas.

43. ILGIAUSIA ŽODŽIŲ GRANDINĖ

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
					9	

Du bebrai žaidžia žodžių grandinės žaidimą. Vienas bebras pradeda pasakydamas žodį. Tada kiekvienas pačiam sako po dar nepanaudotą žodį, prasidedantį trečia ankstesnio žodžio raide. Deja, bebrai žino labai nedaug žodžių – visą jų žodyną galima pavaizduoti taip:



Kiek žodžių sudaro ilgiausia žodžių seka, pasakyta per vieną žaidimą?

PAAIŠKINIMAS

Bebrai gali panaudoti daugiausiai 8 žodžius. Galima žodžių seka:

medis→drevė→ežeras→erdvė→debasis→bebras→burtas→radinys

Yra ir kitų tokio pat ilgio sekų. Turime pavyzdį iš 8 žodžių, bet dar nežinome, ar įmanoma sudaryti 9 žodžių seką. Tokiu atveju mums reiktų panaudoti visus žodyno žodžius. Kadangi žodyne nėra žodžio, kurio trečia raidė m, turėtume pradėti nuo žodžio „medis“, todėl prieš žodžius radinys ir riba turi būti du skirtingi žodžiai, kurių trečia raidė „r“, bet tokį žodį turime tik vieną – burtas. Todėl neįmanoma viename žaidime panaudoti visų 9 žodžių.

TAI INFORMATIKA

Spręsdami šį uždavinį turime suprasti taisyklių rinkinį, aiškų ir patogų duomenų vaizdavimą (grafą) ir tada surasti optimalų sprendimą duotoje sistemoje (žodyne). Grafiškai dažnai naudojami informatikoje. Jais modeliuojamos įvairiausios struktūros iš objektų ir jų ryšių, pavyzdžiui, traukinių maršrutų arba vandentiekių išdėstymo schemos.

Turėdami daugiau informatikos žinių, galėtume pakeisti uždavinį, pavyzdžiui, ilgiausio maršruto paieška grafe. Tai glaudžiai susiję su žinomu Keliaujančio pirklio uždaviniu (https://lt.wikipedia.org/wiki/Keliaujan%C4%8Dio_pirklio_u%C5%BEdavinsys).

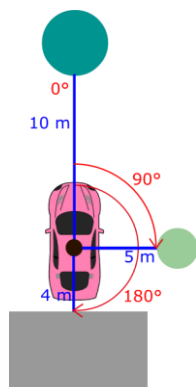
Informatikos mokslininkai teigia, kad surasti ilgiausią žaidimą didžiausiame žodyne (turint galvoje, kad žinome tūkstančius žodžių!) nėra įmanoma. Tai užimtų per daug laiko net su geriausiu algoritmu.

Raktiniai žodžiai: grafas, dedukcija, ilgiausias kelias, duomenų vaizdavimas.

Informatinis mąstymas: algoritmavimas, duomenų apdorojimas.

44. DINGĘS AUTOMOBILIS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
					9	



Nesėkmė! Autonominis automobilis pats negrįžo namo ir pasiliko kažkur mieste.

Prieš išsikraunant baterijai, autonominis automobilis rado parkavimo vietą ir nusiuntė namo kelią (bet ne visų) netoli esančių objektų, kuriuos atpažino jutikliai, padėtis. Kiekvieną objektą nusako dvi reikšmės:

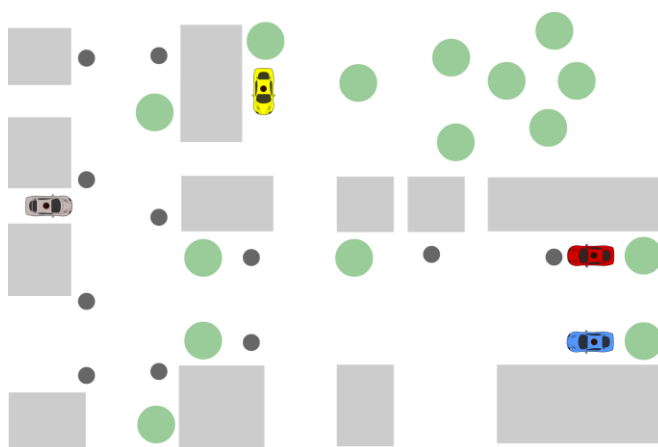
- kampas (susijęs su 360° jutikliu ant automobilio stogo; 0° kampas yra mašinos priekyje – žiūrėkite paveiksluką).
- objekto atstumas nuo automobilio jutiklio.

Pavyzdyje duotos reikšmės: [(0, 10), (90, 5), (180, 4)].

Dingęs automobilis atsiuntė tokias objektų reikšmes: [(0, 5), (90, 4), (180, 5), (270, 12)]

Raskite dingusį automobilį žemėlapyje!

Spustelėkite vieną iš automobilių, jei norite pažymėti jį dingusiu. Paspaudus jau pažymėtą automobilį (arba kitą automobilį) buvęs pažymėtas automobilis tampa nepažymėtu.



PAAIŠKINIMAS

Raudonas automobilis (žemėlapyje - dešinėsios pusės viduryje) yra dingęs automobilis.

Objektai yra 5 m priešais dingusį automobilį, 4 m nuo jo dešiniojo šono, 5 m už jo ir 12 m nuo jo kairiojo šono.

Pilkas automobilis nėra dingęs automobilis, nes objektai jo dešinėje ir kairėje nutolę tokiu pačiu atstumu nuo automobilio.

Geltonas automobilis nėra dingęs automobilis, nes objektas už jo (žalias apskritimas) žymiai arčiau nei objektas priešais jį (pilkas stačiakampis).

Mėlynas automobilis nėra dingęs automobilis, nes objektas priešais jį (žalias apskritimas) žymiai arčiau nei objektas už jo (mažas juodas apskritimas).

Raudonas automobilis yra dingęs automobilis: objektai priešais jį ir už jo tiek pat nutolę nuo jo, o objektas jo dešinėje (pilkas stačiakampis) žymiai arčiau nei jo kairėje pusėje (mėlynas automobilis).

TAI INFORMATIKA

Autonominiai automobiliai naudoja LIDAR (šviesos aptikimo ir atstumo) metodą skenuoti aplinką, pasitelkus lazerių technologijas. Jų navigacinė programa sukuria sudėtingą trimatį visų objektų, nutolusių per kelis šimtus metrų, modelį.

Šiame uždavinyje modelis labai paprastas ir aptinka tik artimiausius objektus. Paprastai modelis yra realybės abstrakcija ir apima tik tuos realybės aspektus, svarbius tam tikram tikslui. Autonominio vairavimo atveju, modelis gali būti naudojamas avarių prevencijai. Šiam tikslui reikalinga santykinė aplinkinių objektų vieta. Kiti aspektai, pavyzdžiui, spalva, nėra svarbūs ir gali būti ignoruojami.

<https://en.wikipedia.org/wiki/Lidar>

Raktiniai žodžiai: robotika, duomenų vaizdavimas, modelis.

Informatinis mąstymas: abstrahavimas, duomenų apdorojimas.

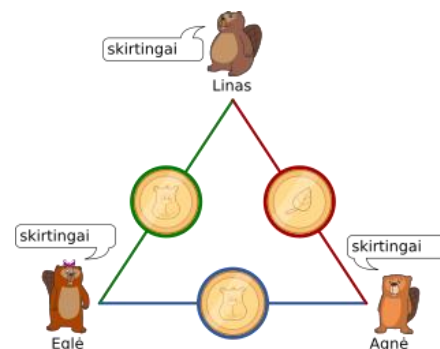
45. PASLAPČIŲ PASLAPTIS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
					12	

Linus, Agnė ir Eglė žaidžia loterijoje, kurioje yra ne daugiau nei vienas laimėtojas. Tačiau žaidėjas niekam negali sakyti, ar jis laimėjo, ar pralaimėjo. Žaidėjai:

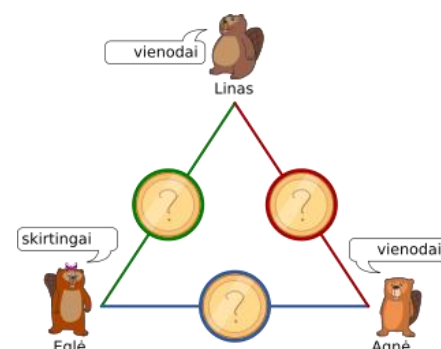
1. Linas ir Agnė slapta meta monetą.
2. Linas ir Eglė slapta meta monetą.
3. Agnė ir Eglė slapta meta monetą.
4. Kiekvienas žaidėjas pasako, ar jo mestos dvi monetos atvirto vienodai, ar skirtingai.
 - Pralaimėjusieji loterijoje sako tiesą.
 - Laimėjusieji loterijoje sako netiesą.

Pavyzdžiui, jei mestos monetos atvirto, kaip parodyta paveikslėlyje, Eglė laimi loterijoje, o kiekvienas žaidėjas pasako, kad atvirto „skirtingai“.



Jei Linas pasako, kad atvirto „vienodai“, Agnė pasako – „vienodai“, o Eglė pasako – „skirtingai“, tada kuris iš teiginių yra teisingas?

- A. Loterijoje nelaimėjo nei vienas žaidėjas.
- B. Kažkuris žaidėjas laimėjo loterijoje, bet nežinoma kas.
- C. Kažkuris žaidėjas laimėjo loterijoje ir tikrai žinoma kas.
- D. Nežinoma, ar kas nors laimėjo, ar pralaimėjo loterijoje.



PAAIŠKINIMAS

Teisingas atsakymas: B. Pateikiami keli sprendimai.

Ar laimėjimas verčia meluoti? Jei žaidėjas meluoja, jis laimėjo loterijoje. Laimėjimas loterijoje priverčia kažkurį žaidėją pameluoti. Tai leidžia manyti, kad tas, kuris meluoja, yra laimėtojas. Paprastai gyvenime laimėtojai tikriausiai nemeluoja. Įsivaizduokime, kad esam tokiam pasaulyje, kuriame galioja šio uždavinio taisyklės, ir laimėtojas pameluoja.

Mes turime melagį! Galima įsitikinti, kad kažkuris žaidėjas pagal šio uždavinio scenarijų (atvirto vienodai, vienodai, skirtingai) meluoja. Tai galima išsiaiškinti dviem būdais.

Pirmasis būdas: prieštaros metodas.

Žaidėjai pasakė, kad atvirto „vienodai, vienodai, skirtingai“. Tarkime, kad visi šie žaidėjai nemeluoja. Linas pasakė, kad atvirto „vienodai“, todėl LA ir LE metimų rezultatai turėtų būti vienodi. Kadangi Agnė pasakė, kad atvirto „vienodai“, tada LA ir AE metimų rezultatai turėtų būti vienodi. Todėl visų trijų metimų rezultatai turėtų būti vienodi. O tai prieštarauja teiginiui, kad „visi žaidėjai sakė tiesą“, nes užduotyje žaidėjai teigė, kad atvirto „vienodai, vienodai, skirtingai“.

Pastaba: tačiau tai nereiškia, kad Eglė laimėjo loterijoje, o tiesiog pasako, kad teiginys „visi sako tiesą“ neteisingas.

Antrasis būdas: sudaryti lentelę.

Galima sudaryti visų aštuonių galimybių lentelę. Pirmuose trijuose stulpeliuose surašomi visų metimų rezultatai. Kituose trijuose stulpeliuose surašomi visų žaidėjų pasisakymai, kai jie sako tiesą.

Metimai			Pasisakymai		
LA	LE	EA	L	A	E
Herbas	Herbas	Herbas	Vienodai	Vienodai	Vienodai
Herbas	Herbas	Moneta	Vienodai	skirtingai	Skirtingai
Herbas	Moneta	Herbas	Skirtingai	Vienodai	Skirtingai
Herbas	Moneta	Moneta	Skirtingai	Skirtingai	Vienodai
Moneta	Herbas	Herbas	Skirtingai	Skirtingai	Vienodai
Moneta	Herbas	Moneta	Skirtingai	Vienoda	Skirtingai
Moneta	Moneta	Herbas	Vienodai	Skirtingai	Skirtingai
Moneta	Moneta	Moneta	Vienodai	Vienodai	Vienodai

Pastebėsime, kad lentelėje nėra eilutės, kurioje būtų sakoma, kad monetos atvirto „vienodai, vienodai, skirtingai“. Taigi kažkuris žaidėjas meluoja. Todėl galima atmesti atsakymus A ir D.

Ar galima teigti, kad laimėtojas žinomas? Aišku, kad bent vienas žaidėjas melavo.

Jei žaidėjas L melavo, tada žaidėjų pranešimai turėjo būti toks: monetos atvirto „skirtingai, vienodai, skirtingai“.

Jei žaidėjas A melavo, žaidėjai turėjo pasakyti, kad monetos atvirto „vienodai, skirtingai, skirtingai“.

Jei žaidėjas E melavo, žaidėjai turėjo sakyti, kad monetos atvirto „vienodai, vienodai, vienodai“.

Daugiau informacijos, kuri padėtų nuspręsti, kuris iš šių teiginių teisingas, uždavinio sąlygoje nėra. Todėl negalima pasakyti, kuris žaidėjas meluoja. Žinoma, kad kažkuris žaidėjas laimėjo loterijoje, bet nežinoma, kas.

TAI INFORMATIKA

Lino, Agnės ir Eglės situacija žinoma Pietaujančių kriptografų problemos pavadinimu. Kriptografijoje Pietaujančių kriptografų problema analizuoja, kaip atlikti saugų loginės funkcijos ARBA daugianarį skaičiavimą. Pirmasis šią problemą pasiūlė David Chaum 1980 m. Jis panaudojo ją kaip pavyzdį, rodantį, kad galima siųsti anoniminius pranešimus su besąlygišku siuntėjo ir gavėjo atsekamumu. Anoniminiai komunikacijos tinklai, pagrįsti šia problema, dažnai vadinami DC tinklais (DC reiškia „Dining Cryptographers“).

Šiuo metu aktuali problema – internetas, kur asmens duomenys dažnai renkami be jo sutikimo.

Raktiniai žodžiai: kodas, kintamo ilgio kodas.

Informatinis mąstymas: loginis pagrindimas.

46. REPETICIJŲ PLANAVIMAS

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
					12	

Baletų mokykla planuoja balerinių šokio poromis pasirodymą. Yra 6 balerinos: Agota, Brigita, Cecilija, Dovilė, Evelina ir Fausta.

Jos šoks tokiomis poromis:

Agota - Brigita
 Evelina - Dovilė
 Agota - Evelina
 Brigita - Cecilija
 Dovilė - Agota
 Fausta - Brigita
 Cecilija - Evelina
 Brigita - Dovilė
 Dovilė - Fausta
 Fausta - Evelina

Baletų mokytojas nori suplanuoti popietės repeticijas taip, kad kiekviena pora galėtų pašokti tam tikrą laiką, o vėlesnėje repeticijoje porą sudarytų viena šokėja iš buvusios repeticijos ir viena nauja šokėja.

Pavyzdžiui, jeigu Agota ir Evelina yra pirmoji šokančių pora, tai kitoje repeticijoje turėtų dalyvauti Cecilija ir Evelina.

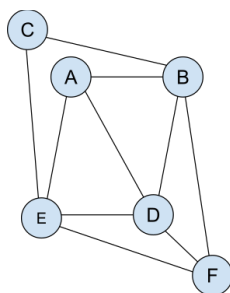
Kuri balerina negali būti pirmojoje poroje, kad ir kokią tvarkaraštį mokytojas suplanuotų?

A. Agota B. Brigita C. Cecilija D. Dovilė E. Evelina F. Fausta

PAAIŠKINIMAS

Teisingas atsakymas yra: C (Cecilija).

Apsvarstykite žemiau pateiktą diagramą.



Šioje diagramoje apskritimai vaizduoja balerinas. Du apskritimai sujungti, kai dvi balerinos šoka poroje. Repeticijų tvarkaraštis, sudarytas pagal taisykles, atitinka kelią per šią diagramą: pradėkite nuo apskritimo ir keliaukite linijomis nuo apskritimo prie apskritimo, kol kiekviena linija bus praėjusi lygiai vieną kartą. Eidami šiuo keliu turite išeiti iš kiekvieno apskritimo, į kurį įeinate, todėl kiekvienas apskritimas turėtų būti sujungtas lyginiu linijų skaičiumi, išskyrus pirmąjį ir paskutinįjį kelyje. Taigi, pirmasis apskritimas kelyje turėtų būti A arba F. Kitaip tariant, kad ir koks tvarkaraštis būtų pasirinktas, pirmojoje poroje turėtų būti arba Agota, arba Fausta. O Cecilija yra vienintelė balerina, kuri nešoka poroje nei su viena iš jų, todėl ji negali būti pirmojoje poroje.

TAI INFORMATIKA

Diagrama, naudojama paaiškinime, yra grafas. Kelias yra Oilerio maršrutas.

Grafas – būdas sukurti situacijos modelį. Jį sudaro viršūnės, šiuo atveju, viršūnės yra šokėjos, ir briaunos. Kiekviena briauna jungia dvi viršūnes. Šioje užduotyje briaunos vaizduoja poras. Spręsdami šią problemą panaudosite visas poras taip, kad kiekviena briauna būtų panaudota tik vieną kartą. Oileris įrodė, kad tai įmanoma tik tokiu atveju, jei pradėsite ir užbaigsite tokioje viršūnėje, kurią jungia nelyginis briaunų skaičius.

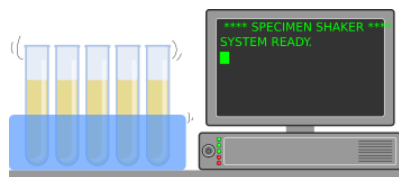
Informacijos pavaizdavimas naudojant grafą gali padėti suprasti problemos struktūrą. Kiti pritaikymo būdai gali būti, pavyzdžiui, maršrutų planavimas ar tvarkaraščio planavimas.

Raktiniai žodžiai: grafas, Oilerio maršrutas.

Informatinis mąstymas: algoritmavimas, uždavinio skaidymas.

47. MEDICINOS LABORATORIJA

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
					12	



Diagnostikos aparatas medicinos laboratorijoje turi reguliariai supurtyti iš pacientų paimtus mėginius. Aparatas veikia pagal programą, pateiktą sunumeruotomis eilutėmis. Aparatas skaito programos eilutes iš eilės ir vykdo jose parašytas komandas. Jei eilutėje yra komanda eiti į X, tai pereinama į eilutę X ir nuo jos tęsiama programos vykdymas.

Programa turi saugyklą A, kurioje telpa vienas skaičius. Programa gali įrašyti skaičių į A, pridėti 1 prie ten esančio skaičiaus, joje esantį skaičių palyginti su kuriuo nors kitu skaičiumi.

Kiek kartų aparatas supurtys mėginius, vykdydamas tokią programą?

- 1 į A įrašyti 0
- 2 prie A pridėti 1
- 3 eiti į 6
- 4 jei A = 60, eiti į 8
- 5 į A įrašyti 0
- 6 prie A pridėti 1
- 7 eiti į 2
- 8 mėginius supurtyti A kartų
- 9 baigti darbą.

Atsakymai

- A. Mėginiai niekada nebus supurtyti.
- B. Mėginiai bus supurtyti vieną kartą.
- C. Mėginiai bus supurtyti 60 kartų.
- D. Programa veiks be paliovos, ir aparatas nuolat purtys mėginius.

PAAIŠKINIMAS

Programa visada pereis nuo 3-os eilutės prie 6-os, nuo 7-os prie 2-os ir taip nuolat vykdys komandas, esančias 2-oje, 3-oje, 6-oje ir 7-oje eilutėse. Mėginių supurtymo komanda yra 8-oje eilutėje, kuri niekada nebus pasiekta. Tai reiškia, kad aparatas, veikdamas pagal šią programą, mėginių niekada nepurtys. Todėl teisingas atsakymas yra A. Be to, niekada nebus įvykdyta ir komanda, esanti 9-oje eilutėje. Taigi, programos vykdymas niekada nesibaigs.

TAI INFORMATIKA

Programos, parašytos pirmosiomis programavimo kalbomis, kurios sukurtos praėjusio šimtmečio 5-ajame ir 6-ajame dešimtmečiuose ir vadinamomis assemblerio kalbomis, buvo panašios į pateiktą šiame uždavinyje. Jose vienintelė komanda kitų komandų vykdymo eilei keisti buvo nukreipimo komanda, trumpiau – eiti į (angl. *goto*). Tokias programas buvo sunku skaityti ir ieškoti jose klaidų. Dėl to nuo 6-ojo dešimtmečio į modernesnes programavimo kalbas įtraukiamos specialios konstrukcijos – ciklai, sąlyginiai sakiniai, procedūros, įgalinančios atsisakyti pavojingų nukreipimo komandų.

Šiame uždavinyje parodoma ir dažnai pasitaikanti programavimo klaida – begalinis ciklas, kai programa „užsicsiklina“, t. y. negali užbaigti ciklo.

<https://homepages.cwi.nl/~storm/teaching/reader/Dijkstra68.pdf>

Raktiniai žodžiai: programavimo kalba, „goto“ komanda, programos skaitymas.

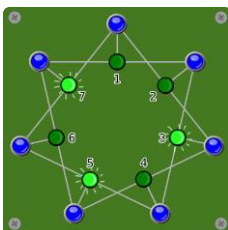
Informatinis mąstymas: algoritmavimas.

48. ĮJUNKITE ŠVIESAS!

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
					12	

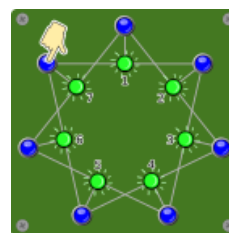
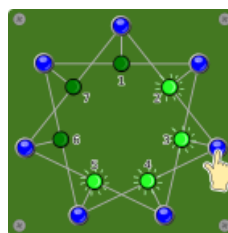
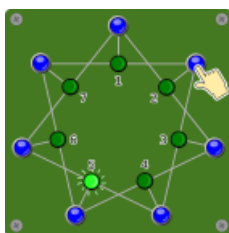
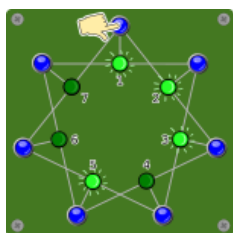
Kambaryje yra jungikliai, kuriuos spaudžiant įjungiamos lemputės. Paspaudus jungiklį, trys lemputės, su kuriomis tas jungiklis sujungtas, pakeis būsenas: jei lemputės buvo įjungtos, jos išjungiamos, jei buvo išjungtos – ims šviesti.

Įjunkite visas lemputes! Spustelėdami po vieną jungiklį, įjunkite visas lemputes.



PAAIŠKINIMAS

Teisingas sprendimas: jungikliai, esantys šalia lempučių, kurių numeriai 1, 2, 3, ir 7, gali būti spustelėti bet kuria tvarka, norint įjungti visas lemputes. Galima pradėti nuo galutinės tikslo būsenos (tikslas) ir ieškoti sprendinio, einant nuo tikslo pradinės būsenos link. Spustelėjus paskutinį jungiklį, jis įjungs tris lemputes tik tokiu atveju, jei trys lemputės prieš tai buvo išjungtos. Taigi pirmiausia mums reikia išjungti keletą lempučių. Spustelime jungiklius, esančius arti 1-os lemputės, o tada – esančius šalia 2-os ir 7-os lempučių, o paskui šalia 1-os, 2-os ir 3-os lempučių. Galiausiai lieka įjungta tik 5-a lemputė. Dabar galima spustelėti jungiklius, esančius šalia 3-os ir 7-os lemputės, ir bus uždegtos visos lemputės.



Ieškant sprendinio gali pagelbėti šie pastebėjimai:

- Neverta du kartus (ar lyginį skaičių kartų...) spustelėti jungiklį, nes antrasis spustelėjimas panaikina pirmojo spustelėjimo rezultatą;
- Jei spustelime jungiklį X ir po to jungiklį Y, rezultatas bus tas pats, kai spustelėtume pirma Y, o paskui X.

Todėl galimų jungiklių spustelėjimų sekų skaičius ribotas: blogiausiu atveju visi septynių jungiklių poaibiai. Tokie poaibiai yra 128, jei iš viso yra 7 elementai aibėje. Panaudojus kiekvieną poaibį, gausime skirtingą rezultatą. Kadangi yra tik 27 (= 128) skirtingos tinklo konfigūracijos, kiekviena jų gali būti gaunama iš pradinės konfigūracijos.

TAI INFORMATIKA

Daugelyje uždavinių žinome uždavinio tikslą ir pradinę būseną. Šio uždavinio tikslas – įjungti visas lemputes, o pradinė būseną: įjungtos 3-a, 5-a ir 7-a lemputės, o likusios lemputės išjungtos.

Kol dirbtinis intelektas dar buvo neišplėtotas, problemų sprendimas, naudojant kompiuterines sistemas, dažnai buvo suprantamas kaip veiksmų seka, vedanti nuo pradinės būsenos prie tikslo. Tokia samprata turi prasmę, jei yra baigtinė galimų veiksmų seka ir baigtinė aibė objektų, kurių būseną galima keisti, atliekant nurodytus veiksmus. Buvo sukurta daug įvairių tokių paiešką realizuojančių strategijų. Pavyzdžiui, „daugelio pabaigos būsenų analizė“ pasirenka veiksmą, kuris sumažina skirtumą tarp dabartinės ir galutinės būsenos. O šiame uždavinyje pravartu taip pat ieškoti veiksmų, kurie pakeistų kažkurią kitą būseną galutine būseną. Jei paieška atliekama tiek iš pradinės, tiek iš galutinės būsenos, tokia paieška vadinama „dvikrypte paieška“.

https://en.wikipedia.org/wiki/Bidirectional_search

Raktiniai žodžiai: kompiuterinė sistema, logika, grandinės, paieška.

Informatinis mąstymas: algoritmavimas, loginis pagrindimas.

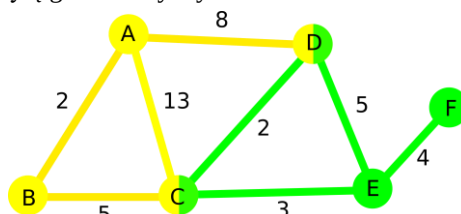
49. TELEGRAFO TINKLAI

Nykštukai	Mažyliai	Bičiuliai	Draugai	Jauniai	Kolegos	
					12	

Paveiksle pavaizduotas tarptautinis duomenų tinklas. Kiekvieną tinklą sudaro mazgai, nuspelvinti ta pačia spalva. Dvispalviai mazgai vaizduoja skirtingų tinklų ryšius. Tarptinkliniam ryšiui taikomas mokestis. Jei pranešimas siunčiamas iš vieno tinklo mazgo į kitą tinklą, siuntimo kaina yra:

(kelio, kurį pranešimas turi nukelti pirmame tinkle, kaina) + (kelio iki gavėjo visuose kituose tinkluose kaina) × 2.

Kiekviena tinklų šaka turi savo kainą, parodytą grafe. Pavyzdys:

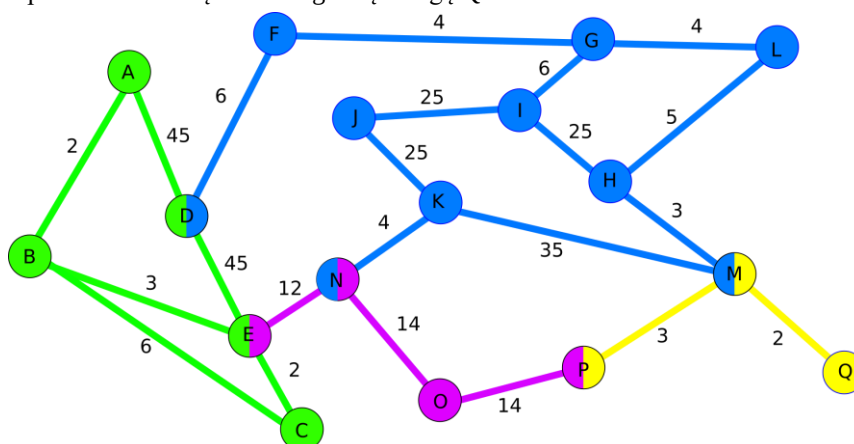


Pigiausia kaina informacijai iš mazgo A į mazgą F perduoti yra 21:

A→B	B→C	C→E	E→F
2	5	3	4

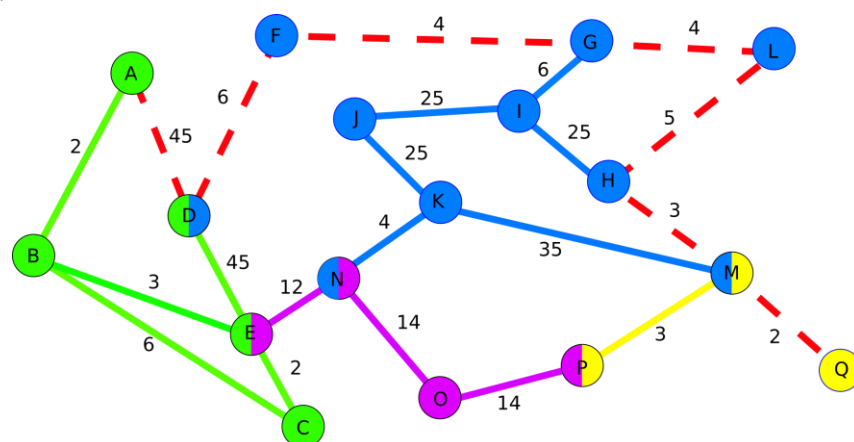
$$2+5+(3+4)*2=21$$

Kokia yra mažiausia kaina pranešimui nusiųsti iš mazgo A į mazgą Q?



PAAIŠKINIMAS

Teisingas atsakymas: 93.



Atkreipsime dėmesį, kad pigiausias kelias ne visada yra trumpiausias. Kad pranešimas iš žaliojo tinklo nukeltų iki violetinio tinklo, mažiausia kaina 5. Tačiau iš mazgo E keliaujant mazgais N, K, M į mazgą Q arba mazgais N, O, P, M į mazgą Q, kaina didesnė, negu keliaujant mazgais D, F, G, L, H, M į mazgą Q.

Tada:

- Kaina kelio A, D, F, G, L, H, M, Q = $45 + (6 + 4 + 4 + 5 + 3 + 2) \times 2 = 93$
- Kaina kelio A, B, E, N, K, M, Q = $(2 + 3) + (12 + 4 + 35 + 2) \times 2 = 111$
- Kaina kelio A, B, E, N, O, P, M, Q = $(2 + 3) + (12 + 14 + 14 + 3 + 2) \times 2 = 95$

TAI INFORMATIKA

Šis uždavinys skirtas rasti mažiausios kainos kelią grafe iš vieno mazgo į kitus mazgus, kreipiant dėmesį į pograpių susiejimo taškus. Kelio grafe radimas – vienas svarbių informatikos uždavinių, taikomas daugelyje sričių. Pigiausiam keliui rasti galima naudoti, pavyzdžiui, Dijkstros algoritmą. Dijkstros algoritmas taikomas daugelyje informatikos sričių trumpiausiam keliui rasti, pavyzdžiui, kompiuterių tinklų srityje (maršrutų parinkimo sistemose). Šis algoritmas taip pat taikomas „Google“ žemėlapiuose trumpiausiam maršrutui rasti iš vienos geografinės vietos į kitą. Biologijoje šis algoritmas taikomas konstruojant infekcinių ligų plitimo tinklo modelį. Deja, dėl šiame uždavinyje pasirinkto būdo kainoms skaičiuoti, Dijkstros algoritmas negali būti pritaikytas tiesiogiai.

https://en.wikipedia.org/wiki/Dijkstra%27s_algorithm

Raktiniai žodžiai: pigiausio kelio paieška, grafai, pografiai, grafo apėjimas.

Informatinis mąstymas: algoritmavimas, uždavinio skaidymas.

Airija, Australija, Austrija,
Azerbaidžanas, Baltarusija,
Belgija, Bosnija ir Hercegovina,
Brazīlija, Bulgārija, Čekija, D. Britānija,
Egiptas, Estija, Grieķija, Honkongas,
Indonēzija, Irānas, Islandija, Ispanija,
Itālija, Izraēlis, Japānija, JAV, Jordānija,
Kanāda, Kazachstānas, Krievija, Kipras,
Krievija, Latvija, Lietuva,
Makedonija, Malaizija, Malta, Mongolija,
Naujoji Zēlandija, Nigērija, Nyderlandi,
Pakistānas, Pietu Āfrika, Pietu Korēja,
Portugālija, Prancūzija, Rumunija, Rusija,
Serbija, Singapūras, Slovākija, Slovēnija,
Suomija, Švedija, Šveicarija, Taīlandas,
Taivānas, Tunisas, Turķija, Ukraina,
Vengrija, Vietnāmas, Vokietija.

Šalys, kuriose
vyksta konkursas