



**Vilniaus
universitetas**

Informatikos ir informatinio mąstymo mokomoji veikla

Klaidos informatikoje: jų aptikimas ir taisymas

Mokyklos pedagogika



Kuriame
Lietuvos ateitį
2014–2020 metų
Europos Sąjungos
fondų investicijų
veiksmų programa



**Vilnius
universitetas**

Informatikos ir informatinio mąstymo mokomosios veiklos sukurtos įgyvendinant projektą „Aukštųjų mokyklų tinklo optimizavimas ir studijų kokybės gerinimas Šiaulių universitetą prijungiant prie Vilniaus universiteto“ (Nr. 09.3.1-ESFA-V-738-03-0001), finansuojamą iš Europos socialinio fondo lėšų pagal 2014–2020 metų Europos Sąjungos fondų investicijų veiksmų programos 9 prioriteto „Visuomenės švietimas ir žmogiškųjų išteklių potencialo didinimas“ įgyvendinimo priemonę Nr. 09.3.1-ESFA-V-738 „Aukštųjų mokyklų tinklo tobulinimas“.

Metodinė medžiaga „Klaidos informatikoje: jų aptikimas ir taisymas“ skirta informatikos ir informatinio mąstymo ugdymui. Tikslinė grupė – būsimi informatikos pagrindinio ugdymo mokytojai. Medžiaga siejasi su informatikos ir matematikos Bendrosiomis programomis, algoritmų ir programavimo bei technologinių sprendimų pasiekimų sritimi. Atlikdami veikloje numatytas užduotis būsimi mokytojai išsiaiškins, kaip veikia kompiuteris, kai ieško ir taiso klaidas. Pateikiamas teorinis temos pagrindimas mokytojui, aptariamos pagrindinės srities sąvokos.

Veikloje naudojamos *CS Unplugged* (csunplugged.org) užduotys pagal *Creative Common* licenciją.

Veiklą parengė doc. dr. Eglė Jasutė ir prof. Valentina Dagiienė

Redagavo Viktoras Dagys

Vilnius, 2022

Įvadas

Imkime paprastą pavyzdį. Tarkime, reikia įnešti 10 € į banko sąskaitą. Banko kasininkas įveda sumą ir siunčia duomenis į centrinį kompiuterį. Tačiau atsiranda trikdžių ir į centrinį kompiuterį vietoj kodo 10 € nusiunčiamas kodas 1000 €. Klientui tai būtų džiaugsmas, tačiau bankui – problema.

Tai tik vienas paprastas pavyzdys. Bet iš to matome, kaip svarbu tikrinti siunčiamus duomenis ir aptikti klaidas. Priimantis duomenis kompiuteris turi patikrinti, ar siunčiami duomenys nebuvo sugadinti elektros ar kitokių trikdžių. Kartais klaidingai perduodamų duomenų siuntimas pakartojamas, tačiau kai kada tai neįmanoma, pavyzdžiui, kai diską sugadina magnetinė ar elektrinė spinduliuotė, karštis ar diskas pažeistas fiziškai. Kitas pavyzdys: kai duomenys gaunami iš kosmoso, norint pakartoti jų siuntimą, tektų labai ilgai laukti retransliacijos. (Signalas nuo Jupiterio iki Žemės eina pusę valandos, o juk ši planeta yra arčiausiai Žemės.)

Reikia mokytis nustatyti klaidingus duomenis (*klaidų aptikimas*), mokėti atpažinti sugadintus duomenis (*klaidos radimas*) ir atkurti originalius duomenis (*klaidos ištaisymas*).

Pristatysime ir aptarsime žaidimą su kortelėmis, vadinamą „Magišku triuku“, beje, panašus būdas naudojamas ir kompiuteriuose, ypač ieškant klaidų dvejetainiuose duomenyse. Pagrindinė idėja tokia: bitai surašomi į įsivaizduojamus stulpelius ir eilutes, į kiekvieną eilutę ir stulpelį prirašomi papildomi lyginumo bitai. Tokiu būdu galima ne tik nustatyti klaidą, bet ir pasakyti, kur ji yra. Pažeistas bitas atkuriamas. Klaida ištaisoma.

Tačiau kompiuteriuose naudojamos daug sudėtingesnės klaidų kontrolės sistemos, kurios gali nustatyti ir ištaisyti komplikuotas klaidas. Dalis kompiuterio standžiojo disko yra skirta klaidoms taisyti, todėl diskas gali patikimai veikti net tada, jei dalis jo yra pažeista. Klaidų kontrolė duomenyse grindžiama lyginumo principu. Kas tai, pasiaiškinsime atlikdami tolesnes veiklas.

Klaidos radimas

Siunčiamas pranešimas gali būti sugadinamas dėl įvairių priežasčių, pavyzdžiui, esant triukšmui duomenys gali būti perduodami netiksliai, pažeidžiami. Kad to išvengtume, naudojame klaidų aptikimo kodus, t. y., papildomus skaitmeninio pranešimo duomenų bitus, kurie padeda nustatyti, ar perdavimo metu įvyko klaida.

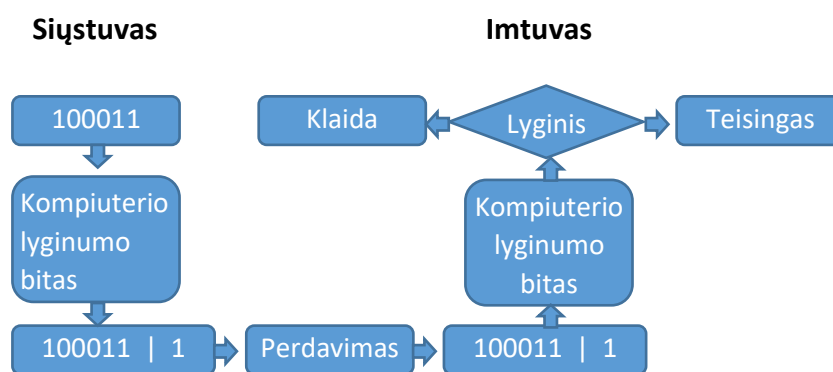
Paprastas lyginumo tikrinimas

Perduodamas vienas papildomas bitas, kad dvejetainio kodo vienetų skaičius būtų lyginis, jei lyginumas yra lyginis, arba nelyginis, jei lyginumas yra nelyginis.

Siuntėjo programa skaičiuoja, kiek yra vienetų. Jei naudojamas lyginis lyginumas ir vienetų yra lyginis skaičius, pridedamas vienas nulinis (0) bitas. Tokiu būdu vienetų skaičius išlieka lyginis. Jei vienetų skaičius nelyginis, pridedamas vieneto (1) bitas, kad vienetų skaičius būtų lyginis.

Gaunantysis prietaisas (imtuvai) skaičiuoja, kiek vienetų yra perduodama. Jei vienetų skaičius yra lyginis ir naudojamas lyginis lyginumas, perduoti duomenys laikomi nepažeistais ir pranešimas patvirtinamas. Net jei vienetų skaičius yra nelyginis ir naudojamas nelyginis lyginumas, pranešimas nėra sugadintas.

Imtuvai gali nustatyti vieno bito pakeitimą perdavimo metu (vieną klaidą), suskaičiuodamas vienetus. Tačiau kai neteisingų duomenų yra daugiau nei vienas bitas, imtuvui kur kas sunkiau nustatyti klaidą.



1 pav. Lyginio lyginumo tikrinimo pavyzdys

Pasiaiškinkite ir aptarkite. Išnagrinėkite 1 pav. schemą, sekite, kaip atliekami veiksmai. Parinkite kitokių pavyzdžių ir remdamiesi schema atlikite veiksmus.

Dvimatis lyginumo tikrinimas

Kiekvienai eilutei apskaičiuojami lyginumo tikrinimo bitai. Tai atliekama analogiškai, kaip prieš tai darėme. Analogiškai kiekvienam stulpeliui apskaičiuojami ir kartu su duomenimis perduodami lyginumo tikrinimo bitai. Jie lyginami su priimančiojoje pusėje gautų duomenų lyginumo bitais.

Siunčiami duomenys

10011001 11011001 10111011 00011001

Eilučių lyginumas

10011001		0
11011001		1
10111011		0
00011001		1
<hr/>		
11100010		0

Stulpelių lyginumas

100110010 110110011 101110110 000110011 111000100

Išsiųsti duomenys

2 pav. Dvimačio lyginumo papildomų bitų priskyrimo pavyzdys

Kontrolinė suma

Naudojant kontrolinės sumos klaidų aptikimo metodą, duomenys padalijami į k segmentų po m bitų.

Norint gauti bendrą sumą, segmentai sumuojami siuntėjo pusėje naudojant 1 papildinio aritmetiką. Norint gauti kontrolinę sumą, imamas sumos papildinys. Kontrolinės sumos segmentas siunčiamas kartu su duomenų segmentais. Kad būtų gauta bendra suma, visi gauti segmentai imtuvo pusėje sumuojami naudojant 1 papildinio aritmetiką. Tada apskaičiuojama suma. Jei rezultatas lygus 0, duomenys priimami; priešingu atveju jie atmetami.

Siunčiami duomenys

11001100 10101000 01001001 00111000

1 2 3 4

$k = 4, m = 8$

	Siųstuvas		Imtuvas
1	11001100	1	11001100
2	+ 10101000	2	+ 10101000
	<hr/>		<hr/>
	+ 101110100		+ 101110100
	1		1
	<hr/>		<hr/>
3	+ 01110101	3	+ 01110101
	+ 01001001		+ 01001001
	<hr/>		<hr/>
4	+ 10111110	4	+ 10111110
	+ 00111000		+ 00111000
	<hr/>		<hr/>
Suma	11110110		+ 11110110
			+ 00001001
Kontrolinė suma	00001001	Suma	11111111
		Rezultatas	00000000

Išvada: Duomenis priimti

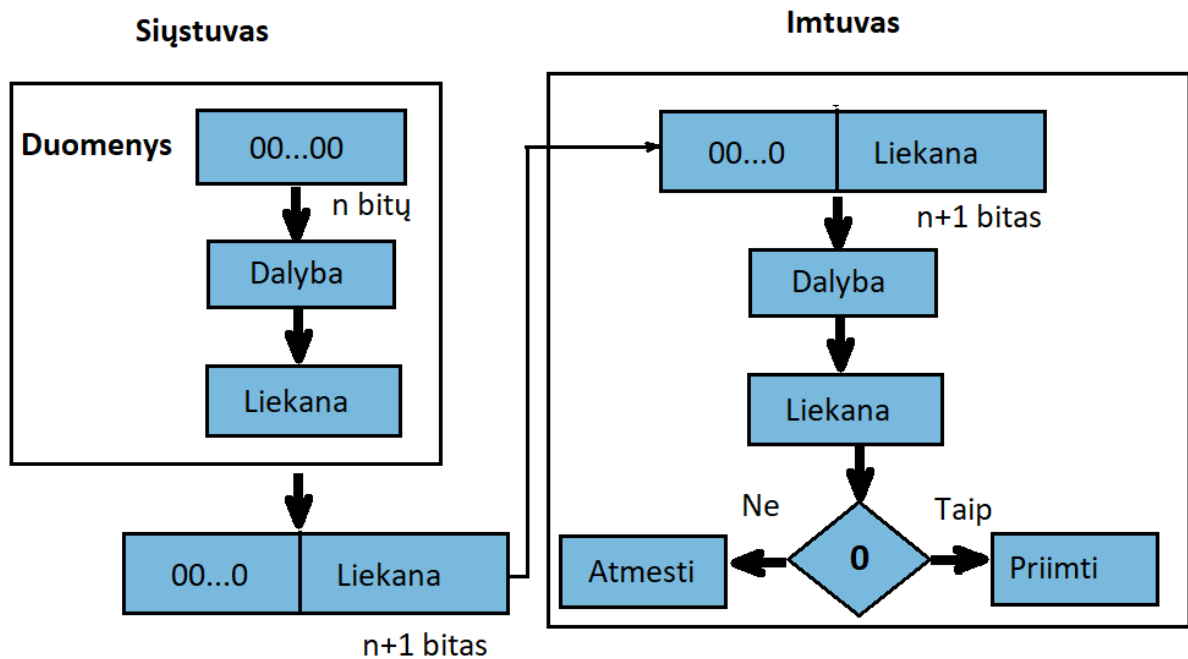
3 pav. Kontrolinės sumos naudojimo pavyzdys

Ciklinis perteklinis tikrinimas

Ciklinis perteklinis tikrinimas (angl. *cyclic redundancy check*, CRC) yra alternatyvus metodas, leidžiantis nustatyti, ar gautame cikle yra teisingų duomenų. Taikant šį metodą naudojamas perduodamų duomenų bitų dvejetainis dalijimas. Dalikliui generuoti naudojami polinamai.

Siuntėjas dalija perduodamus bitus ir apskaičiuoja liekaną. Prieš siųsdamas tikruosius bitus, siuntėjas įterpia liekaną pradinių bitų pabaigoje. Išsiųstus duomenis sudaro tikrieji duomenų bitai ir likutis.

Imtuvas gautus duomenis padalija naudodamas tą patį daliklį. Jei liekana sudaro vien nuliai, duomenų bitai yra patvirtinti; priešingu atveju manoma, kad perduodant duomenys buvo pažeisti.



4 pav. Ciklinis perteklinis tikrinimas

Kortelių keitimo magija

Saugant duomenis laikmenoje ar siunčiant iš vieno kompiuterio į kitą įprasta manyti, kad jie nesikeičia. Tačiau kartais įvyksta kas nors nenumatyta ir duomenys pasikeičia. Šioje veikloje studentams magišku triuku parodoma, kaip nustatyti, kada duomenys yra sugadinti, ir kaip tai ištaisyti. Žaidimo metu studentai išsiaiškina, kaip tikrinti gautus duomenis naudojant lyginimo tikrinimo metodą.

Ryšiai su ugdymo programomis

Matematika: skaičiavimas ir įvertinimas, koordinatės

Informatika: technologinės problemos, algoritmai

Priemonės

Kiekvienai studentų porai duodamos 36 kortelės (geriausia magnetinės), kurių viena pusė yra vienoda.

Vaizdo įrašas

Pateiktuose vaizdo įrašuose parodoma, kaip atliekamas šis magiškas triukas, kuris daugumai auditorijos narių atrodo intriguojantis. Šį triuką demonstruojantis sugeba „stebuklingai“ nustatyti, kuri iš kelių dešimčių kortelių buvo apversta, naudodamasis tais pačiais metodais, kuriuos naudoja kompiuteriai, nustatydami, ar duomenų saugykloje įvyko klaida.

https://www.youtube.com/watch?v=-35-sPTYPJg&t=51s&ab_channel=UCComputerScienceEducation

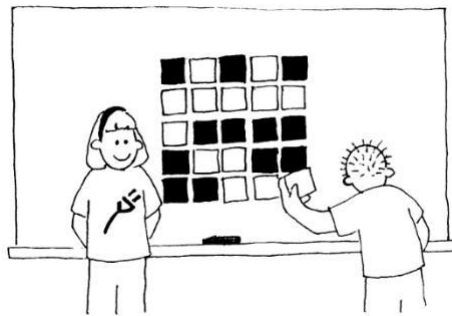
https://www.youtube.com/watch?v=vogghyZbZxo&t=322s&ab_channel=UCComputerScienceEducation

Magiškas triukas

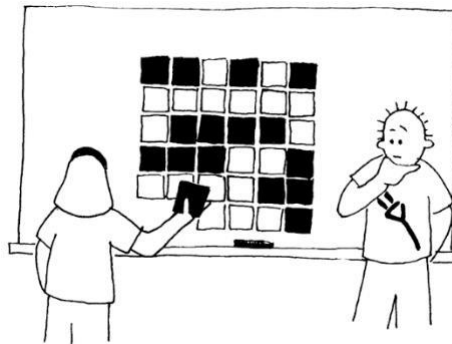
Demonstracija

Reikia 36 vienodų kortelių, kurių abi pusės yra skirtingų spalvų. (Galima iškirpti iš spalvoto popieriaus lapo, kurio viena pusė yra spalvota, o kita – balta, galima naudoti bet kokias žaidybines kortas, atminties žaidimo korteles ir pan. – svarbu, kad visų kortelių viena pusė būtų vienoda.) Demonstracijai patogiu naudoti korteles su magnetais iš abiejų pusių, tik reikia, kad abi jų pusės dar būtų ir skirtingų spalvų. (Dažniausiai tokios kortelės turi magnetus iš vienos pusės. Dvi korteles galima suklijuoti ta puse, kurioje nėra magneto, tada ant vienos iš „magnetinių“ naujos kortelės pusių nupiešti baltą tašką ar tą pusę nuspalvinti.)

1. Studento paprašoma išdėlioti korteles kvadratu 5×5 (kuri kortelės pusė bus viršuje nesvarbu).



Dėstytojas išdėlioja dar vieną eilutę ir stulpelį, kad šiek tiek pasunkintų užduotį.



Papildomos kortelės ir yra šio uždavinio – triuko – pagrindas. Papildomas korteles reikia sudėti taip, kad kiekvienoje eilutėje ir stulpelyje būtų lyginis nuspalvintų kortelių skaičius.

2. Tegul studentas vieną kortelę apverčia, kai dėstytojas nežiūri. Apverstos kortelės eilutėje ir stulpelyje bus nelyginis nuspalvintų kortelių skaičius. Taip nesunkiai nustatoma, kuri kortelė buvo apversta.

3. Padiskutuokite su studentais, kaip atliekamas šis triukas. Paašškinkite teoriją, kuria remiasi šis triukas.

Klaidos aptikimo metodas

1. Studentai susiskirsto poromis ir išdėlioja savo korteles ant stalo kvadratu 5×5.
2. Atkreipiamas dėmesys, po kiek nuspalvintų kortelių yra kiekvienoje eilutėje ir stulpelyje. Ar jų skaičius yra lyginis, ar nelyginis? Primenama, kad nulį šiame žaidime laikome lyginiu skaičiumi.

3. Pridedama po vieną papildomą kortelę prie kiekvienos eilutės ir įsitikinama, kad kiekvienoje eilutėje nuspalvintų kortelių yra nelyginis skaičius. Papildoma kortelė vadinama lyginumo kortele.
4. Taip pat pridedama dar viena kortelių eilutė, kad kiekviename stulpelyje būtų po nelyginį skaičių spalvotų kortelių.
5. Dabar viena kortelė apverčiama. Ką galima pastebėti apverstos kortelės stulpelyje ir eilutėje? (Nuspalvintų kortelių skaičius – nelyginis.) Lyginumo kortelė parodo, kur buvo pakeista.
6. Kortelės kaitaliojamos toliau – šis triukas atliekamas iš naujo.

Papildomos veiklos

Šį triuką galima išbandyti ir su kitais objektais. Pavyzdžiui, su loginio žaidimo kortelėmis (balta ir juoda), monetomis (herbas ir skaičius) arba kortelėmis su 0 ir 1 skirtingose pusėse (siejama su dvejetainė sistema).

Galima išbandyti triuką su daugiau kortelių, pavyzdžiui, išdėstyti iš jų didelį kvadratą 9×9 . (Šis triukas gali būti daromas su bet kiek kortelių, be to, jų nebūtina išdėstyti kvadratu.)

Diskusija su studentais

- Kas nutiktų, jei būtų apverstos dvi ar daugiau kortelių? (Negalima visada tiksliai pasakyti, kurios dvi kortelės buvo apverstos, tačiau visada galima pasakyti, kad kai kas buvo pakeista. Galima aptarti, kas įvyksta, jei apverčiamos viena ar daugiau kortelių porų. Įmanoma, kad apvertus 4 korteles lyginumo kortelės neparodys klaidos.)
- Kitas įdomus pratimas – nagrinėti apatinę dešiniąją kortelę. Jei ši lyginimo kortelė bus teisinga stulpeliui virš jos, tai ar ši kortelė bus teisinga ir eilutei iš kairės? (Taip, visada, jei naudojamas nelyginis lyginumas.)
- Šiame pratime lyginumas yra nelyginis – nelyginis nuspalvintų kortelių skaičius eilutėse ir stulpeliuose. Ar galima tą patį padaryti su lyginiu lyginumu? (Tai įmanoma. Tačiau apatinės dešinėsios lyginumo kortelės pratimas tinka tik tada, kai yra lyginis (arba nelyginis) ir eilučių, ir stulpelių skaičius išdėstyname plote. Pavyzdžiui, korteles išdėsčius į 5×9 arba 4×6 stačiakampį pratimą galima atlikti, o išdėsčius 4×7 pratimas netinka.)

Brūkšniniai kodai

Remiantis panašia metodika tikrinami knygų ir kitų prekių brūkšniniai kodai. Apžiūrėkite ant užpakalinio knygos viršelio išspausdintą 10 arba 13 skaitmenų kodą. Paskutinis skaitmuo yra kontrolinis skaitmuo, panašiai kaip anksčiau žaidžiant lyginumo kortelė.

Knygos ISBN numerio tikrinimas

Visoms knygoms suteiktas tarptautinis standartinis knygos numeris ISBN (angl. *International Standard Book Number*), tad galime patikrinti, ar ženklinant (numeruojant) knygą nepadaryta klaida.

Paprastai tikrinama kontrolinė suma.

Toliau su studentais aptariama, kaip tikrinama kontrolinė 10 skaitmenų suma. Pirmasis skaitmuo dauginamas iš 10, antrasis – iš 9, trečiasis – iš 8 ir t. t., devintasis dauginamas iš 2. Visos gautos sandaugos sudedamos. Pavyzdžiui, kodo ISBN 0–13–911991–4 suma yra

$$(0 \times 10) + (1 \times 9) + (3 \times 8) + (9 \times 7) + (1 \times 6) + (1 \times 5) + (9 \times 4) + (9 \times 3) + (1 \times 2) = 172$$

Gautas skaičius padalijamas iš 11. Kokia liekana?

$$172 \div 11 = 15 \text{ liekana } 7$$

Jei liekana yra 0, tai kontrolinė suma yra 0. Kitu atveju iš 11 atimama liekana ir gaunama kontrolinė suma

$$11 - 7 = 4$$

Ar skaitmuo 4 yra ISBN knygos kodo paskutinis skaitmuo? Taip.

Jei paskutinis skaitmuo būtų ne 4, būtų aišku, kad numeruojant padaryta klaida.

Galima apskaičiuoti 10 skaitmenų kodo kontrolinę sumą, kuriai reikia daugiau kaip vieno skaitmens (iš 11 atėmus liekaną gaunama 10). Kai taip atsitinka, kontrolinio skaičiaus vietoje rašomas ženklas X.

Brūkšninio kodo tikrinimas

Kitas kontrolinio skaitmens pavyzdys yra bakalėjos gaminio brūkšninis kodas. Jis nustatomas kita formule (tokia pat formule tikrinami ir 13 skaitmenų knygų kodai).

Brūkšninis kodas (angl. *barcode*) – įvairaus pločio nevienodu atstumu išdėstytų lygiagrečių brūkšnelių seka. Tai vizualus, įrenginiams įskaitomos informacijos pateikimo būdas. Kiekvienas skaitmuo žymimas dviem skirtingo pločio brūkšniais ir dviem tarpais.

Brūkšninius kodus nuskaito brūkšninių kodų skaitytuvai (dar vadinami optiniais skeneriais). Brūkšninį kodą sudaro: šalies kodas, gamintojo kodas, prekės kodas, kontrolinis skaičius.

Jei brūkšninis kodas nuskaitomas neteisingai, tai paskutinis (kontrolinis) skaitmuo skiriasi nuo jo apskaičiuotos reikšmės. Kai taip nutinka, kasininkas dar kartą nuskaito kodą. Kontrolinis skaitmuo naudojamas banko sąskaitų, mokesčių, traukinių numeriuose ir kitur, kur žmonėms reikia įsitikinti, kad naudojamas numeris yra teisingas.



Traškos duonos riekelių „Jorė“ dėžutės brūkšninis kodas

Labiausiai paplitę yra 13 eilučių europietiški kodai EAN-13 (angl. *European Article Numbering*) ir suderinamas su jais 13 eilučių kodas UPC, naudojamas Kanadoje ir JAV.

Kontrolinės sumos skaičiavimas

1. Sudedami lyginėse kodo vietose esantys skaitmenys, suma patrigubinama.
2. Prie gauto rezultato pridedami visi kiti kodo skaitmenys, išskyrus kontrolinį.
3. Paskutinis gautos sumos skaitmuo atimamas iš 10.

Jei skirtumas nesutampa su kontroliniu skaičiumi, kodas netikras, prekė padirbta.

Apskaičiuokite „Jorės“ dėžutės brūkšninio kodo kontrolinį skaičių. Ar jis tikras?

Brūkšinių kodų tikrinimo užduotys

1. Apžiūrėkite įvairių prekių brūkšnius kodus ir patikrinkite jų kontrolines sumas.
2. Raskite daugiau informacijos apie brūkšnius kodus, paskaitykite ir pasidalinkite įdomesniais faktais.



Patikrink šios knygos brūkšninį kodą!

3. Bibliotekoje, namie ar internete suraskite knygų ir patikrinkite jų ISBN kodus. Ar jų kontrolinės sumos teisingos?

Kartais būna klaidų.

Dažnai pasitaikančios klaidos:

- ☹️ pakeista skaitmens reikšmė;
- ☹️ du skaitmenys sukeisti vietomis;
- ☹️ skaitmuo įterptas į numerį;
- ☹️ skaitmuo praleistas numeryje.

Užduotis – galvosūkis. Ar galėtumėte surasti knygą, kurios kontrolinė suma lygi 10?

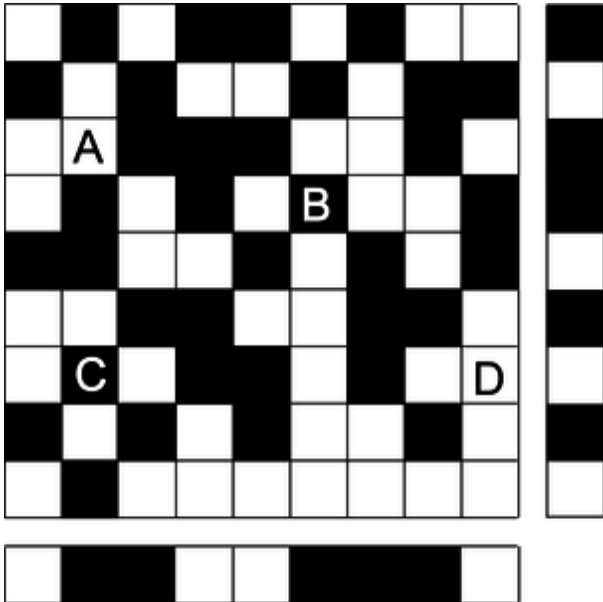
Diskusija

Kokios klaidos nebūtų galima nustatyti? Ar galima pakeisti skaitmenį ir gauti teisingą kontrolinę sumą? Kas nutiks, jei du skaitmenys bus sukeisti vietomis (dažna skaitmenų rinkimo klaida)?

Klaidų paieškos uždaviniai

Kodavimo klaida

Bebras sudarė paveikslą iš juodų ir baltų kvadratėlių. Telefonu papasakojo kitam bebrui, kaip jį persibraižyti. Norėdamas įsitikinti, kad paveikslas buvo perbraižytas be klaidų, bebras pridėjo papildomus stulpelį ir eilutę. Jeigu paveikslo eilutėje yra lyginis skaičius juodų kvadratėlių – papildomo stulpelio eilutės gale kvadratėlis nudažomas juodai, priešingu atveju – paliekamas baltas. Tą patį jis padarė ir su stulpeliais.



Bebras įsitikinęs, kad papildomai nubraižytuose eilutėje ir stulpelyje klaidų nėra. Tačiau viena klaida buvo padaryta, perbraižant paveikslą. Kuris langelis nuspalvintas neteisingai?

A B C D

Paiškinimas. Tai 2018 metų „Bebro“ konkurso vienas iš uždavinių, pateiktų spręsti vyresniųjų klasių mokiniams. Šiame uždavinyje pristatomas klaidų aptikimas ir taisymas. Kompiuterio žodžio (baito) persiuntimo teisingumas tikrinamas, papildant jį tokiu lyginimo bitu, kad viso žodžio bitų suma būtų lyginė. Jeigu patikrinus tokį persiūtą žodį paaiškėja, kad jo bitų suma nelyginė, reiškia, kad įvyko klaida.

Tokie kodai kaip šiame pavyzdyje naudojami net QR kode (tai – standartizuotas dvimatis brūkšninio kodo atitikmuo), kuriuo galima užkoduoti iki 7000 ženklų.

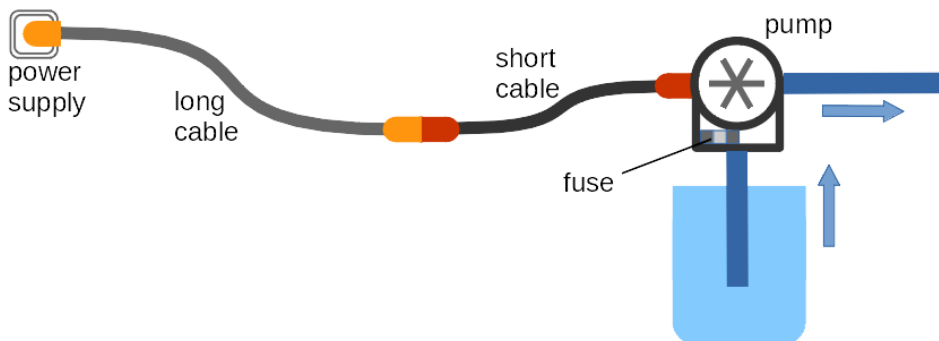
Laistymo sistema

Bebras augina gėles ir daržoves. Jis sukonstravo dvi vienodas laistymo sistemas (vieną gėlėms, kitą daržovėms laistyti), viena iš jų pavaizduota paveiksle.

Laistymo sistemą sudaro:

- Ilgas laidas
- Trumpas laidas
- Siurblys
- Siurblio saugiklis (siurblys neveikia, jei saugiklis yra sugedęs)

Sistema jungiama į elektros tinklą.



Vieną dieną ši gėlių laistymo sistema sugedo. Bebras tvirtina, kad vandens talpykla ir vandens žarnos yra tvarkingos. Visos daržovių laistymo sistemos dalys yra tvarkingos ir jos gali būti naudojamos gėlių sistemos gedimui aptikti.

Klausimas

Sugedusioje sistemoje yra tik viena bloga dalis. Bebras ieško gedimo.

Kurie teiginiai yra teisingi?

1. Ieškant gedimo, būtina pradėti nuo siurblio, nes tai svarbiausia laistymo sistemos dalis.
2. Keičiant sistemos dalis po vieną galima visada nustatyti sugedusią dalį: jei, pakeitus gėlių laistymo sistemos vieną dalį atitinkama daržovių laistymo sistemos dalimi, sistema pradės veikti, tai bus aišku, kad ta dalis yra sugedusi.
3. Pirmiausia reikia įsitikinti, ar yra elektros srovė ir toliau tęsti keitimą keičiant po vieną dalį.
4. Ieškant gedimo reikia keisti po dvi dalis, nes toks būdas visada bus greitesnis už keitimą po vieną dalį.

Paaiškinimas. Tai 2017 metų vienas iš „Bebro“ konkurso uždavinių. Informacinių sistemų derinimas yra labai svarbus programavimo etapas. Nėra paprasta rasti programoje vieną klaidą, o dar sudėtingiau, jei klaidų daugiau. Dėl to programų kūrėjai išsamiai testuoja programas, atskirai ir po kelias, derina įvairius komponentus tarpusavyje, atnaujinus kurį nors komponentą vėl derina, kad būtų galima iš karto aptikti klaidas.

https://en.wikiquote.org/wiki/John_Gall, <https://en.wikipedia.org/wiki/Systemantics>

Sprendimai ir užuominos

ISBN 10 skaitmenų kodo kontrolinės sumos klaida yra nenustatoma, kai vienas skaitmuo padidėja, o kitas sumažėja. Tada sumos yra vienodos, o kodai skirtingi. Naudojant aprašytą skaičiavimo metodą, vargu ar tai kada nutiks. 13 skaitmenų ISBN sistemoje yra kitokio tipo nenustatomų klaidų: trys iš eilės einantys skaitmenys apverčiami. Tačiau dauguma dažniausių klaidų (vienas neteisingai surinktas skaitmuo arba vietomis sukeisti du skaitmenys) dažniausiai aptinkama.

Kodavimo klaida

Teisingas atsakymas yra C.

Papildomas paskutinis kvadratėlis antrame stulpelyje, kur yra A ir C kvadratėliai (antra ir septinta eilutės), nuspalvintas ne ta spalva. Vadinasi, neteisingai nuspalvintas arba A, arba C kvadratėlis.

Paskutinio septintosios eilutės, kur yra D, kvadratėlio spalva netinka, vadinasi, bent vienas šios eilutės kvadratėlis nuspalvintas netinkamai. Panagrinėję sankirtą su antruoju stulpeliu, matome, kad joje yra kvadratėlis C, vadinasi, jis ir nuspalvintas neteisingai.

Laistymo sistema

Antras ir trečias atsakymai yra teisingi. Jie abu atskleidžia teisingas sugedusios dalies nustatymo strategijas.

Pirmas atsakymas yra neteisingas, nes kiekvienos dalies gedimas vienodai tikėtinas.

Ketvirtas atsakymas yra neteisingas vien dėl to, kad yra žodis „visada“: vidutiniškai keitimas po dvi dalis yra greitesnis, tačiau ne visada. Jei sugedus pirmoji dalis, tai keičiant po vieną, ji iš karto aptinkama, o keičiant po dvi dalis, reikės dar vieno tikrinimo.

Šaltiniai

Error Detection And Correction in Computer Network. Free online course. Data Flair. <https://data-flair.training/blogs/error-detection-and-correction-in-computer-network/>

Computer Science Unplugged. csunplugged.org

Error Detection in Computer Networks | Simple Parity Check, Checksum, CRC. DigitalNotesHub. <https://digitalnoteshub.com/error-detection-in-computer-networks/>