

Turinys

1. Tiesiniai algoritmai	6
1. Pamoka.....	6
2. Akvariumas.....	6
3. Taupyklė.....	6
4. Automobilis.....	6
5. Statybininkas.....	7
6. Trapecijos plotas.....	7
7. Puodelių pakavimas.....	7
8. Keltas.....	7
9. Gimtadienis.....	8
10. Mariaus saldainiai.....	8
2. Šakotieji algoritmai	8
1. Parašutinininkas.....	8
2. Didžioji kūdrinė varlė.....	9
3. Matematika.....	9
4. Bandelės.....	9
5. Geriausia klasė.....	10
6. Žiemojantys paukščiai.....	10
7. Jonuko problema.....	10
8. Skaičiai.....	11
3. Cikliniai algoritmai. Žinomo kartojimų skaičiaus ciklas	11
1. Konkursas.....	11
2. Kelias į mokyklą.....	11
3. Snaigės už lango.....	12
4. Kalėdinės eglutės.....	12
5. Draugų skaičiai.....	13
6. Kauliukai.....	13
7. Šuoliai per virvutę.....	14
8. Bėgimo varžybos.....	14
9. Pirkiniai.....	15
4. Cikliniai algoritmai. Nežinomo kartojimų skaičiaus ciklas	15
1. Voverytė.....	15
2. Degalai.....	15
3. Saldainiai.....	16
4. Knyga.....	16
5. Kurjeris.....	16
6. Skaičiuotuvas.....	17
7. Karnavalas.....	17
8. Atvirukai.....	18
5. Skaitymas iš failo. Rašymas į failą	18
1. Bendrabutis.....	18
2. Piramidės.....	19
3. Viščiukai.....	20
4. Žaidimo kauliukas.....	20
5. Vėjo jėgainė.....	21

6. Ciklas cikle	21
1. Skaičių dalyba.....	21
2. Uždavinių sprendimo varžybos.....	21
3. Automobilių lenktynės.....	22
4. Kauliukai.....	23
5. Beveik tobulieji skaičiai.....	23
7. Šakotieji ir cikliniai algoritmai	24
1. Temperatūra.....	24
2. LEGO robotas.....	25
3. Keliautojai.....	26
4. Lenktynės.....	26
5. Kelių eismo taisyklės.....	26
8. Funkcija, gražinanti apskaičiuotą reikšmę per funkcijos vardą	27
1. Skaitymas.....	27
2. Plytelės.....	27
3. Trikampio plotas.....	28
4. Laimingi bilietai.....	28
5. Taupymas.....	28
6. Mobilusis telefonas išsimokėtinai.....	29
7. Kelionė.....	30
8. Kodavimas.....	31
9. Funkcija su parametrais-nuorodomis	32
1. Bilietų kaina.....	32
2. Kelionė.....	32
3. Lyginumas.....	33
4. Gamyba.....	34
5. Dieta.....	34
6. Skrajutės.....	35
7. Paprastoji trupmena.....	36
10. Masyvai. Veiksmai su masyvų elementais. Sumos, kiekio, vidurkio skaičiavimas	37
1. Avalynės parduotuvė.....	37
2. Moliūgai.....	38
3. Taupymas.....	38
4. Varžybos.....	40
5. Balsavimas.....	42
6. Svetainės populiarumas.....	42
7. Geografijos kontrolinio darbo pažymių pasiskirstymas.....	43
11. Masyvai. Didžiausios ir mažiausios reikšmių paieška	44
1. Dailusis čiuožimas.....	44
2. Indėliai.....	44
3. Teleloto.....	45
4. Pinigų keitimas.....	46
5. Mokinių ūgiai.....	47
6. Autobuso vairuotojo darbo ir poilsio režimas.....	48

12. Masyvo elementų šalinimas ir įterpimas	49
1. Informacinių technologijų konkursas.....	49
2. Komanda.....	50
3. Žalgirio komandos 2013/2014 metų sezonas.....	51
4. Parduotuvė.....	51
13. Dvimatis masyvas	52
1. Didžiausio ir mažiausio elementų minimali suma.....	52
2. Matematikos kontroliniai darbai.....	53
3. Sandaugos įstrižainė.....	53
4. Magiškas kvadratas.....	54
5. Laivų mūšis.....	55
14. Įvairūs uždaviniai, kuriuose naudojamas masyvo duomenų tipas	56
1. Slėpynės.....	56
2. Pakuotės.....	56
3. Informacijos šifravimas.....	57
4. Televizijos laidos reitingas.....	59
5. Automobiliai.....	60
15. Simboliai	61
1. Balsės A ir E.....	61
2. Linksmas sukeitimas.....	61
3. Kodavimas.....	62
4. Kentauras.....	62
5. Būdvardžių linksniuotės.....	63
16. Simbolių eilutės	64
1. Žodžių paieška.....	64
2. Simbolių eilutės: Skirtingi Simboliai.....	64
3. Žodžiai palindromai.....	65
4. Kalėdinė Eglutė.....	66
5. Olimpinės plaukimo varžybos.....	67
17. Struktūros (įrašai). Struktūrų (įrašų) masyvai	68
1. Oro temperatūros matavimai.....	68
2. Įnoringiausia princesė.....	69
3. Padangų žymėjimas.....	69
18. Veiksmai su struktūrų (įrašų) masyvų elementais	70
1. Prekyba elektronikos prekėmis.....	70
2. Gimtadieniai.....	71
3. Mokinių vidurkiai.....	72
4. Šviesuolių miestelio skaitytojai.....	73
19. Struktūrų (įrašų) masyvų elementų rikiavimas	74
1. Populiariausios programavimo kalbos.....	74
2. Įdomiausi filmai.....	75
3. Skaniausi saldainiai.....	76
4. Geriausi draugai.....	77
20. Struktūrų (įrašų) masyvų elementų šalinimas	78
1. Prekyba buitine technika.....	78
2. Registracija.....	79
3. Muzikos albumų parduotuvė.....	80

21. Įvairūs uždaviniai su struktūrų (įrašų) masyvais	82
1. <i>Buto paieška.</i>	82
2. <i>Programuotojų klasė.</i>	82
3. <i>Krepšinis.</i>	83
4. <i>Orientacinio sporto varžybos.</i>	85

1. Tiesiniai algoritmai

1. Pamoka. Parašykite programą, kuri padėtų Petriukui suskaičiuoti, kiek pamokų jis turi per savaitę ir kiek tai sudarys minučių. Klaviatūra įvedami 5 skaičiai, reiškiantys kiekvienos dienos pamokų skaičių.

Duomenys	Rezultatai
Kiek pamokų yra pirmadienį? 5 Kiek pamokų yra antradienį? 6 Kiek pamokų yra trečiadienį? 4 Kiek pamokų yra ketvirtadienį? 5 Kiek pamokų yra penktadienį? 4	Pamokų skaičius: 24 Tai sudaro minučių: 1080

2. Akvariumas. Akvariume gyvena a žuvų. Kiekvieną dieną Petriukas į akvariumą įdeda b žuvų. Parašykite programą, kuri suskaičiuotų kiek iš viso bus žuvų po n dienų. Rezultatą reikia išvesti su paaiškinamaisiais žodžiais.

Duomenys	Rezultatai
Kiek žuvų gyvena akvariume? 5 Kiek žuvų į akvariumą įdedama kiekviena diena? 3 Kiek dienų praėjo? 3	Po 3 dienų akvariume gyvens 14 žuvų.

3. Taupyklė. Jonas turi kiaulę taupyklę, kurioje yra a monetų po 5 ct, b monetų po 20 ct ir c monetų po 2 Lt. Kitokios vertės monetų taupyklėje nėra. Parašykite programą, kuri suskaičiuotų, kiek pinigų kiaulėje taupyklėje iš viso turi Jonas. Atsakymą pateikite litais, pvz.: kai taupyklėje yra 12 monetų po 5 ct, 5 monetos po 20 ct ir 6 monetos po 2 Lt, tuomet ekrane turi būti rodoma: Taupyklėje yra 13.60 Lt.

Duomenys	Rezultatai
Kiek yra monetų po 5 ct? 5 Kiek yra monetų po 20 ct? 0 Kiek yra monetų po 2 Lt? 3	Taupyklėje yra 6.25 Lt.

4. Automobilis. Automobilių tunelio po Nepriklausomybės aikšte Vilniuje ilgis lygus 264 m. Parašykite programą, kuri apskaičiuotų, kelias sekundes s automobilis važiuoja šiuo tuneliu, jei jo greitis yra v km/h? Rezultatus pateikite šimtųjų tikslumu.

Duomenys	Rezultatai
Koks automobilio greitis? 60	Automobilis tunelį pravažiuos per 15.84 s

5. Statybininkas. Statybininkui reikia pastatyti sieną, kurios ilgis yra a metrų, o aukštis h metrų (a ir h – sveikieji skaičiai). Kiek reikės plytų, kurių ilgis 20 cm, o aukštis 10 cm ir kiek kainuos plytos, jeigu vienos plytos kaina k Lt. Pinigų sumą pateikti šimtųjų tikslumu.

Duomenys	Rezultatai
Sienos ilgis 4 m Sienos aukštis 3 m Plytos kaina 0.5 Lt	Plytų kiekis: 600 Plytos kainuos 300.00 Lt

6. Trapecijos plotas. Parašykite programą, kuri, įvedus trapecijos pagrindų a ir b bei aukštinės h ilgius, apskaičiuotų trapecijos plotą.

Duomenys	Rezultatai
Trapecijos ilgesniojo pagrindo ilgis: 5 Trapecijos trumpesniojo pagrindo ilgis: 3 Trapecijos aukštinės ilgis: 4	Trapecijos plotas: 16

7. Puodelių pakavimas. Į vieną kartoninę dėžutę telpa trys puodeliai. Pakuotojas užključiuoja dėžutę ir išsiunčia ją į parduotuvę, jei ji pilna. Iš viso reikia supakuoti m puodelių. Parašykite programą, kuri apskaičiuotų, kelios bus pilnos dėžutės ir kiek puodelių liks nesupakuota.

Duomenys	Rezultatai
Puodelių, kuriuos reikia supakuoti, skaičius: 7	Pilnų dėžučių skaičius: 2 Nesupakuotų puodelių skaičius: 1

8. Keltas. Parašykite programą, kuri suskaičiuotų, kelis kartus keltui teks kelti per upę k automobilių, jeigu vienu metu jis gali perkelti m automobilių. Keltas kelia tik tada, kai yra pilnas (susidaro m automobilių.) Taip pat išveskite automobilių skaičių, kuriems persikelti per upę nepavyks (jei buvo 11 automobilių, o į keltą telpa 10, tai 10 perkels, o vienas liks neperkeltas).

Duomenys	Rezultatai
Automobilių skaičius: 27 Į keltą telpa automobilių: 10	Perkels per kartų: 2 Liks neperkelta: 7

9. Gimtadienis. Tautvydas rengia gimtadienio šventę. Norėdamas pavaišinti svečius, jis iškepė a sausainių. Prasidėjus šventei, jis pastebėjo, kad dar b draugų iškepė lygiai tiek pat sausainių kaip ir jis, ir atsinešė į gimtadienį. Šventėje iš viso dalyvavo c žmonių (įskaitant ir patį jubiliatą). Norėdamas, kad nei vienas neliktų nuskriaustas, Tautvydas sausainius visiems svečiams padalijo po lygiai ir, kadangi šiandien jo gimtadienis, likusius po dalybų nusprendė pasilikti sau. Parašykite programą, kuri apskaičiuotų, po kiek sausainių gavo kiekvienas gimtadienio dalyvis ir kiek papildomai sausainių atiteko Tautvydui.

Duomenys	Rezultatai
Kiek sausainių iškepė Tautvydas? 20 Keli draugai dar atsinešė po tiek pat sausainių? 4 Kiek žmonių iš viso dalyvavo šventėje? 24	Kiekvienas šventės dalyvis gavo po 4 sausainius Tautvydui papildomai atiteko 4 sausainiai

10. Mariaus saldainiai. Marius kiekvieną dieną gauna po n saldainių. Jis suvalgo po a saldainių, o likusius kaupia Kalėdų dovanoms. Kai Marius pradėjo kaupti saldainius, iki Kalėdų buvo likę k dienų. Parašykite programą, kuri suskaičiuotų keliems draugams d Marius galės paruošti kalėdinius saldainių rinkinius, jei kiekviename rinkinyje bus po a saldainių ir kiek saldainių s liks supakavus dovanas.

Duomenys	Rezultatai
Kiek saldainių gauna Marius? 15 Po kiek saldainių suvalgo? 4 Kelios dienos liko iki Kalėdų? 22	Marius dovanas paruoš 60 draugų. Supakavus dovanas liks 2 saldainiai.

2. Šakotieji algoritmai

1. Parašiutininkas. Parašiutininkas šoka iš h metrų aukščio. Vos iššokęs, iškart pradeda skleisti parašiutą, kuris iki galo išsiskleidžia per t sekundžių. Parašykite programą, kuri apskaičiuotų, ar parašiuotas spės išsiskleisti prieš parašiutininkui pasiekiant žemę. Laikas, per kurį objektas nukrenta iš aukščio h randamas pagal formulę: $t = \sqrt{\frac{2h}{g}}$. Čia g - laisvojo kritimo pagreitis, kuris yra lygus 9.8 m/s^2 .

Pavyzdžiui, įvedus tokius pradinis duomenis, į ekraną turėtų būti išvedami rezultatai:

Iš kokio aukščio šoka parašiutininkas? 200 Per kiek sekundžių išsiskleidžia jo parašiutas? 2	Parašiutas išsiskleis
Iš kokio aukščio šoka parašiutininkas? 50.9 Per kiek sekundžių išsiskleidžia jo parašiutas? 3.3	Parašiutas neišsiskleis

2. Didžioji kūdrinė varlė. Didžioji kūdrinė varlė – beuodegis varliagyvis. Ji sveria m gramų (m - realusis skaičius). Mokslininkai, tyrinėjantys didžiąsias kūdrines varles, nusprendė n varlių stebėti. Jei stebimos varlės svers daugiau kaip 5 kilogramus, ekrane turi būti rodomas pranešimas „Varlių stebėjimui pakanka“, jei mažiau – „Varlių stebėjimui per mažai“.

Kiek sveria varlė? 100 Kiek varlių norima stebėti? 1000	Varlių stebėjimui pakanka
Kiek sveria varlė? 75 Kiek varlių norima stebėti? 50	Varlių stebėjimui per mažai

3. Matematika. Petriukas per pusmetį gavo 5 matematikos pažymius. Mokytoja nusprendė padaryti vaikams staigmeną: mokiniai, kurių pažymių vidurkis yra didesnis už 9, gaus tris saldinius, o mokiniams, kurių vidurkis yra tarp 7 ir 9, įskaitant intervalo galus, bus apdovanoti dviem saldainiais. Visi likusieji gaus po vieną saldainį.

Parašykite programą, kuri pagal įvestus Petriuko pažymius apskaičiuotų, kiek saldainių jis gaus.

Kokius pažymius gavo Petriukas? 8 9 6 5 10	Petriukas gaus du saldainius
Kokius pažymius gavo Petriukas? 10 10 8 9 10	Petriukas gaus tris saldainius
Kokius pažymius gavo Petriukas? 5 5 4 5 5	Petriukas gaus vieną saldainį

4. Bandelės. Julius, grįžęs iš mokyklos namo, rado tokį mamos raštelį:

Nueik į parduotuvę ir nupirk bandelių su varške.

Jei viena bandelė kainuos ne daugiau kaip a Lt, nupirk $n1$ bandelių;

Jei daugiau nei a , bet mažiau nei b , nupirk $n2$ bandelių;

O jei kainuos b arba daugiau, negu $b - n3$ bandelių.

Nuėjęs į parduotuvę, Julius pamatė, kad bandelė kainuoja k Lt.

Parašykite programą, kuri apskaičiuotų, kiek pinigų už bandeles sumokės Julius.

(a visada mažiau nei b).

Pavyzdžiui, ekrane įvedus tokius duomenis, turėtų būti išvedami tokie rezultatai:

Įveskite kainas a ir b : 1.50 1.70 Įveskite kiekius $n1$, $n2$, $n3$: 3 2 1 Įveskite bandelės kainą: 1.30	Už bandeles bus sumokėta: 3.90 Lt.
Įveskite kainas a ir b : 1.50 1.70 Įveskite kiekius $n1$, $n2$, $n3$: 3 2 1 Įveskite bandelės kainą: 1.60	Už bandeles bus sumokėta: 3.20 Lt.
Įveskite kainas a ir b : 1.50 1.70 Įveskite kiekius $n1$, $n2$, $n3$: 3 2 1 Įveskite bandelės kainą: 2.05	Už bandeles bus sumokėta: 2.05 Lt.

5. Geriausia klasė. Giraitės mokykloje yra keturios dešimtos klasės: a, b, c ir d. Direktorius atlieka analizę, nori surasti geriausiai besimokančią dešimtokų klasę, pasižiūrėti, keliais balais kiekvienos kitos klasės vidurkis yra mažesnis už geriausiai besimokančios klasės vidurkį. Parašykite programą, kuri surastų, koks yra didžiausias vidurkis ir keliais balais skiriasi likusių klasių vidurkiai nuo geriausiai besimokančios klasės vidurkio.

Įveskite klasių vidurkius: 7 8.5 9 6	Didžiausias vidurkis: 9 Kitų klasių vidurkiai skiriasi: 2, 0.5, 3
Įveskite klasių vidurkius: 7 8 9.5 6	Didžiausias: 9.5 Kitų klasių vidurkiai skiriasi: 2.5, 1.5, 3.5

6. Žiemojantys paukščiai. Gamtininkas registruoja likusius žiemoti paukščius. Jo tikslas yra nustatyti, kurios iš trijų paukščių rūšių atstovų liko žiemoti daugiausia. Klaviatūra įvedami trijų paukščių rūšių kiekiai, parašykite programą, kuri surikiuotų juos iš eilės nuo didžiausio iki mažiausio ir apskaičiuotų, kiek skiriasi didžiausias ir mažiausias kiekiai.

Įveskite kiek kiekvienos rūšies paukščių liko žiemoti: 1000 1500 1800	1800 1500 1000 Skirtumas tarp didžiausio ir mažiausio kiekio: 800
Įveskite kiek kiekvienos rūšies paukščių liko žiemoti: 5000 3500 6200	6200 5000 3500 Skirtumas tarp didžiausio ir mažiausio kiekio: 2700

7. Jonuko problema. Jonukas žino, kad mokytoja jam gali duoti spręsti vieną iš trijų kontrolinio darbo variantų. Atlikdamas užduotį jis gaus du skaičius a ir b , o jam reikės apskaičiuoti x reikšmę pagal vieną iš trijų formulių.

Pirmas variantas: $x = ab+3$;

Antras variantas: $x = a+b$;

Trečias variantas: $x = a-b$.

Parašykite programą, kuri padėtų Jonukui patikrinti, ar jis teisingai skaičiuoja x reikšmę sprendamas kiekvieno varianto užduotį.

Įveskite užduoties variantą. 1 Įveskite a reikšmę. 5 Įveskite b reikšmę. 2	Atsakymas: $x = 13$
Įveskite užduoties variantą. 2 Įveskite a reikšmę. 5 Įveskite b reikšmę. 2	Atsakymas: $x = 2$

Įveskite užduoties variantą. 3 Įveskite a reikšmę. 2 Įveskite b reikšmę. 3	Atsakymas: $x = -1$
--	---------------------

8. Skaičiai. Per matematikos pamoką teko nustatyti, kurie iš duotųjų teigiamų skaičių a , b ir c yra skaičiaus trys kartotiniai. Jei yra keli, reikia išvesti pirmą rastą, jeigu nėra nei vieno, reikia išvesti - nėra.

Parašykite programą, sprendžiančią šį uždavinį.

Įveskite a reikšmę. 5 Įveskite b reikšmę. 2 Įveskite c reikšmę. 3	Atsakymas: 3
Įveskite a reikšmę. 6 Įveskite b reikšmę. 3 Įveskite c reikšmę. 7	Atsakymas: 6
Įveskite a reikšmę. 1 Įveskite b reikšmę. 2 Įveskite c reikšmę. 5	Atsakymas: nėra

3. Cikliniai algoritmai. Žinomo kartojimų skaičiaus ciklas

1. Konkursas. Restorano „Skanu“ vadybininkas sugalvojo surengti konkursą lankytojams, kurio metu galima laimėti marškinėlius su restorano logotipu. Visi lankytojai kartu su sąskaita gauna po vieną kortelę, ant kurios parašytas sveikasis teigiamas skaičius iš intervalo $[a;b]$ (a – intervalo pradžia, b – intervalo pabaiga). Laimi tie lankytojai, kurių kortelėse įrašytas skaičius dalijasi iš 6. Parašykite programą, kuri apskaičiuotų, kiek marškinėlių reikia užsakyti restorano vadybininkui.

Duomenys	Rezultatai
Įveskite intervalo pradžia: 5 Įveskite intervalo pabaiga: 24	Reikalingų marškinėlių skaičius: 4
Įveskite intervalo pradžia: 31 Įveskite intervalo pabaiga: 62	Reikalingų marškinėlių skaičius: 5

2. Kelias į mokyklą. Kiekvieną dieną Petriukas, eidamas į mokyklą, skaičiuoja kiekvieną savo žingsnį ir žaidžia tokį žaidimą: kai žingsnių skaičius baigiasi nuliu, Petriukas suploja rankomis, o kai penketu - spragteli pirštais. Parašykite programą, kuri suskaičiuotų, kiek kartų Petriukas suplos rankomis ir kiek – spragtels pirštais, jei jam iki mokyklos yra lygiai n žingsnių.

Duomenys	Rezultatai
Įveskite žingsnių kiekį iki mokyklos: 15	Suplojimų bus: 1 Spragtelėjimų bus: 2
Įveskite žingsnių kiekį iki mokyklos: 426	Suplojimų bus: 42 Spragtelėjimų bus: 43

3. Snaigės už langą. Per atostogas Simas turėjo daugiau laisvo laiko ir nutarė suskaičiuoti, kiek sningant po jo namo langą nukrenta snaigių. Jis pastebėjo, kad kiekvieną kitą sekundę nukrenta dvigubai daugiau snaigių, nei prieš tai buvusią. Parašykite programą, skaičiuojančią kiek snaigių s bus nukritę per n sekundžių, kai per pirmąją sekundę nukrito k snaigių.

Duomenys	Rezultatai
Įveskite, kiek snaigių nukrito per pirmąją sekundę ir kiek sekundžių snigo: 5 3	35
Įveskite, kiek snaigių nukrito per pirmąją sekundę ir kiek sekundžių snigo: 2 4	30

4. Kalėdinės eglutės. Prieš Kalėdas miško urėdijos prekiauja įvairaus aukščio eglutėmis. Į prekybos vietą atvežta n eglučių. Jų aukščiai yra e_1, e_2, \dots, e_n . Parašykite programą, skaičiuojančią vidutinį eglutės aukštį.

Duomenys	Rezultatai
Kiek eglučių atvežta? 6 Įveskite 1 eglutės aukštį: 167 Įveskite 2 eglutės aukštį: 134 Įveskite 3 eglutės aukštį: 145 Įveskite 4 eglutės aukštį: 156 Įveskite 5 eglutės aukštį: 155 Įveskite 6 eglutės aukštį: 176	Eglutės aukščio vidurkis: 155.5 cm

5. Draugų skaičiai. 10 draugų sugalvojo po vieną skaičių iki 100 ir juos užrašė ant kortelių. Parašykite programą, surandančią keli draugai sugalvojo lyginius skaičius. Jeigu tokių skaičių bus, ekrane turi būti rodomas jų kiekis, priešingu atveju ekrane turi būti rodomas žodis Nėra.

Duomenys	Rezultatai
Įveskite draugų sugalvotus skaičius: 2 9 4 100 25 5 6 3 5 85	Atsakymas: 4
Įveskite draugų sugalvotus skaičius: 5 3 99 55 35 47 11 63 51 91	Atsakymas: Nėra

6. Kauliukai. Tomas dalyvauja loterijoje. Jis meta n standartinių lošimo kauliukų (ant jų sienelių skaičiai nuo 1 iki 6). Kiekvieno skaičiaus iškritimo tikimybė yra vienoda. Loterija laimima tada, jei Tomo išridentų skaičių suma yra didesnė nei pusė visos galimos taškų sumos. Parašykite programą, kuri nustatytų:

- koks maksimalus taškų kiekis;
- kiek iš viso taškų surinko Tomas;
- koks jo surinktų taškų vidurkis
- ar berniukas laimėjo loterijoje.

Pradiniai duomenys – kauliukų kiekis n ir išridentas kiekvieno kauliuko skaičius.

Duomenys	Rezultatai
Įveskite kauliukų kiekį: 4 1-o kauliuko taškų kiekis: 5 2-o kauliuko taškų kiekis: 3 3-o kauliuko taškų kiekis: 2 4-o kauliuko taškų kiekis: 1	Iš viso buvo galima surinkti taškų: 24 Tomas iš viso surinko: 11 taškų Jo taškų vidurkis: 2.8 Loterija pralaimėta.

Įveskite kauliukų kiekį: 7 1-o kauliuko taškų kiekis: 6 2-o kauliuko taškų kiekis: 6 3-o kauliuko taškų kiekis: 2 4-o kauliuko taškų kiekis: 2 5-o kauliuko taškų kiekis: 5 6-o kauliuko taškų kiekis: 4 7-o kauliuko taškų kiekis: 5	Iš viso buvo galima surinkti taškų: 42 Tomas iš viso surinko: 30 taškų Jo taškų vidurkis: 4.3 Loterija laimėta.
--	--

7. Šuoliai per virvutę. Kazys ir Onutė tikra komanda: Kazys padeda Onutei ruošti sporto rungčiai, kurioje Onutė šokinėja per virvutę. Ji kartoja šuolius m kartų ir pirmuoju bandymu atlieka $k1$, antruoju – $k2$, ..., m -tuoju – km šuoliukų. Parašykite programą, kuri padėtų Kaziui suskaičiuoti, kiek kartų iš viso Onutė peršoko per virvutę ir koks vidutinis vienu bandymu atliktų šuoliukų skaičius.

Duomenys	Rezultatai
Kiek kartų šokinėjo:3 Kiek sušokinėjo kartų 1 bandymu: 20 Kiek sušokinėjo kartų 2 bandymu: 40 Kiek sušokinėjo kartų 3 bandymu: 30	Iš viso: 90 Vidutiniškai: 30
Kiek kartų šokinėjo:2 Kiek sušokinėjo kartų 1 bandymu: 10 Kiek sušokinėjo kartų 2 bandymu: 20	Iš viso: 30 Vidutiniškai: 15

8. Bėgimo varžybos. Varžybose dalyvauja n bėgikų. Pirmasis bėgikas įveikė distanciją per $k1$ sekundžių, antrasis – per $k2$, ..., n -tasis – per kn . Parašykite programą, surandančią, kuris bėgikas įveikė distanciją greičiausiai (išspausdinti bėgiko laiką) ir keliomis sekundėmis jis buvo greitesnis už vidutiniškai bėgusį bėgiką.

Duomenys	Rezultatai
Kiek dalyvavo bėgikų: 3 Įveskite 1 bėgiko laiką: 25 Įveskite 2 bėgiko laiką: 20 Įveskite 3 bėgiko laiką: 32	Greičiausio bėgiko laikas: 32 sek. Jis buvo 7 sek geresnis už vidurkį.
Kiek dalyvavo bėgikų: 4 Įveskite 1 bėgiko laiką: 22 Įveskite 2 bėgiko laiką: 20 Įveskite 3 bėgiko laiką: 25 Įveskite 4 bėgiko laiką: 25	Greičiausio bėgiko laikas: 20 sek. Jis buvo: 3 sek geresnis už vidurkį.

9. Pirkiniai. Mama liepė Petriukui nupirkti n pirkinių. Kiekvienas pirkinys turi savo kainą (centais) ir svorį (gramais). Petriukas gali panešti tik iki 5 kg. Parašykite programą, kuri suskaičiuotų, kiek vidutiniškai kainuoja vienas pirkinys, ir ar Petriukas galės parnešti visus pirkinius iš parduotuvės.

Duomenys	Rezultatai
Kiek buvo pirkinių: 2 Įveskite 1 pirkinio kainą ir svorį: 105 2000 Įveskite 2 pirkinio kainą ir svorį: 1655 2550	Pirkinio vidutinė kaina: 8 Lt 80 ct. Petriukas galės parnešti pirkinius.
Kiek buvo pirkinių: 2 Įveskite 1 pirkinio kainą ir svorį: 105 3000 Įveskite 2 pirkinio kainą ir svorį: 2655 2550	Pirkinio vidutinė kaina: 13 Lt 80 ct. Petriukas negalės parnešti pirkinių.

4. Cikliniai algoritmai. Nežinomo kartojimų skaičiaus ciklas

1. Voverytė. Per vasarą voverytė sukaupe X riešutėlių. Prasidėjus žiemai voverytė kartu su savo V voveriukų sugrauzia pusryčiams, pietums ir vakarienei po vieną riešutą kiekvienas. Apskaičiuokite, kelioms dienoms K voverytei su voveriukais užteks sukauptų riešutų. Patikrinkite, ar riešutų užteks visai žiemai, jeigu žiema trunka 92 dienas.

Duomenys	Rezultatai
Sukauptas riešutų kiekis $X = 900$ Voveriukų skaičius $V = 2$	Riešutų sukaupta 99 dienoms Riešutų užteks visai žiemai
Sukauptas riešutų kiekis $X = 800$ Voveriukų skaičius $V = 3$	Riešutų sukaupta 66 dienoms Riešutų neužteks visai žiemai

2 Degalai. Šeima išsirengė į kelionę automobiliu. Jie pripildė kuro baką, kurio talpa t litrų ir nusprendė važiuoti tol, kol bake bus degalų. Lyginėmis kelionės dienomis automobilis suvartos n litrų degalų, o nelyginėmis - $2n$ litrų. Parašykite programą, kuri surastų, kiek dienų truks šeimos kelionė.

Duomenys	Rezultatai
Įveskite kuro bako talpą: 20 Įveskite kuro sąnaudas n : 5	Keliauti bus galima 3 dienų/(as)/(a).
Įveskite kuro bako talpą: 112 Įveskite kuro sąnaudas n : 11	Keliauti bus galima 7 dienų/(as)/(a).

3. Saldainiai. Petriukas gavo n saldainių. Kiekvieną dieną jis nori suvalgyti skirtingą skaičių saldainių x . Kelias dienas Petriukas galės mėgautis saldainiais ir kiek jam dar liks nesuvalgytų saldainių tuo atveju, jei paskutinei dienai saldainių nebeužtektų.

Pastaba: kiekvieną dieną suvalgomi saldainiai turi būti įvedinėjami atskirai, jie turi būti įvedinėjami tol, kol Petriukas nebus suvalgęs visų saldainių.

Duomenys	Rezultatai
Petriukas gavo saldainių: 25 Per dieną suvalgė: 7 Per dieną suvalgė: 7 Per dieną suvalgė: 8 Per dieną suvalgė: 5	Petriukui saldainių užteks 3 dienoms ir jam liks 3 saldainiai.

4. Knyga. Tadas mėgsta skaityti knygas, tačiau jam labai sunku pradėti skaityti. Knygoje yra m skyrių Pirmą dieną Tadas perskaitė 1 skyrių, antrą – 2, trečią – 3 ir t.t. Kiekvieną kitą dieną jis perskaito vienu skyriumi daugiau, negu prieš tai buvusią dieną. Programa turi apskaičiuoti, per kelias dienas d Tadas perskaitys visą knygą ir kelis skyrius s vidutiniškai per dieną perskaito Tadas. Paskutinei dienai gali likti mažiau skyrių.

Duomenys	Rezultatai
Įveskite knygos skyrių skaičių: 8	Tadas visą knygą perskaitys per 4 dienas (-ų). Tadas vidutiniškai per dieną perskaitė 2 skyrius (-ų).
Įveskite knygos skyrių skaičių: 17	Tadas visą knygą perskaitys per 6 dienas (-ų). Tadas vidutiniškai per dieną perskaitė 2.83 skyrius (-ų).

5. Kurjeris. Siuntų pervežimo įmonėje dirbantis kurjeris gavo užduotį parengti pervežimų statistiką:

1. kiek įvykdė užsakymų, kurių suma viršijo 100 Lt;
2. už kokią vidutinę sumą per dieną išvežiojo prekių;
3. kiek iš viso prekių išvežiojo.

Jis nežino, kiek užsakymų įvykdys per dieną, todėl baigęs darbą į programą įves nulį (0), tai reikš, kad darbo diena baigta ir reikia pateikti rezultatus. Parenkite programą, kuri leistų nežinomą skaičių kartų kurjeriui įvesti užsakymo sumą (skaičiavimai baigiami įvedus nulį, nulis skaičiuojant vidurkį ir kiekį nebus pridėtas) ir pateiktų skaičiavimų rezultatus.

Duomenys	Rezultatai
Įveskite sumą: 110 80 50 0	1. 1 2. 80 3. 3

Įveskite sumą: 60 90 150 200 0	1. 2 2. 125 3. 4
---	------------------------

6. Skaičiuotuvai. Danutė dirba buhalterė, todėl jai kartais reikia atlikti aritmetinius skaičiavimus su dideliais skaičių kiekiais. Ji skundžiasi, kad sunku ir nepatogu daug kartų spaudinėti skaičiuotuvo klavišus, todėl ji paprašė Jūsų, kad parašytumėte programą, kurios pradžioje pakaktų įvesti aritmetinio veiksmo simbolį, ir būtų galima įvedinėti skaičius, su kuriais bus atliekama ta operacija, operacijos pabaiga užfiksuojama įvedus nulį.

Pavyzdžiui, Danutė išsirenka sumos skaičiavimo operaciją, tada įvedinėja skaičius, juos reikia sumuoti tol, kol įves nulį, tada baigti skaičiavimą ir pateikti rezultatą.

Reikia sukurti skaičiuotuvą tokiems veiksams atlikti: suma - 1, atimtis - 2, daugyba - 3, didžiausia reikšmė sraute - 4, mažiausia reikšmė sraute - 5. (Danutė įves veiksmą reiškiantį skaičių, nepamirškite jos informuoti įjungus programą, koks skaičius kokį veiksmą reikš).

Duomenys	Rezultatai
Įveskite veiksmą: 4 15 20 70 -20 0	max: 70
Įveskite veiksmą: 1 12 10 50 0	sum: 72

7. Karnavalas. Mokykloje rengiamas karnavalas. Prieš karnavalą visi mokiniai turėjo pasiruošti kaukes. Kiekvieno mokinio kaukė buvo įvertinta tam tikru balu nuo 1 iki 10 (įvertinimai sveikieji skaičiai). Karnavalo dieną visi į karnavalą atvykę mokiniai, turėjo pranešti, kokius įvertinimus gavo. Į karnavalą galėjo patekti tik tie mokiniai, kurių kaukės įvertintos ne mažiau kaip 5 balais.

Kiek mokinių atvyko į karnavalą yra nežinoma. Parašykite programą, kuri suskaičiuotų keli mokiniai iš viso bandė patekti į karnavalą, ir keli iš jų pateko.

Pastaba: Turėtų būti įvedinėjami kiekvieno mokinio kaukės įvertinimo balai. Duomenų įvedimas baigiamas nuliu.

Duomenys	Rezultatai
Įveskite kiek balų gavo mokinys: 4 Įveskite kiek balų gavo mokinys: 2 Įveskite kiek balų gavo mokinys: 6 Įveskite kiek balų gavo mokinys: 8 Įveskite kiek balų gavo mokinys: 5	Į karnavalą ėjo 5 mokiniai, pateko 3.
Įveskite kiek balų gavo mokinys: 5 Įveskite kiek balų gavo mokinys: 5 Įveskite kiek balų gavo mokinys: 1 Įveskite kiek balų gavo mokinys: 2 Įveskite kiek balų gavo mokinys: 6	Į karnavalą ėjo 5 mokiniai, pateko 2.

8. Atvirukai. Linas nori nupirkti vienos rūšies atvirukus savo m draugams. Parduotuvėje yra n rūšių atvirukų, kurių kiekvienos rūšies kiekiai yra $k_1, k_2, k_3, \dots, k_n$. Parašykite programą, kuri apskaičiuotų, kelių rūšių atvirukų x iš parduotuvėje esančių n rūšių užtektų visiems Lino draugams.

Duomenys	Rezultatai
Keliems draugams Linas nori dovanoti atvirukus? 7 Kiek rūšių atvirukų yra parduotuvėje? 3 Kiek yra 1 rūšies atvirukų? 6 Kiek yra 2 rūšies atvirukų? 10 Kiek yra 3 rūšies atvirukų? 9	2 rūšių atvirukų užtektų visiems Lino draugams
Keliems draugams Linas nori dovanoti atvirukus? 5 Kiek rūšių atvirukų yra parduotuvėje? 3 Kiek yra 1 rūšies atvirukų? 6 Kiek yra 2 rūšies atvirukų? 10 Kiek yra 3 rūšies atvirukų? 9	3 rūšių atvirukų užtektų visiems Lino draugams

5. Skaitymas iš failo. Rašymas į failą

1. Bendrabutis. Bendrabutyje kiekviename kambaryje yra fiksuojama, kiek elektros energijos (kilovatvalandėmis) išnaudojama per mėnesį. Bendrabučio gyventojai susitarė, kad kiekvieno kambario gyventojai kas mėnesį mokės vienodą įmoką už elektrą - tai yra tiek, kiek vidutiniškai kambaryje buvo suvartota elektros. Parašykite programą, kuri surastų tą sumą.

Pradiniai duomenys pateikti duomenų faile *Duomenys.txt* atskirose eilutėse:

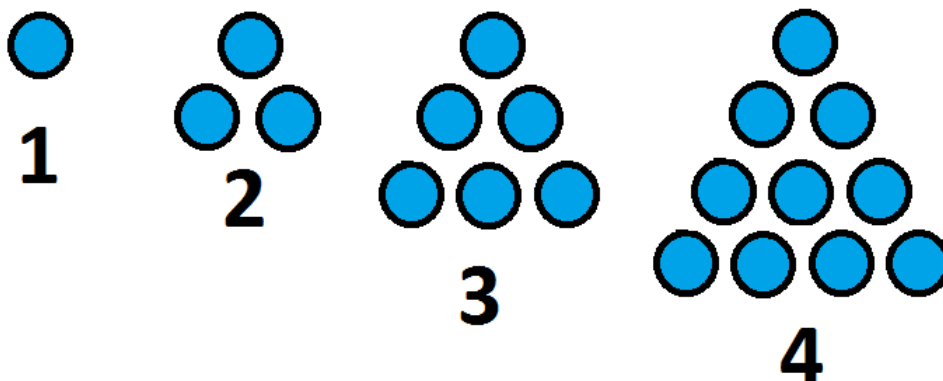
- pirmas skaičius - kambarių skaičius bendrabutyje;
- antras skaičius - elektros kaina litais už kilovatvalandę;
- likusieji n skaičių - kiekvieno kambario suvartota elektros energija kilovatvalandėmis.

Rezultatų faile *Rezultatai.txt* turi būti įrašyta:

- už kokią sumą suvartota energijos kiekviename kambaryje, nurodant kambario numerį ir pinigų sumą dviejų ženklų po kablelio tikslumu;
- kokią pinigų sumą už sunaudotą elektros energiją turi sumokėti viso bendrabučio gyventojai;
- kokią pinigų sumą už elektros energiją turės sumokėti vieno kambario gyventojai.

<i>Duomenys.txt</i>	<i>Rezultatai.txt</i>
5 0.5 50 60 55 70 5	1. 25.00 Lt. 2. 30.00 Lt. 3. 27.50 Lt. 4. 35.00 Lt. 5. 2.50 Lt. Iš viso sunaudota: už 120.00 Lt. Vieno kambario gyventojai mokės: 24.00 Lt.
8 0.5 40 30 25.3 6 11 0 70 80	1. 20.00 Lt. 2. 15.00 Lt. 3. 12.65 Lt. 4. 3.00 Lt. 5. 5.50 Lt. 6. 0.00 Lt. 7. 35.00 Lt. 8. 40.00 Lt. Iš viso sunaudota: už 131.15 Lt. Vieno kambario gyventojai mokės: 16.39 Lt.

2. *Piramidės*. Iš rutuliukų dėliojamos tokios piramidės, kaip parodyta paveikslėlyje:



Piramidės numeris nurodo, kiek laiptelių ji turi. Kiekvienos piramidės sekantis laiptelis turi vienu rutuliuku daugiau, nei prieš tai buvęs. Parašykite programą, kuri tekstiniame faile *Rezultatai.txt* išspausdintų tokią seką: kiekvienas i -asis sekos narys yra i -osios piramidės rutuliukų suma. Kintamasis n nurodo, kiek sekos narių iš viso reikia išspausdinti. Jį reikia perskaityti iš failo *Duomenys.txt*.

<i>Duomenys.txt</i>	<i>Rezultatai.txt</i>
4	1 3 6 10
10	1 3 6 10 15 21 28 36 45 55

3. Viščiukai. Turguje registruojami parduodami viščiukai. Duomenys surašyti faile *Duomenys.txt*, kurio pirmojoje eilutėje yra parduodamų viščiukų skaičius, o kitose eilutėse – masė ir amžius (savaitėmis).

Parašykite programą, kuri surastų ir rezultatų faile *Rezultatai.txt* išspausdintų pradinius viščiukų duomenis, didžiausios masės viščiuko numerį ir jauniausio viščiuko numerį.

<i>Duomenys.txt</i>	<i>Rezultatai.txt</i>
4 42 5 47 6 40 5 35 4	Pradiniai duomenys: Viščiukas Nr.1: masė: 42 amžius: 5 Viščiukas Nr.2: masė: 47 amžius: 6 Viščiukas Nr.3: masė: 40 amžius: 5 Viščiukas Nr.4: masė: 35 amžius: 4 Rezultatai: Didžiausias viščiukas: Nr.2 Jauniausias viščiukas: Nr.4

4. Žaidimo kauliukas. Aurelija mėtė žaidimo kauliuką (su akutėmis nuo 1 iki 6) ir į duomenų failo *Duomenys.txt* pirmąją eilutę įrašė metimų skaičių, o į kitas eilutes rašė išmestų akučių skaičių. Parašykite programą, kuri suskaičiuotų, kiek kartų buvo išmestos 1, 2, 3, 4, 5, 6 akutės. Rezultatą išveskite rezultatų faile *Rezultatai.txt*.

<i>Duomenys.txt</i>	<i>Rezultatai.txt</i>
8 6 4 5 6 2 4 6 1	1 akutė - 1 2 akutės - 1 3 akutės - 0 4 akutės - 2 5 akutės - 1 6 akutės - 3

5. Vėjo jėgainė. Pradinių duomenų failo *Duomenys.txt* pirmoje eilutėje įrašytas mėnesio dienų d skaičius, kai buvo atlikti vėjo krypties ir greičio matavimai. Likusiose d eilučių įrašyta po du skaičius, atskirtus tarpeliais. Pirmasis skaičius yra sveikasis ir nurodo tą dieną vyravusią vėjo kryptį (1 – šiaurė, 2 – pietūs, 3 – rytai, 4 – vakarai), antrasis skaičius yra realusis ir nurodo tos dienos vidutinį vėjo greitį. Parašykite programą, kuri rezultatų failo *Rezultatai.txt* pirmoje eilutėje įrašytų keturis tarpeliais vienas nuo kito atskirtus skaičius, kurie reiškia, kiek kartų iš viso pūtė kiekvienos krypties vėjas. Antroje failo eilutėje turi būti įrašytas žodis TAIP, jei galima statyti vėjo jėgainę. Sąlygos vėjo jėgainei: ne mažiau kaip 75 procentus visų dienų vėjo greitis buvo ne mažesnis kaip 5.0 m / s.

<i>Duomenys.txt</i>	<i>Rezultatai.txt</i>
10 1 4.3 1 5.2 2 4.3 3 4.1 2 5.0 4 5.1 3 4.2 4 4.5 2 4.5 4 5.2	2 3 2 3 NE

6. Ciklas cikle

1. Skaičių dalyba. Parašykite programą, kuri iš duoto skaičių intervalo $[a;b]$ imtų iš eilės po vieną skaičių ir surastų, kiek daliklių turi kiekvienas skaičius, neskaitant vieneto ir paties skaičiaus. Pavyzdžiui, turime skaičių 6. Jis be liekanos dalijasi iš 3 ir 2, t.y. jis dalijasi iš 2 skaičių.

Duomenys	Rezultatai
Įveskite intervalo pradžią ir pabaigą: 2; 5	Skaičius Dalijasi iš: 2 0 3 0 4 1 5 0
Įveskite intervalo pradžią ir pabaigą: 15; 19	Skaičius Dalijasi iš: 15 2 16 2 17 0 18 3 19 0

2. Uždavinių sprendimo varžybos. Mokiniai, ruošdamiesi matematikos egzaminui, sugalvoja suorganizuoti savaitės trukmės uždavinių sprendimo varžybas. Visi mokiniai sprendė skirtingus uždavinius. Reikia apskaičiuoti, kiek uždavinių išsprendė kiekvienas mokinys per savaitę ir kiek uždavinių iš viso mokiniai išsprendė per savaitę.

Duomenų failo *Duomenys.txt* pirmoje eilutėje įrašytas varžybose dalyvavusių mokinių skaičius n . Tolesnėse eilutėse surašyti skaičiai tokia tvarka: pirmas skaičius d rodo, kelias dienas mokinys sprendė uždavinius, o likusieji d skaičiai – kiek uždavinių išsprendė kiekvieną dieną. Atsakymai turi būti surašyti į rezultatų failą *Rezultatai.txt*.

<i>Duomenys.txt</i>	<i>Paaiškinimai</i>
5 5 3 2 3 1 2 3 6 2 4 4 2 2 1 2 3 3 3 3 2 3 4	Moksleivių skaičius Pirmas moksleivis dirbo 5 dienas; toliau yra kiekvienos dienos uždavinių sk. Antras moksleivis dirbo 3 dienas; toliau yra kiekvienos dienos uždavinių sk. Trečias moksleivis dirbo 4 dienas; toliau yra kiekvienos dienos uždavinių sk. Ketvirtas moksleivis dirbo 3 dienas; toliau yra kiekvienos dienos uždavinių sk. Penktas moksleivis dirbo 2 dienas; toliau yra kiekvienos dienos uždavinių sk.
<i>Rezultatai.txt</i>	<i>Paaiškinimai</i>
11 12 7 9 7 46	Pirmas moksleivis išsprendė 11 uždavinių. Antras moksleivis išsprendė 12 uždavinių. Trečias moksleivis išsprendė 7 uždavinius. Ketvirtas moksleivis išsprendė 9 uždavinius. Penktas moksleivis išsprendė 7 uždavinius. Visi moksleiviai išsprendė 46 uždavinius.

3. Automobilių lenktynės. Vyksta žiedinės automobilių lenktynės. Reikia išsiaiškinti santykinai mažiausiai degalų sunaudojusį sportininką (sunaudotų degalų kiekis / nuvažiuotų ratų kiekis). Duomenys apie lenktynes įrašyti tekstiniam failui *Duomenys.txt*. Pirmoje eilutėje įrašytas lenktynininkų skaičius n . Tolesnėse n failo eilučių įrašytas sportininko numeris, jo nuvažiuotų ratų skaičius ir išvardyta, kiek kiekviename rate jis sunaudavo degalų.

Parašykite programą, randančią, kuris lenktynininkas sunaudavo mažiausiai degalų.

Pavyzdžiai:

Duomenys	Rezultatai
3 1 4 2 3 2 4 2 2 4 5 3 4 4 4 5 4	Mažiausiai degalų sunaudavo 1 lenktynininkas
2 1 5 7 8 8 7 9 2 4 7 7 8 6	Mažiausiai degalų sunaudavo 2 lenktynininkas

4. Kauliukai. n draugų sugalvojo pažaisti kauliukais. Kiekvienas žaidėjas meta kauliuką vieną kartą, jei iškritęs akučių skaičius yra lyginis, jis ridena dar kartą, taip ridendamas, kol iškrenta nelyginis akučių skaičius. Surinkti taškai sumuojasi. Parašykite programą, skaičiuojančią, kuris žaidėjas surinko daugiausia taškų.

Pirmoje duomenų failo *Duomenys.txt* eilutėje pateikiamas žaidėjų skaičius n . Tolesnėse n eilučių pateikiamas kauliukų metimų skaičius ir ridenant kauliuką atsivertę akučių skaičiai.

<i>Duomenys</i>	<i>Paaiškinimai</i>
3 2 4 5 1 9 4 4 2 2 7	Žaidėjų skaičius Pirmas žaidėjas kauliuką rideno du kartus ir išrideno: 4 ir 5 Antras žaidėjas kauliuką rideno vieną kartą ir išrideno: 9 Trečias žaidėjas kauliuką rideno keturis kartus ir išrideno: 4, 2, 2, 7
<i>Rezultatas</i>	<i>Paaiškinimai</i>
3	Daugiausiai taškų surinko 3 žaidėjas.

5. Beveik tobulieji skaičiai. Tobuluoju skaičiumi vadinamas toks skaičius, kurio daliklių, mažesnių už patį skaičių, suma lygi jam pačiam. Pvz.: toks yra 6, nes $1+2+3=6$. (Skaičius 6 be liekanos dalijasi iš 1, 2 ir 3). Tokių skaičių nėra daug. Vien intervale $[1;10000]$ tokių skaičių tėra 4. Jūsų užduotis - parašyti programą, kuri intervale $[a;b]$ rastų tobuluosius skaičius tam tikru tikslumu c . Tokiu atveju tobuluoju bus laikomas toks skaičius s , kurio daliklių, mažesnių už jį patį, suma bus intervale $[s-c;s+c]$. Pradiniai duomenys, intervalo rėžiai a ir b bei tikslumas c , įrašyti tekstiniame faile *Duomenys.txt*. Rezultatai spausdinami tekstiniame faile *Rezultatai.txt*. Jūsų programa papildomai turi:

- išspausdinti, kiek pirminių skaičių (tokių, kurie turi tik du daliklius: 1 ir save patį) yra intervale $[a;b]$;
- patikrinti intervalo rėžius (a negali būti daugiau už b). Tokiu atveju - programa nevykdoma;
- spausdinant prie kiekvieno skaičiaus nurodyti jo daliklių kiekį ir tikruosius tobuluosius skaičius pažymėti dviem žvaigždutėmis (**).

<i>Duomenys</i>	<i>Rezultatai</i>
6 100 1 Paaiškinimas: (Intervalas $[6;100]$ ir tikslumas $c = 1$)	6**; Suma: (6); Daliklių skaičius: 3 8*; Suma: (7); Daliklių skaičius: 3 16*; Suma: (15); Daliklių skaičius: 4 28**; Suma: (28); Daliklių skaičius: 5 32*; Suma: (31); Daliklių skaičius: 5 64*; Suma: (63); Daliklių skaičius: 6 Pirminių skaičių kiekis intervale: 22

<pre>5 50 5 Paaiškinimas: (Intervalas [5;50] ir tikslumas c = 5)</pre>	<pre>5; Suma: (1); Daliklių skaičius: 1 6**; Suma: (6); Daliklių skaičius: 3 8; Suma: (7); Daliklių skaičius: 3 9; Suma: (4); Daliklių skaičius: 2 10; Suma: (8); Daliklių skaičius: 3 12; Suma: (16); Daliklių skaičius: 5 14; Suma: (10); Daliklių skaičius: 3 16; Suma: (15); Daliklių skaičius: 4 18; Suma: (21); Daliklių skaičius: 5 20; Suma: (22); Daliklių skaičius: 5 28**; Suma: (28); Daliklių skaičius: 5 32; Suma: (31); Daliklių skaičius: 5 44; Suma: (40); Daliklių skaičius: 5 Pirminių skaičių kiekis intervale: 13</pre>
<pre>200 199 10 Netinkami rėžiai (a > b)</pre>	<pre>Netinkami rėžiai.</pre>

7. Šakotieji ir cikliniai algoritmai

1. Temperatūra. Duomenų faile *Duomenys.txt* stulpeliu surašyti kiekvienos dienos temperatūros matavimai. Duomenų kiekis nėra nurodytas. Parašykite programą, kuri rezultatų faile *Rezultatai.txt* įrašytų:

1. dviejų gretimų dienų temperatūros pokytį,
2. rastų dienų skaičių (kiek matavimų duomenų faile),
3. didžiausią temperatūros pokytį tarp dviejų dienų absoliutine verte (nesvarbu, ar atšilo, ar atvėso).

Temperatūra pateikiama dešimtųjų tikslumu.

<i>Duomenys.txt</i>	<i>Rezultatai.txt</i>
<pre>15.1 16.3 14.0 13.3 17.5 17.6 17.6 16.7</pre>	<pre>15.1 (+1.2) 16.3 (-2.3) 14.0 (-0.7) 13.3 (+4.2) 17.5 (+0.1) 17.6 (+0.0) 17.6 (-0.9) 16.7 ----- Dienų skaičius: 8 Didžiausias temperatūros pokytis: +4.2</pre>

-0.9	-0.9
0.2	(+1.1)
0.0	0.2
0.3	(-0.2)
1.0	0.0
2.3	(+0.3)
-0.5	0.3
-0.5	(+0.7)
-0.6	1.0
-1.4	(+1.3)
0.0	2.3
-1.5	(-2.8)
	-0.5
	(+0.0)
	-0.5
	(-0.1)
	-0.6
	(-0.8)
	-1.4
	(+1.4)
	0.0
	(-1.5)
	-1.5

	Dienų skaičius: 12
	Didžiausias temperatūros pokytis: -
	2.8

2. LEGO robotas. Linas nusprendė 6 mėnesius taupyti pinigus LEGO robotui, kuris kainuoja a Lt. Kiekvieną mėnesį jis skirtingą kartų kiekį n į taupyklę mesdavo pinigus. Duomenų faile *Duomenys.txt* yra 6 eilutės, kuriose pirmasis skaičius nurodo, kiek kartų n tą mėnesį Linas į taupyklę įmetė pinigų, o kiti skaičiai toje eilutėje nurodo, kiek pinigų m (litais) Linas įmetė. Duomenų failo paskutiniojoje (septintoje) eilutėje yra nurodyta norimo LEGO roboto kaina a (litais). Parašykite programą, kuri suskaičiuotų, kiek Linas per šešis mėnesius sutaupė pinigų ir ar jį užteks nusipirkti norimą LEGO robotą, o rezultatus įrašytų į rezultatų failą *Rezultatai.txt*.

Duomenys	Paaiškinimai
4 6 1.7 4.3 5	Pirmą mėnesį Linas į taupyklę įmetė 4 kartus (6 Lt, 8.7 Lt, 4.3 Lt, 5 Lt)
6 7 5.5 6.2 6.1 2.2 6	Antrą mėnesį Linas įmetė 6 kartus (7 Lt, 5.5 Lt, 6.2 Lt, 6.1 Lt, 2.2 Lt, 6 Lt)
3 8.56 3.33 7	Trečią mėnesį Linas įmetė 3 kartus (...)
5 6 8 3.3 6 1.4	Ketvirtą mėnesį Linas įmetė 5 kartus (...)
5 7 9.2 2.5 6 5	Penktą mėnesį Linas įmetė 5 kartus (...)
6 7 3.3 5 2 5 7	Šeštą mėnesį Linas įmetė 6 kartus (...)
150.99	Norimas „Lego“ robotas kainuoja 150.99 Lt
Rezultatai	Paaiškinimai
152.59	Linas per 6 mėn. sutaupė 152.59 Lt
Užteks	Linui užteks sutaupytų pinigų nusipirkti norimą „Lego“ robotą.

3. Keliautojai. Draugai sutarė keliauti į Latviją praleisti atostogų. Kiekvienas susikrovė po kuprinę ar lagaminą ir atvyko į autobusų stotį. Už vieną kilogramą bagažo reikia mokėti b litų bagažo mokestį. Duomenų failo *Duomenys.txt* pirmoje eilutėje parašyta, kiek draugų n važiuos į Latviją, antroje eilutėje surašyta, kiek $m_1, m_2 \dots m_n$ kg svėrė kiekvieno draugo kuprinė arba lagaminas, trečioje – kiek litų b reikia mokėti už vieną kilogramą bagažo. Parašykite programą, kuri suskaičiuotų ir į rezultatų failą *Rezultatai.txt* įrašytų, kiek kilogramų bagažo kb vežasi visi draugai ir kokią sumą s jie turės sumokėti už bagažą.

Duomenys	Paiškinimai
5 5.5 3.6 4.8 5.1 6 2	Važiavo 5 draugai Kiekvieno draugo bagažo svoris Bagažo mokestis už vieną kilogramą bagažo
Rezultatai	Paiškinimai
25 kg 50 Lt	Visi draugai vežasi 25 kg bagažo Daugams reiks sumokėti 50 Lt už bagažą

4. Lenktynės. Lenktynėse dalyvavo n pilotų. Apie kiekvieną pilotą žinoma tokia informacija: jo vidutinis greitis, bei tai, ar finišavo, ar ne. Sudarykite sąrašą pilotų, kurių greitis buvo didesnis už v . Jei pilotas nefinišavo, jo greitis turi būti sumažinamas 20%.

Duomenys	Paiškinimai
3 45 50 1 60 1 45 0	Dalyvavo trys pilotai, siektinas greitis (v) 45 Piloto greitis 50 , finišavo. (1 - finišavo, 0 - nefinišavo) Piloto greitis 60, finišavo Piloto greitis 45, nefinišavo, todėl jo greitis 36
Rezultatai	Paiškinimai
50 60	Du pilotai viršijo duotąjį greitį.
Duomenys	Paiškinimai
2 25 30 1 20 1	Dalyvavo du pilotai, siektinas greitis (v) 25 Piloto greitis 30 , finišavo. (1 - finišavo, 0 - nefinišavo) Piloto greitis 20, finišavo
Rezultatai	Paiškinimai
30	Vienas pilotas viršijo duotąjį greitį.

5. Kelių eismo taisyklės. Vairavimo mokyklai kiekvienos vasaros pradžioje iškyla problema, kad ateina labai daug mokinių, todėl reikia sukurti programą, padedančią paskirstyti srautus vairavimo instruktoriams. Mokykloje iš viso yra trys vairavimo instruktoriai. Pradinių duomenų faile *Duomenys.txt* faile surašyta informacija apie mokinius. Pirmoje eilutėje įrašytas mokinių skaičius. Likusiose eilutėse įrašyta: mokinio numeris ir jo mokėjimo vairuoti balas 10-ies balų sistemoje.

Į failą *Rezultatai.txt* reikia surašyti kiekvienam instruktoriui jo mokinių numerius, bet mokinius reikia paskirstyti taip, kad pirmasis instruktorius dirbtų tik su 1-3 balus, antrasis – 4-6 balus, o trečiasis – 7-10 balų surinkusiais mokiniais.

<i>Duomenys.txt</i>	<i>Rezultatai.txt</i>
6	Pirmasis: 5 6
1 4	Antrasis: 1 2
2 6	Trečiasis: 3 4
3 7	
4 9	
5 1	
6 3	

8. Funkcija, gražinanti apskaičiuotą reikšmę per funkcijos vardą

1. Skaitymas. Mokinys perskaitė n puslapių knygą. Pirmąją dieną jis perskaitė a puslapių. Kiekvieną sekančią dieną mokinys perskaitydavo b puslapių daugiau, negu prieš tai buvusią dieną. Per kelias dienas mokinys perskaitė knygą?

Pradinių duomenų faile *Duomenys.txt* įrašytas pradiniai duomenys n , a ir b .

Rezultatų faile *Rezultatai.txt* turi būti įrašytas dienų skaičius.

Reikalavimai programai:

- sukurkite funkciją, skaičiuojančią per kelias dienas mokinys perskaitys knygą.

Pradiniai duomenys	Rezultatai
120 20 3	5
132 20 3	6
200 15 5	7

2. Plytelės. Sukurkite programą, kuri grindų plytelių klojėjui padėtų apskaičiuoti, kiek reikės stačiakampių plytelių, norint pilnai iškloti stačiakampio ploto formos grindis. Jei plytelę reikia nupjauti, nupjautoji dalis grindims kloti nebenaudojama.

Pradinių duomenų faile *Duomenys.txt* įrašyti plytelės matmenys centimetrais (x – plytelės ilgis, y – plytelės plotis) ir grindų matmenys metrais (s_x – grindų ilgis, s_y – grindų plotis).

Rezultatų faile *Rezultatai.txt* reikia išvesti plytelių skaičių.

Reikalavimai programai:

- sukurkite funkciją, skaičiuojančią, kiek plytelių reikės grindims iškloti.

Pradinių duomenų pavyzdžiai	Rezultatai
30 30 3 6	200
40 50 2 3	30
40 50 3 3	48

3. Trikampio plotas. Stačiakampėje koordinačių plokštumoje nubraižytas trikampis, kurio viršūnių A, B ir C koordinatės žinomos. Parenkite programą, skaičiuojančią trikampio plotą.

Pradinių duomenų failo *Duomenys.txt* trijose eilutėse įrašyta po du sveikuosius skaičius – kiekvienos trikampio viršūnės koordinatės x ir y .

Rezultatų faile *Rezultatai.txt* reikia pateikti trikampio plotą šimtųjų tikslumu.

Reikalavimai programai:

1. sukurkite funkciją, skaičiuojančią trikampio kraštinės ilgį (atstumą tarp dviejų taškų);
2. sukurkite funkciją, skaičiuojančią trikampio plotą.

Pradiniai duomenys	Rezultatai
10 20 20 -5 -5 -6	317.50

4. Laimingi bilietai. Autobusų parko administracija nusprendė keleiviams, kurių bilietai laimingi, dovanoti dovanas. Autobuso bilietas laikomas laimingu, jei jo numerio skaičius ir jam atbulas skaičius sutampa (pvz., 12321). Bilietai numeruojami nuo n iki m (didžiausia m reikšmė ne didesnė už 1000000). Parašykite programą, kuri suskaičiuotų, kiek bus laimingų bilietų.

Pradinių duomenų faile *Duomenys.txt* įrašytos n ir m reikšmės. Rezultatų faile *Rezultatai.txt* reikia įrašyti laimingų bilietų skaičių.

Reikalavimai programai:

1. sukurkite funkciją, formuojančią natūrinio skaičiaus atbulą skaičių.

Pradiniai duomenys	Rezultatai
120 142	3

5. Taupymas. Jonas buvo geras technikas ir kartais atlikdavo naujų kompiuterių elektros sunaudojimo testus. Jis apibendrintus testų duomenis surašė į failą, kad Petras priimtų sprendimą, ar verta visus organizacijos kompiuterius pakeisti naujais. Kadangi Petras yra labai užimtas žmogus, todėl paprašė padėti jam parašyti programą, kuri apskaičiuotų, kiek būtų sutaupyta/prarasta pinigų mokant už elektrą per nurodytą laikotarpį, kad būtų galima priimti sprendimą pakeisti kompiuterius.

Pradinių duomenų faile *Testai.txt* įrašyta:

- pirmoje eilutėje įrašyta vienos laiko zonos elektros energijos kaina Lt/kWh;
- antroje eilutėje įrašyti keturi skaičiai: senųjų kompiuterių minimali naudojama galia vatais, įprastai dirbant naudojama galia vatais, maksimalia apkrova dirbančio kompiuterio naudojama galia vatais, bei kompiuterio likutinė vertė litais;
- trečioje eilutėje įrašyti keturi skaičiai: naujų kompiuterių minimali naudojama galia vatais, įprastai dirbant naudojama galia vatais ir maksimalia apkrova dirbančio kompiuterio naudojama galia vatais, bei naujo kompiuterio kaina litais;
- ketvirtoje eilutėje įrašytas skaičius rodo, kelis kartus naujas kompiuteris dirba sparčiau už seną (skaičiuojant CPU greitį);
- penktoje eilutėje yra įrašas, kelis metus dar planuojama naudoti seną arba naują kompiuterį.

Rezultatų faile *Patarimas.txt* turi būti įrašyta:

- pirmoje eilutėje – kelis kartus naujas kompiuteris dirba sparčiau už seną kompiuterį;
- antroje eilutėje – kokią pinigų sumą vidutiniškai sutaupys naujas kompiuteris per planuojamą naudoti laikotarpį;
- trečioje eilutėje – kokią papildomą pinigų sumą kainuos naujas kompiuteris;
- ketvirtoje eilutėje – kiek vertas senas kompiuteris.

Reikalavimai programai:

1. sukurkite funkciją, gražinančią seno ir naujo kompiuterių elektros energijos sąnaudų skirtumą litais.

Sprendžiant uždavinį galima daryti prielaidą, kad kompiuteriai yra serveriai ir niekada neišjungiami. $(120\text{ W} - 42\text{ W}) * 0.001\text{ (į kW)} * 4 * 365 * 24\text{ (į kWh)} * 0.487 = 1331,03\text{ Lt}$

Duomenys	Rezultatai
0.487	7.2
40 42 70 1200	1331.03
100 120 170 400	1200
7.2	400
4	

6. Mobilusis telefonas išsimokėtinai. Sukurkite programą, kuri apskaičiuotų, keliais procentais pabrangs išsimokėtinai perkamas mobilusis telefonas. Kiekvieno vėlesnio mėnesio palūkanos litais skaičiuojamos nuo likusios neišmokėtos telefono kainos. Pvz., jei telefono kaina - 600 Lt, o išsimokėjimo laikotarpis - 12 mėn., tada pirmąjį mėnesį palūkanos skaičiuojamos nuo 600 Lt, antrąjį nuo 550 Lt ir t.t. Be to, sudarant išsimokėjimo sutartį, yra skaičiuojamas fiksuotas sutarties mokestis. Pradinių duomenų faile *Duomenys.txt* įrašyta: telefono pradinė kaina K , mėnesinė palūkanų norma N , išsimokėjimo už telefoną laikotarpis mėnesiais L , sutarties mokestis M .

Rezultatų faile *Rezultatai.txt* reikia pateikti telefono pabrangimo kainą procentais.

Reikalavimai programai:

1. sukurkite funkciją, skaičiuojančią, kiek iš viso pabrangs telefonas per laikotarpį L ;
2. sukurkite funkciją, skaičiuojančią keliais procentais pabrangs telefonas.

Duomenys	Rezultatai
500.0 0.2 3 20.0	4.4
500.0 0.15 12 20.0	5
700.0 0.25 24 25.0	6.7

7. **Kelionė.** n asmenų – a mokinių ir b mokytojų – planuoja vykti į kaimo turizmo sodybą „Nemunėlis“. Vienos paros šioje sodyboje kainą galima apskaičiuoti remiantis žemiau pateikta informacija:

Žmonių skaičius: mažiau nei 10; nuo 10 iki 19; nuo 20 iki 30.	Kaina vienam žmogui x Lt y Lt z Lt
--	---

Parašykite programą, skaičiuojančią:

- kiek kainuotų viena para sodyboje n žmonių grupei;
- kiek kainuotų viena para šioje sodyboje žmonių grupei, jei kartu su n asmenų grupe vyktų keturi vaikų namų auklėtiniai;
- po kiek litų už vieną parą šioje sodyboje turėtų sumokėti kiekvienas n asmenų grupės narys, jeigu jie nutartų sumokėti ne tik už save, bet ir už keturis vaikų globos namų auklėtinius.

Pradinių duomenų failo *Duomenys.txt* pirmoje eilutėje įrašyti trys skaičiai: n , a ir b , antroje eilutėje įrašytos kainos, kurias mokėtų turistai – x , y ir z .

Rezultatų faile *Rezultatai.txt* turi būti įrašyta:

- pirmoje failo eilutėje – kiek kainuos para sodyboje visai n žmonių grupei, kiek mokėtų a mokiniai ir kiek b mokytojų, jeigu jie vyktų atskirai;
- antroje eilutėje – kiek kainuos para sodyboje visai n žmonių grupei ir keturiems vaikų globos namų auklėtiniais;
- trečioje eilutėje – po kiek litų už vieną parą šioje sodyboje mokės kiekvienas n asmenų grupės narys, už save ir už keturis vaikų globos namų auklėtinius.

Reikalavimai programai:

1. sukurti funkciją, skaičiuojančią kelionės kainą.

Pradiniai duomenys	Rezultatai
9 7 2 55 50 45	495.00 385.00 110.00 650.00 72.22

8. Kodavimas. Draugės Aistė ir Valdonė sugalvojo susirašinėti specialiu formatu. Joms buvo svarbiausia tik tai, kad kiti nesuprastų apie ką jos susirašinėja. Draugės susitarė, kad kiekvienas simbolis, kurio joms prireiks, bus atvaizduotas skaičiumi, o skaičiai vienas nuo kito bus atskirti tarpais. Skaičiaus susiejimą su simboliu jos pavadino AV (Aistė Valdonė) koduote ir sukūrė atitikmenų lentelę, kad prisimintų koks skaičius kokį simbolių atitinka.

Simbolis	Skaičius	Simbolis	Skaičius	Simbolis	Skaičius	Simbolis	Skaičius
32	tarpas	65	A	82	R	105	i
33	!	66	B	83	S	106	j
44	,	67	C	84	T	107	k
46	.	68	D	85	U	108	l
48	0	69	E	86	V	109	m
49	1	70	F	87	W	110	n
50	2	71	G	88	X	111	o
51	3	72	H	89	Y	112	p
52	4	73	I	90	Z	113	q
53	5	74	J	97	a	114	r
54	6	75	K	98	b	115	s
55	7	76	L	99	c	116	t
56	8	77	M	100	d	117	u
57	9	78	N	101	e	118	v
63	?	79	O	102	f	119	w
		80	P	103	g	120	x
		81	Q	104	h	121	y
						122	z

Aistė parašė laišką Valdonei ir nori įsitikinti, ar laiške yra tik leistini simboliai. Kadangi Aistei sudėtinga pamatyti savo klaidas, tad reikia padėti Aistei ir parašyti programą, kuri patikrintų, ar laiške visi skaičiai turi AV koduotės atitikmenis.

Laiškas, kurį reikia patikrinti, įrašytas duomenų faile *Valdonei.txt*. Atsakymą įrašyti faile *Tikrinimo_rezultatai.txt*. Jeigu laiškas yra teisingas, faile turi būti įrašytas žodis *Teisingai*, kitu atveju kiekvienam skaičiui, neturinčiam atitikmens AV koduotėje, spausdinti *Neteisingas simbolis*. Vienam neteisingam skaičiui skiriama viena eilutė.

Reikalavimai programai:

sukurkite funkciją, kuri grąžina reikšmę true, jei skaičius turi atitikmenį AV koduotėje, arba reikšmę false, jei skaičius tokio atitikmens neturi.

<i>Valdonei.txt</i>	<i>Tikrinimo_rezultatai.txt</i>
65 114 32 110 111 114 105 32 114 121 116 111 106 32 115 117 115 105 116 105 107 116 105 32 107 105 101 109 101 32 49 53 32 104 63	Teisingai
1	Neteisingas simbolis
1 2	Neteisingas simbolis Neteisingas simbolis

9. Funkcija su parametrais-nuorodomis

1. Bilietų kaina. Dramos teatro salės planas yra aprašomas vienetukų (1) ir nuliukų (0) lentele. 1 – reiškia, kad salės vieta yra užimta, 0 – neužimta. Pirmą lentelės eilutę atitinka pirmą teatro salės eilę, antra – antrą ir t. t. Bilietai pirmose dvejose eilėse kainuoja po 100 Lt, trečioje ir ketvirtoje – po 70 Lt, o likusiose – po 40 Lt. Sukurkite programą, kuri apskaičiuotų spektaklyje už parduotus bilietus surinktą pinigų sumą ir vidutinę bilietų kainą.

Pradinių duomenų faile *Duomenys.txt* yra duotas salės užimtumo planas. Pirmoje eilutėje įrašyta, kiek salėje yra eilių ir kiekvienoje eilėje esančių vietų skaičius.

Rezultatų faile *Rezultatai.txt* reikia išvesti spektaklyje už parduotus bilietus surinktą pinigų sumą ir vidutinę bilietų kainą.

Reikalavimai programai:

1. Sukurti funkciją apskaičiuojančią vienos vietos kainą, priklausomai nuo eilės.
2. Sukurti funkciją void (procedūrą), apskaičiuojančią gautą pinigų sumą už parduotus bilietus ir vidutinę bilietų kainą.

Duomenys	Rezultatai
6 10 0 1 0 1 0 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 1 1 0 0 1 1 1 1 0 1 0 1 1 0 1 1 1 1 1 0 1 1 1 0 1 1 1 0 1 1 0 1 1	3080 70.00
3 11 0 1 0 1 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1 1 1 1 1 0 1	1330 83.12

2. Kelionė. Turistas nusprendė aplankyti n ($1 \leq n \leq 10$) miestų juose nesustodamas pailsėti. Iš pradinio taško jis išvyko *val* valandų ir *min* minučių. Yra žinomi atstumai tarp miestų s_i kilometrais ir vidutinis turistų greitis vi km/val kiekviename etape. Apskaičiuokite, kuriuo paros laiku (valandomis ir minutėmis) turistą pasieks kiekvieną miestą. Turistas gali keliauti ne vieną parą. Visi duomenys yra sveikieji skaičiai.

Pradinių duomenų failo *Duomenys.txt* pirmoje eilutėje užrašyta išvykimo laikas valandomis ir minutėmis; antroje eilutėje užrašytas aplankytų miestų skaičius n ; likusiose n eilučių nurodyta po du sveikuosius skaičius: atstumai tarp miestų A_{i-1} ir A_i bei vidutinis turistų greitis km/val tame etape.

Rezultatų faile *Rezultatai.txt* reikia pateikti miesto numerį ir kuriuo laiku (valandos, minutės) jis bus tame mieste.

Reikalavimai programai:

1. sukurti funkciją void C++ (procedūrą Pascal), skaičiuojančią, kuriuo laiku (valandos, minutės) turistas atvyksta į konkretų miestą.

Duomenys	Rezultatai
15 35 5 100 50 50 65 150 70 120 55 110 50	1 17 35 2 18 21 3 20 29 4 22 39 5 0 51

3. Lyginumas. Įterptinių sistemų programuotojas sugalvojo parašyti programą, patikrinančią duomenų perdavimą bevieliu būdu, kad galėtų žinoti, ar duomenys buvo persiųsti be klaidų. Jei duomenys persiunčiami su klaidomis, tada reikia siųsti dar kartą. Duomenys siunčiami iš A į B, B vietoje jie yra patikrinami. Kad būtų įmanoma patikrinti, susitarta, kad prie siunčiamų duomenų bus pridomas papildomas skaičius, kuris parodo, ar visų duomenų skaičių suma yra lyginis skaičius, ar nelyginis (paskutiniojo skaičiaus reikšmė yra lygi 0, kai suma yra lyginis; 1 - kai suma nelyginis skaičius).

Patikrinus atsiųstus duomenis ir nustatčius, kad duomenys buvo nepažeisti keliaujant bevieliu terpe, reikia išvesti „Duomenys atsiųsti sėkmingai“, o nustatčius, kad duomenys buvo pažeisti, arba jeigu specialiojo papildomo skaičiaus reikšmė nelygi 1 arba 0, išvesti pranešimą „Duomenų atsiųsti be klaidų nepavyko, reikia pakartoti siuntimą“.

Duomenys	Duomenų paaiškinimas	Rezultatai
2 11 10 1	2 - realių duomenų kiekis 11 ir 10 - perduodami realūs duomenys 1 - papildomas specialusis skaičius	Duomenys atsiųsti sėkmingai
2 11 21 0	2 - realių duomenų kiekis 11 ir 21 - perduodami realūs duomenys 0 - papildomas specialusis skaičius	Duomenys atsiųsti sėkmingai
1 11 0	1 - realių duomenų kiekis 11 - perduodamas realus duomuo 0 - papildomas specialusis skaičius	Duomenų atsiųsti be klaidų nepavyko, reikia pakartoti siuntimą

Reikalavimas programai:

1. Privalo būti bent viena funkcija, kurioje bent vienas parametras būtų nuoroda. Pavyzdžiui, gali būti tokia funkcija:

a. C++: void skaityti(ifstream & in, int & n, int & duomenų_lyginumas, int & lyginumas);

b. Pascal: procedure skaityti(var ivedimas : text; var n : integer; var duomenų_lyginumas : integer; var lyginumas : integer);

4. Gamyba. Keturiuose cecho baruose gaminami dviejų rūšių gaminiai. Reikia sudaryti tokį gamybos planą, kad atidavus pagamintus gaminius, būtų gautas didžiausias pelnas. Pirmame bare dirbama ne daugiau $h1$ valandų, antrame – $h2$ valandų, trečiame – $h3$ valandų ir ketvirtame – $h4$ valandų. Lentelėje nurodyta, kiek laiko (valandomis) reikia kiekvienos rūšies gaminiui pagaminti atitinkamame bare. Nulis rodo, kad gamins atitinkamame bare negaminamas.

Gaminiai	Baras 1	Baras 2	Baras 3	Baras 4
I	4	3	0	2
II	2	6	4	0
Bare galima dirbti (val.)	16	30	16	12

Atidavęs vieną I rūšies gaminį cechas gauna a Lt pelną, II rūšies – b Lt pelną.

Parašykite programą, kuri apskaičiuotų, kiek barai turi pagaminti I ir II rūšies gaminių, kad įmonė gautų didžiausią pelną.

Duomenų faile *Duomenys.txt*:

- pirmoje eilutėje nurodyta, kokį pelną gauna už I rūšies gaminį ir už II rūšies gaminį;
- antroje eilutėje nurodyta kiek valandų dirba kiekvienas baras;
- trečioje eilutėje I gaminio gaminimo laikas baruose;
- ketvirtoje eilutėje II gaminio gaminimo laikas baruose.

Rezultatų faile *Rezultatai.txt* įrašyti:

- du skaičiai: kiek gaminama I ir II rūšies gaminių.
- įmonės pelnas pagaminus gaminius.

Reikalavimai programai:

1. sukurti procedūrą (funkciją void C++), skaičiuojančią, kiek I ir II rūšies gaminių gamina barai, ir įmonės pelną.

Duomenys	Rezultatai
3 4	2 4
16 30 16 12	pelnas 22 Lt
4 3 0 2	
2 6 4 0	

5. Dieta. Vaida nutarė numesti keletą kilogramų. Ilgai rinkosi dietą. Žiniasklaidoje rado straipsnį apie grikių dietą. Laikantis šios dietos galima valgyti tik grikius. Yra žinoma, kad 100 gramų grikių kruopų yra 69,3 g angliavandenių, 12,6 gramų baltymų ir 3,1 gramų riebalų. Kadangi Vaidai ši dieta patiko dar ir todėl, kad beveik nereikia gaminti, tai ji nusprendė n dienų maitintis tik grikiiais.

Vaida žinodama dietų pasekmes nutarė pasikonsultuoti su dietologe. Dietologė rekomendavo suvartoti ne mažiau 1800 kcal per dieną. Žinoma, kad 100 gramų grikių turi 349 kcal.

Pradinių duomenų faile *Duomenys.txt* įrašytas dienų skaičius n , nurodantis kiek dienų Vaida valgė tik grikius. Antroje eilutėje nurodyta, kiek 100 g grikių yra angliavandenių, baltymų ir

riebalų. Likusiose n eilučių įrašyti sveikieji skaičiai nurodo, kiek gramų grikių suvartojo Vaida per dieną.

Rezultatų faile *Rezultatai.txt* įrašykite:

- kiek gramų angliavandenių, baltymų ir riebalų suvartota kiekvieną dieną;
- kiek buvo dienų, kai Vaida suvartojo reikiamą kiekį kalorijų.

Reikalavimai programai:

1. sukurti procedūrą (funkciją void C++), skaičiuojančią kiek gramų angliavandenių, baltymų ir riebalų suvartota kiekvieną dieną,
2. funkciją, skaičiuojančią, kiek buvo dienų, kai Vaida suvartojo reikiamą kalorijų kiekį.

Duomenys	Rezultatai
5	415.8 75.6 18.6
69.3 12.6 3.1	103.95 18.9 4.65
600	207.9 37.8 9.3
150	277.2 50.4 12.4
300	485.1 88.2 21.7
400	Vaida suvartojo reikiamą kalorijų
700	kiekį 2 dienas

6. Skrajutės. Mokinys vasaros atostogų metu įsidarbino išnešioti skrajutes. Mokinys dirbo pagal grafiką: pirmadieniais – 5 darbo valandas, antradieniais – 6 val., trečiadieniais – 4 val. Tačiau mokinys, kai kuriomis dienomis dirbo daugiau valandų, kai kuriomis mažiau.

Sukurkite programą, kuri apskaičiuotų kiek valandų mokinys dirbo, kiek buvo dienų, kai viršijo valandų kiekį ir kiek dienų, kai nepasiekė darbo grafike numatyto valandų skaičiaus. Šiuos skaičiavimus atlikite kiekvienai savaitei atskirai ir bendrai visoms savaitėms.

Pradinių duomenų faile *Duomenys.txt* pirmoje eilutėje duotas dirbtų savaitių skaičius, kitose eilutėse – kiek valandų mokinys dirbo savaitės pirmadienį, antradienį ir trečiadienį. Vienai savaitei skiriama viena eilutė.

Rezultatų faile *Rezultatai.txt* įrašykite **kiekvienos savaitės** darbo valandų skaičių, dienų skaičių, kai buvo viršytas ir nepasiektas darbo grafike numatytas valandų skaičius.

Paskutinėje eilutėje įrašykite kiek valandų dirbo iš viso, kiek iš viso buvo dienų, kai viršijo ir kai nepasiekė darbo grafike numatyto valandų skaičiaus.

Reikalavimai programai. Reikia sukurti dvi funkcijas:

1. Funkciją, suskaičiuojančią kiek valandų mokinys dirbo, kiek buvo dienų, kai viršijo ir kiek dienų, kai nepasiekė darbo grafike numatyto valandų skaičiaus per vieną savaitę.
2. Funkciją, išvedančią rezultatus į rezultatų failą. Funkcija turi būti tinkama vienos savaitės ir bendriems visų savaitių rezultatams išvesti.

Pradinių duomenų pavyzdys	Rezultatai	Rezultatų paaiškinimai
4 8 4 11 7 9 7 5 6 7 8 9 4	23 1 2 23 2 1 18 1 2 21 2 1 85 6 6	Pirmasis skaičius rodo dirbtų valandų skaičių, antrasis – kiek buvo dienų, kai viršijo numatytą valandų skaičių, trečiasis – kiek buvo dienų, kai nepasiekė numatyto valandų skaičiaus.

7. Paprastoji trupmena. Paprastąją trupmeną sudaro trys dalys: sveikoji dalis, skaitiklis ir vardiklis. Mokiniai mokosi sudėti ir atimti paprastąsias trupmenas. Rengdama savarankiško darbo užduotis, mokytoja parengė pradinių duomenų failą. Pirmoje eilutėje ji nurodė užduočių skaičių n . Tolesnėse n eilučių įrašė po šešis tarpais atskirtus sveikuosius skaičius: pirmosios trupmenos sveikoji dalis, pirmosios trupmenos skaitiklis, pirmosios trupmenos vardiklis, antrosios trupmenos sveikoji dalis, antrosios trupmenos skaitiklis, antrosios trupmenos vardiklis. Jei trupmena sveikosios dalies neturi, tai duomenų faile jos vietoje rašomas nulis. Parenkite programą, kuri apskaičiuotų trupmenų sumą ir skirtumą. Rezultatai: suprastintos trupmenos su išskirtomis sveikosiomis dalimis. Rezultatą pateikite lentelė tekstiniaame faile.

Reikalavimai programai:

1. sukurkite funkcijas void C++ (procedūras Pascal), skaičiuojančias trupmenų sumą ir skirtumą;
2. sukurkite funkciją void C++ (procedūras Pascal) trupmenos prastinimui; sukurkite funkciją void C++ (procedūras Pascal) vienos trupmenos išvedimui.

Duomenys	Rezultatai
3 1 4 5 0 3 4 1 3 5 2 2 9 0 7 8 2 7 8	<pre> ----- T1 T2 Suma Skirtumas ----- 1 4 / 5 3 / 4 2 11 /20 1 1 /20 1 3 / 5 2 2 / 9 3 37 /45 -28 /45 7 / 8 2 7 / 8 3 3 / 4 -2 </pre>

10. Masyvai. Veiksmai su masyvų elementais. Sumos, kiekio, vidurkio skaičiavimas

1. Avalynės parduotuvė. Parduotuvė prekiauja vyriška ir moteriška avalyne. Kiekvieną dieną yra vedama apskaita: užrašoma, kiek batų porų yra parduota ir pasižymima informacija apie parduotą porą – vyriška ar moteriška bei poros kaina. Parašyti programą, kuri rastų:

- už kokią pinigų sumą parduota moteriškos ir vyriškos avalynės atskirai;
- kiek vidutiniškai kainavo moteriškų ir vyriškų batų pora;
- už kokią pinigų sumą parduota avalynės per dieną.

Pradinių duomenų faile *Duomenys.txt* įrašyta:

- pirmoje eilutėje nurodomas per dieną parduotų batų porų skaičius n ($1 \leq n \leq 100$);
- tolesnėse n eilučių įrašyta informacija apie vieną batų porą: simbolis v (vyriški batai) arba m (moteriški batai) bei batų poros kaina litais (realusis skaičius).

Skaičiavimų rezultatus pateikite dviejų skaičių po kablelio tikslumu faile *Rezultatai.txt*.

Reikalavimai programai:

1. duomenims saugoti naudokite masyvus;
2. sukurkite pradinių duomenų skaitymo funkciją void C++ (procedūrą Pascal);
3. sukurkite funkciją void C++ (procedūrą Pascal), randančią vyriškų arba moteriškų porų kainų sumą bei vidutinę poros kainą;
4. sukurkite funkciją, gražinančią reikšmę per savo vardą C++, (funkciją Pascal), apskaičiuojančią bendrą pinigų sumą;
5. sukurkite rezultatų rašymo į failą funkciją void C++ (procedūrą Pascal) .

Duomenys	Rezultatai
5 v 102.56 m 215.70 m 99.21 v 200.00 v 175.99	Moteriška avalynė: per dieną parduota už 314.91 Lt vidutiniškai viena batų pora kainavo 157.45 Lt Vyriška avalynė: per dieną parduota už 478.55 Lt vidutiniškai viena batų pora kainavo 159.52 Lt Iš viso per dieną parduota avalynės už 793.46 Lt
2 m 245.70 m 150.99	Moteriška avalynė: per dieną parduota už 396.69 Lt vidutiniškai viena batų pora kainavo 198.35 Lt Vyriška avalynė: Prekiauta nebuvo Iš viso per dieną parduota avalynės už 396.69 Lt.

2. Moliūgai. Ūkininkas augina moliūgus ir pristato į supirkimo punktą. Supirkimo punkte priimami tik tie moliūgai, kurių svoris yra ne mažesnis už $p1$ kilogramų ir ne didesnis už $p2$ kilogramų. Parašykite programą, kuri apskaičiuotų:

- vidutinį visų moliūgų svorį;
- keli moliūgai bus vežami į supirkimo punktą;
- atrinktų moliūgų vidutinį svorį ir pateiktų vežamų moliūgų svorių sąrašą.

Pradinių duomenų failo *Duomenys.txt* pirmoje eilutėje įrašyti 3 skaičiai. Pirmas skaičius yra sveikasis ir parodo kiek moliūgų n ($1 \leq n \leq 100$) ūkininkas užaugino. Tolesni du realieji skaičiai parodo svorio ribas $p1$ ir $p2$. Antroje pradinių duomenų failo eilutėje yra surašyti kiekvieno moliūgo svoriai kilogramais (realieji skaičiai).

Skaičiavimų rezultatus pateikite trijų skaičių po kablelio tikslumu faile *Rezultatai.txt*.

Reikalavimai programai:

1. duomenims saugoti naudokite masyvus;
2. sukurkite pradinių duomenų skaitymo funkciją void C++ (procedūrą Pascal);
3. sukurkite funkciją void C++ (procedūrą Pascal), atrinkančią tinkamus supirkimui moliūgus į naują masyvą ir randančią tų moliūgų kieki;
4. sukurkite funkciją, gražinančią reikšmę per savo vardą C++, (funkciją Pascal), apskaičiuojančią moliūgų svorių vidurkį;
5. sukurkite rezultatų rašymo į failą funkciją void C++ (procedūrą Pascal) .

Duomenys	Rezultatai
5 2.5 5.0 2.00 4.75 3.50 5.80 3.00	Visų moliūgų vidutinis svoris: 3.810 kg Į supirkimą atrinkta: 3 Vidutinis vieno atrinkto moliūgo svoris: 3.750 kg Atrinktų moliūgų svorių sąrašas: 4.750 3.500 3.000 .
2 2.5 5.0 2.00 6.40	Visų moliūgų vidutinis svoris: 4.200 kg Į supirkimo punktą atrinktų moliūgų nėra.

3. Taupymas. Trys klasės draugai: Simas, Linas ir Mantas dalyvavo mokykloje organizuotoje paskaitoje „Kaip taupyti pinigus?“. Paskaitoje jie sužinojo, kad daugiau pinigų galima turėti trim būdais: daugiau uždirbti, priversti pinigus dirbti arba tiesiog mažiau išleisti. Pirmieji du punktai reikalauja naujų įgūdžių ir žymiai daugiau pastangų, tuo tarpu taupymas yra paprastas metodas, kurio efektyvumą apriboja tik gaunamų pajamų suma, todėl draugai daug negalvoję nutarė taupyti.

Visiems trims vaikinams pirmąją mėnesio dieną tėvai duoda pinigų: Simas – gauna a litų, Linas – b litų, Mantas – c litų. Kiekvieną dieną vaikinai tam tikrą sumą pinigų išleidžia. Išlaidas kruopščiai fiksuoja taupymo žurnale.

Parašykite programą, kuri:

- apskaičiuotų kiekvieno mokinio išlaidas per k dienų;
- kiek kiekvienas vaikinai vidutiniškai išleidžia pinigų per vieną dieną;
- kokią pinigų sumą išleido kiekvienas per k dienų;
- kiek buvo dienų iš viso, kai vaikinai išleido daugiau pinigų už vidutinės dienos išlaidas (skaičiuojama bendra dienų suma: pirmo vaikinio dienų skaičius + antro vaikinio dienų skaičius + trečio vaikinio dienų skaičius).

Duomenų faile *Duomenys.txt*:

- pirmoje eilutėje įrašytas dienų kiekis k ,
- antroje eilutėje trys realieji skaičiai – kiekvieno vaikinio turima pinigų suma litais;
- likusiose k eilučių – kiekvieno vaikinio išleidžiami pinigai per dieną (realieji skaičiai).

Rezultatų faile *Rezultatai.txt* įrašykite:

- kiekvieno vaikinio išleistą pinigų sumą per k dienų;
- kiekvieno vaikinio vidutiniškai išleidžiamą pinigų sumą per dieną;
- kiek pinigų liko kiekvienam vaikinui;
- kiek buvo dienų, kai vaikinai išleido daugiau už vidutinės dienos išlaidas.

Reikalavimai programai:

1. naudokite realiųjų skaičių masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite funkciją, skaičiuojančią kiek pinigų išleido kiekvienas vaikinai;
4. sukurkite funkciją, skaičiuojančią kiek vidutiniškai pinigų išleidžia kiekvienas vaikinai per dieną;
5. sukurkite funkciją, kuri apskaičiuotų kiek pinigų liko kiekvienam vaikinui;
6. sukurkite funkciją, kuri apskaičiuotų kiek kartų visi vaikinai išleido daugiau pinigų už vidutinės dienos išlaidas;
7. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

Duomenys	Rezultatai
10 100.00 105.00 120.00 2.50 3.00 1.50 7.00 5.00 4.00 3.25 2.75 1.57 2.50 2.35 2.40 0.00 1.50 2.00 0.00 0.00 0.00 4.25 2.75 3.25 1.25 1.00 1.35 5.25 1.25 6.00 2.00 1.00 3.00	Vaikinai išleido: 28.00 20.60 25.07 Dienos išlaidų vidurkis: 2.80 2.06 2.51 Vaikinams liko pinigų: Simui 72.00; Linui 84.40; Mantui 94.93 Dienų skaičius, kai vaikinai išleido daugiau už savo dienos vidurkį: 13

4. Varžybos. „Ažuolo“ gimnazijoje kiekvienais metais vyksta sportinės varžybos. Varžybose dalyvauja visos klasės. Kiekviena klasė turi suburti tris mišrias komandas (sudarytas iš vaikinių ir merginų): krepšinio, tinklinio ir futbolo. Taisyklėse numatyta, kad mokinys gali būti tik vienos komandos narys, krepšinio komandą turi sudaryti – ne mažiau 7 nariai, futbolo – ne mažiau 12, tinklinio – 7.

1b klasės mokiniai nusprendė sudaryti komandas pagal gebėjimą žaisti (krepšinių, futbolo ar tinklinių), Parašykite programą, kiek komandų suburs 1b klasė.

Duomenų faile *Duomenys.txt*:

- pirmoje duomenų failo eilutėje įrašytas mokinių skaičius n ($n \leq 30$);
- likusiose n eilučių nurodyta informacija apie mokinį: lytis (mergina – M, vaikas – V), ūgis cm ir sporto šaka (kuri mokiniui sekasi geriausiai: krepšinis – K, futbolas – F, tinklinis – T), kai kurie mokiniai neseniai sirgo ir negalėjo dalyvauti varžybose, jie nutarė palaikyti komandas. Duomenų faile jie pažymėti S raide.

Rezultatų faile *Rezultatai.txt* įrašykite:

- kiek mokinių gali žaisti krepšinių,
- jei mokinių gebančių žaisti krepšinių skaičius ne mažesnis už 7, kiek tarp jų vaikinių ir merginų;
- krepšinio komandos sudėtį;
- krepšinio komandos ūgio vidurkį.
- kiek mokinių gali žaisti tinklinių,
- Jei mokinių daugiau skaičius ne mažesnis už 7, kiek tarp jų vaikinių ir merginų;
- tinklinio komandos sudėtį;
- tiek mokinių gali žaisti futbolo,
- Jei mokinių skaičius ne mažesnis už 12, kiek tarp jų vaikinių ir merginų;
- futbolo komandos sudėtį.
- kiek mokinių sudarys komandų palaikymo grupę, kiek tarp jų vaikinių ir merginų;
- palaikymo komandos sudėtį.

Reikalavimai programai:

1. naudokite sveikųjų skaičių ir simbolių masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite procedūrą (funkciją void C++) skaičiuojančią, kiek mokinių žaidžia tik krepšinių, tik tinklinių, tik futbolo, kiek merginų ir vaikinių kiekvienoje komandoje;
4. sukurkite funkciją krepšinio komandos vidurkiui apskaičiuoti;
5. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

Duomenys	Rezultatai
<p>30 M 180 K V 182 K V 175 T V 178 F M 164 F M 175 T M 162 S V 189 K V 188 K M 153 T M 168 S M 175 K M 170 T M 163 T M 169 S M 165 F M 173 K V 178 F V 178 F V 176 T V 189 F V 196 K V 182 T V 177 F M 159 S M 178 F V 169 F M 159 F V 179 F V 176 F</p>	<p>žaidžia krepšinių 7 komandą sudaro vaikinai 4 merginos 3 M 180 K V 182 K V 189 K V 188 K M 175 K M 173 K V 196 K krepšinio komandos ūgio vidurkis 183 žaidžia tinklinių 7 komandą sudaro vaikinai 3 merginos 4 V 175 T M 175 T M 153 T M 170 T M 163 T V 176 T V 182 T žaidžia futbola 12 komandą sudaro vaikinai 8 merginos 4 V 178 F M 164 F M 165 F V 178 F V 178 F V 189 F V 177 F M 178 F V 169 F M 159 F V 179 F V 176 F komandų palaikymo grupė 4 palaikymo komandą sudaro vaikinai 0 merginos 4 M 162 S M 168 S M 169 S M 159 S</p>
<p>30 M 180 K V 182 S V 175 T V 178 F M 164 F M 175 T M 162 S V 189 S V 188 K M 153 T M 168 S M 175 S M 170 T M 163 T M 169 S M 165 F M 173 K V 178 F V 178 F V 176 T V 189 F V 196 K V 182 T V 177 F</p>	<p>žaidžia krepšinių 4 krepšinio komandos sudaryti negalės žaidžia tinklinių 7 komandą sudaro vaikinai 3 merginos 4 V 175 T M 175 T M 153 T M 170 T M 163 T V 176 T V 182 T žaidžia futbola 12 komandą sudaro vaikinai 8 merginos 4 V 178 F M 164 F M 165 F V 178 F V 178 F V 189 F V 177 F M 178 F V 169 F M 159 F V 179 F V 176 F</p>

M 159 S	komandų palaikymo grupė 7
M 178 F	palaikymo komandą sudaro vaikinai 2 merginos 5
V 169 F	V 182 S
M 159 F	M 162 S
V 179 F	V 189 S
V 176 F	M 168 S
	M 175 S
	M 169 S
	M 159 S

5. **Balsavimas.** Reikia sudaryti geriausio konsultanto/moksleivio/dėstytojo/... rinkimų rezultatų lentelę iš užpildytų balsavimo biuletenių. Visi biuleteniai buvo užpildyti teisingai. Uždavinio supaprastinimui naudojamas daugumos metodas.

Reikalavimai programai:

1. naudokite sveikųjų skaičių masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

Pradiniai duomenys pateikti faile *balsavimo_duomenys.txt*:

11 5	Pirmasis skaičius - balsavusiųjų skaičius n , $1 \leq n \leq 1000$; antrasis skaičius - kandidatų skaičius k , $1 \leq k \leq 100$.
1 3 2 1 5 3 4 2 1 3 2	Tolesnėse n eilučių įrašyta, už kurią kandidata balsavo kiekvienas rinkėjas.

Į failą *balsavimo_lentele.txt* išvesti lentelę:

Kandidato nr.	Balsų kiekis
1	3
2	3
3	3
4	1
5	1

6. **Svetainės populiarumas.** Reikia sudaryti internetinės svetainės lankomumo supaprastintą suvestinę: kiek kartų daugiausiai buvo peržiūrėtas vienas puslapis ir kiek puslapių daugiausiai peržiūrėjo vienas lankytojas.

Reikalavimai programai:

1. naudokite sveikųjų skaičių masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite didžiausios reikšmės radimo funkciją;
4. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

Pradiniai duomenys skaitomi iš failo *puslapiu_peržiūros.txt*:

7 3 2	Pirmasis skaičius - peržiūrų skaičius p , $1 \leq p \leq 2^{31}$, antrasis skaičius - puslapių skaičius k , $1 \leq k \leq 1000$, trečiasis skaičius - lankytojų skaičius $lank$, $1 \leq lank \leq 1000$
1 1 1 2 2 1 3 2 3 2 3 2 3 2	Tolesnėse p eilučių įrašyta: puslapio unikalus numeris ir lankytojo unikalus numeris, pvz. 2 1 - puslapio unikalus numeris yra 2, lankytojo unikalus numeris yra 1.

Rezultatų faile *lankomumo_suvestinė.txt* turi būti įrašyta:

Vienas puslapis daugiausiai peržiūrėtas: 4 kartus.
Vienas lankytojas daugiausiai peržiūrėjo: 5 puslapius.

7. Geografijos kontrolinio darbo pažymių pasiskirstymas. Mokyklos dienyne mokytojas surašė geografijos kontrolinio darbo pažymius. Pažymiai kinta nuo 1 iki 10. Parašykite programą, kuri suskaičiuotų po kiek procentų yra 10-ukų, 9-ukų, ..., 1-ukų.

Pradinių duomenų failo *Duomenys.txt* pirmoje eilutėje įrašytas pažymių kiekis, antroje eilutėje – kiekvieno mokinio geografijos kontrolinio darbo pažymiai vienas nuo kito atskirti tarpais.

Rezultatų faile *Rezultatai.txt* reikia įrašyti, kiek procentų kokių pažymių buvo parašyta už kontrolinį darbą: pažymys ir kiek procentų tokių pažymių parašyta dviejų ženklų po kablelio tikslumu.

Reikalavimai programai:

1. naudokite sveikųjų skaičių masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite funkciją, skaičiuojančią pažymių pasiskirstymą procentais;
4. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

Duomenys	Rezultatai
13	1 7.69
1 5 8 7 10 9 9 8 7 7 4 6 8	2 0.00
	3 0.00
	4 7.69
	5 7.69
	6 7.69
	7 23.08
	8 23.08
	9 15.38
	10 7.69

11. Masyvai. Didžiausios ir mažiausios reikšmių paieška

1. Dailusis čiuožimas. Dailiojo čiuožimo teisėjų brigadą sudaro n ($5 \leq n \leq 12$) šios sporto šakos žinovų. Vertindami sportininko pasirodymą, jie skiria balus už atlikimo techniką ir artistišumą. Vertinimo skalė yra šešių balų, naudojant ir skaičiaus dešimtąsias dalis. Vienas pats aukščiausias ir vienas pats žemiausias įverčiai atmetami. Jei keli teisėjai davė didžiausią ar mažiausią įvertinimą, atmetami pirmo sąraše teisėjo balai. Sukurkite programą sportininko balų vidurkiui apskaičiuoti.

Pradinių duomenų faile *Duomenys.txt* yra surašoma:

- Pirmoje eilutėje nurodoma, kiek teisėjų ($5 \leq n \leq 12$) vertina sportininką.
- Tolimesnėse n eilučių yra surašoma po du realiuosius skaičius, nurodančius teisėjo skirtus balus už techniką ir artistišumą.

Rezultatų faile *Rezultatai.txt* reikia pateikti skirtingose eilutėse įverčių už techniką ir artistišumą vidurkius (vidurkius pateikti vieno skaičiaus po kablelio tikslumu).

Reikalavimai programai:

1. duomenims saugoti naudokite masyvus.
2. sukurkite pradinių duomenų skaitymo funkciją void C++ (procedūrą Pascal);
3. sukurkite tris funkcijas, gražinančias reikšmę per savo vardą C++, (funkciją Pascal), apskaičiuojančias: mažiausią įvertinimą, didžiausią įvertinimą, vidurkį be mažiausio ir didžiausio įvertinimų;
4. sukurkite rezultatų rašymo į failą funkciją void C++ (procedūrą Pascal).

Duomenys	Rezultatai
7 5.6 5.8 5.2 5.7 5.5 5.8 5.2 5.4 5.4 5.5 5.3 5.8 5.7 5.6	5.4 5.7

2. Indėliai. Banke yra n ($1 \leq n \leq 10000$) klientų. Jų indėlių sumos surašytos į sąrašą. Kiekvienas klientas užkoduotas unikaliu skaičiumi, kuris sutampa su indėlių sumų sąrašo eilės numeriu. Banko valdytojui prireikė informacijos apie didžiausią ir mažiausią banke laikomus indėlius, t.y. kliento kodo ir indėlio dydžio, vidutinio indėlių dydžio bei visų indėlių sąrašo, išrikiuoto mažėjimo tvarka (pateikiama kodas ir indėlis). Parašyti programą, kuri tai atliktų.

Pradinių duomenų faile *Bankas.txt*: pirmoje eilutėje – banko klientų kiekis n ($1 \leq n \leq 10000$). Antroje eilutėje n realiųjų skaičių – banko klientų indėliai litais.

Reikalavimai programai:

1. duomenims saugoti naudokite masyvus;
2. sukurkite pradinių duomenų skaitymo funkciją void C++ (procedūrą Pascal);
3. sukurkite funkcijas void C++ (procedūrą Pascal), randančią didžiausio arba mažiausio indėlio dydį ir kliento kodą;
4. sukurkite funkciją, gražinančią reikšmę per savo vardą C++, (funkciją Pascal), apskaičiuojančią indėlių dydžių vidurkį;
5. sukurkite funkciją void C++ (procedūrą Pascal), rikiuojančią indėlių sąrašą mažėjimo tvarka;
6. sukurkite rezultatų rašymo į failą funkciją void C++ (procedūrą Pascal).

Rezultatų failo *Rezultatai.txt* pavyzdys yra pateiktas (indėlių reikšmės pateikiamos dviejų skaičių po kablelio tikslumu).

Duomenys	Rezultatai
6 1987.23 456.57 235.89 789.48 1987564.97 978687.67	Didžiausias indėlis 1987564.97 kodas 5 Mažiausias indėlis 235.89 kodas 3 Vidutinis indėlio dydis 494953.64 Kodas Indėlis 5 1987564.97 6 978687.67 1 1987.23 4 789.48 2 456.57 3 235.89

3. Teleloto. Petras kiekvieną savaitę perka po kelis Teleloto bilietus (bilietas kainuoja 2 Lt), tikėdamasis išlošti daug pinigų ir pakeliauti po pasaulį. Jis veda užrašus: kiekvieną kartą įrašo, kiek pinigų išleido bilietams ir kiek išlošė.

Parašykite programą, kuri apskaičiuotų:

- kiek iš viso pinigų buvo išleista bilietams ir kiek pinigų išlošta;
- Petro didžiausią laimėjimą, bei kiek bilietų jis pirkė;
- ar galutinis rezultatas buvo nuostolingas, pelningas ar nulinis, t.y. kiek išleista, tiek ir išlošta.

Duomenų faile *Duomenys.txt*:

- pirmoje eilutėje parašytas bilietų pirkimo skaičius n ($0 < n < 53$),
- kitose n eilučių yra po du skaičius: pirmasis reiškia, už kiek pinigų pirka loterijos bilietų, o antrasis – kiek laimėta.

Visi skaičiai sveikieji.

Rezultatų faile *Rezultatai.txt* įrašykite:

- pirmoje eilutėje – du skaičius: kiek pinigų išleido Petras bilietams ir kiek laimėjo;
- antroje eilutėje – Petro didžiausią laimėjimą;
- trečioje eilutėje – kiek iš viso bilietų pirko Petras;
- ketvirtoje eilutėje – galutinį rezultatą: žodį „Nuostolis“, jeigu išleido pinigų daugiau nei laimėjo, ir nuostolio dydį – praloštų pinigų skaičių; žodį „Pelnas“, jeigu išlošta daugiau, nei išleista pinigų bilietams, ir koks gautas pelnas. Jeigu pinigų išlošta tiek, kiek išleista bilietams pirkti, tuomet tik žodį „Lygiosios“.

Reikalavimai programai:

1. naudokite sveikųjų skaičių masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite funkciją, skaičiuojančią kiek pinigų išleido bilietams pirkti ir kiek laimėjo iš viso.
4. sukurkite funkciją, skaičiuojančią Petro didžiausią laimėjimą;
5. sukurkite funkciją, skaičiuojančią kiek bilietų pirko Petras;
6. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

Duomenys	Rezultatai
10	56 Lt 2198 Lt
2 10	Didžiausias laimėjimas 2000 Lt
4 0	Petras pirko 28 bilietus
6 100	Pelnas 2142 Lt
2 78	
10 2	
2 0	
10 8	
4 0	
6 0	
10 2000	

4. Pinigų keitimas. Jonas ką tik grįžo iš Jungtinių Amerikos valstijų ir ketina keletą mėnesių praleisti Lietuvoje, todėl jam reikia dolerius pakeisti į litus. Jis peržiūrėjo dolerio pirkimo kursą bankuose (grynaisiais ir negrynaisiais pinigais).

Parašykite programą, kuri nustatytų, kuri banką pasirinkti Jonui, kad būtų pelningiausia dolerius pakeisti į litus atsiskaitant grynaisiais pinigais ir atsiskaitant negrynaisiais pinigais ir kuriuose nepelningiausia.

Duomenų faile *Duomenys.txt*:

- pirmoje eilutėje įrašytas bankų skaičius n ($n > 0$),
- kitose n eilučių yra įrašytas banko pavadinimas (20 pozicijų) ir du skaičiai: pirmasis reiškia dolerio vertę perkant grynaisiais pinigais, antrasis perkant negrynaisiais pinigais.

Duomenys	Rezultatai
6	Pelningiausia keisti grynuosius pinigus dolerio vertė 2.5101 Lt
AB Seb bankas 2.4866 2.4901	Šiaulių bankas
Šiaulių bankas 2.5100 2.5040	Pelningiausia keisti
Medicinos bankas 2.5032 2.5096	negrynuosius pinigus dolerio
Swedbank 2.4938 2.5549	vertė 2.5549 Lt Medicinos
Nordea bank 2.4827 2.4356	bankas
DNB bankas 2.5079 2.5044	Pelningiausia keisti
	negrynuosius pinigus dolerio
	vertė 2.5549 Lt Swedbank
	Mažiausiai pelninga keisti
	grynuosius pinigus dolerio
	vertė 2.4827 Lt Nordea bank
	Mažiausiai pelninga keisti
	negrynuosius pinigus dolerio
	vertė 2.4356 Lt Nordea bank

Rezultatų faile *Rezultatai.txt* įrašykite:

- pirmoje eilutėje – kuriame banke Jonui geriausia keisti pinigus mokant grynaisiais;
- antroje eilutėje – kuriame banke Jonui geriausia keisti pinigus mokant negrynaisiais;
- trečioje eilutėje – kuriame banke Jonui labiausiai nuostolinga keisti grynuosius pinigus;
- ketvirtoje eilutėje – kuriame banke Jonui labiausiai nuostolinga keisti negrynuosius pinigus.

Jei yra keli bankai, juos išveskite tokia eilės tvarka, kokia jie nurodyti duomenų faile.

Reikalavimai programai:

1. naudokite eilutės ir realiųjų skaičių masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite funkciją, pelningiausiam pinigų keitimui rasti.
4. sukurkite funkciją, nuostolingiausiam pinigų keitimui rasti.
5. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

5. Mokinių ūgiai. Mokyklos medicinos kabinete buvo matuojami klasės mokinių ūgiai. Gydytojas kiekvienos klasės mokinio ūgius užsirašė į sąsiuvinį. Parašykite programą, padedančią gydytojui apskaičiuoti didžiausią ir mažiausią kiekvienos klasės mokinio ūgį, klasės mokinių ūgių vidurkį, o taip pat nustatyti, kurie ūgiai viršija klasės mokinių ūgių vidurkį.

Pradinių duomenų failo *Duomenys.txt* pirmoje eilutėje saugomas klasės mokinių skaičius n , toliau – n sveikųjų skaičių, nurodančių kiekvieno mokinio ūgį.

Rezultatų faile *Rezultatai.txt* reikia išvesti didžiausią ir mažiausią ūgį, ūgių vidurkį, o taip pat ūgius, kurie viršija klasės mokinių ūgių vidurkį.

Reikalavimai programai:

1. naudokite sveikųjų skaičių masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite funkcijas, skaičiuojančias: didžiausią ir mažiausią ūgį (viena arba dvi funkcijos), ūgių vidurkį, klasės vidurkį viršijančius ūgius.
4. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

Duomenys	Rezultatai
15 164 176 173 152 146 181 190 171 180 158 160 170 184 173 165	Didžiausias ūgis: 190 Mažiausias ūgis: 146 Vidutinis ūgis: 169.53 Viršijantys vidurkį ūgiai: 176 173 181 190 171 180 170 184 173

6. Autobuso vairuotojo darbo ir poilsio režimas. Autobuso vairuotojas dirba 20 dienų per mėnesį ir per vieną dieną turi vidutiniškai nuvažiuoti tam tikrą kilometrų kiekį. Tačiau autobuso vairuotojui labai svarbus darbo ir poilsio režimas, todėl kiekvieną dieną leidžiama nukrypti nuo šios normos ne daugiau kaip 20%. Kitu atveju vairuotojui taikoma bauda mažinant bazinį atlyginimą po 1 Lt už kiekvieną viršytą ar trūkstamą kilometrą. Atlyginimas mažinamas tik už tuos kilometrus, kurie viršija ar kurių trūksta įvertinus leistiną nukrypimo normą.

Pradinių duomenų faile *Duomenys.txt* pirmoje eilutėje įrašytas vairuotojui vidutiniškai per dieną nuvažiuoti priklausantis kilometrų skaičius ir vairuotojo bazinis atlyginimas, o sekančiose – vairuotojo kiekvieną mėnesio darbo dieną nuvažiuotų kilometrų skaičius.

Rezultatų faile *Rezultatai.txt* reikia išvesti:

- kiek vairuotojas vidutiniškai nuvažiuoja kilometrų per dieną (skaičiuojamas vieno mėnesio vidurkis);
- kiek buvo dienų per mėnesį, kai viršijo ir kiek dienų, kai nepasiekė vidutinio reikiamo nuvažiuoti kilometrų skaičiaus;
- kiek buvo dienų, kai pažeidė darbo ir poilsio režimą;
- mėnesio dieną, kurią režimas pažeistas labiausiai;
- kokio dydžio atlyginimą vairuotojas gaus.

Reikalavimai programai:

1. naudokite sveikųjų skaičių masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite funkciją, suskaičiuojančią autobuso vairuotojo per dieną nuvažiuotų kilometrų vidurkį;
4. sukurkite funkciją, suskaičiuojančią kiek dienų per mėnesį vairuotojas viršija ir kiek nepasiekia vidutinio reikiamo nuvažiuoti kilometrų skaičiaus, o taip pat kiek dienų vairuotojas pažeidžia darbo ir poilsio režimą;
5. sukurkite funkciją, apskaičiuojančią mėnesio dieną, kurią režimas pažeidžiamas labiausiai;
6. sukurkite funkciją, suskaičiuojančią galutinį vairuotojo atlyginimą;
7. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

Duomenys	Rezultatai
400 3000	Vidutiniškai nuvažiuoja per dieną: 387.6
500	Dienų kiekis, kai viršijama: 9
300	Dienų kiekis, kai atsiliekama: 11
250	Dienos, kai pažeidžiamas režimas: 5
420	Mėnesio diena, kurią režimas pažeidžiamas
380	labiausiai: 13
410	Galutinis atlyginimas: 2550
360	
330	
370	
390	
360	
370	
100	
600	
430	
425	
429	
470	
458	
400	

12. Masyvo elementų šalinimas ir įterpimas

1. Informacinių technologijų konkursas. Informacinių technologijų konkurso I etape dalyvavo n ($1 \leq n \leq 100$) mokinių. Jų pasirodymas buvo vertinamas k balų ($0 \leq k \leq 100$) ir gauti balai surašyti į sąrašą. Į II etapą patenks tik tie mokiniai, kurių surinkti balai viršys visų mokinių gautų balų vidurkį. Sukurkite programą, kuri sąrašą paliktų tik atrinktų į II etapą mokinių balus ir užrašytų, keli mokiniai atrinkti dalyvauti II konkurso etape.

Pradinių duomenų faile *Duomenys.txt*: pirmoje eilutėje nurodyta, kiek mokinių n dalyvavo konkurse. Antroje eilutėje pateikiama n sveikųjų skaičių, nurodančių mokinių balus.

Rezultatų failo *Rezultatai.txt* pavyzdys pateiktas lentelėje.

<i>Duomenys.txt</i>	<i>Rezultatai.txt</i>
10 95 25 100 12 56 74 5 80 36 45	II etape dalyvaus 5 mokiniai Jie surinko balų: 95 100 56 74 80

Reikalavimai programai:

1. duomenims saugoti naudokite masyvus;
2. sukurkite pradinių duomenų skaitymo funkciją void C++ (procedūrą Pascal);
3. sukurkite funkciją, gražinančią reikšmę per savo vardą C++, (funkciją Pascal), apskaičiuojančią balų vidurkį;
4. sukurkite dvi funkcijas void C++ (procedūras Pascal): 1) elemento šalinimui bei 2) šalinamo elemento suradimui;
5. sukurkite rezultatų rašymo į failą funkciją void C++ (procedūrą Pascal).

2. Komanda. Į šuolininkų į toli komandą reikia išrinkti 10 sportininkų. Atrankos kriterijus – į komandą patenka sportininkės, turinčios geriausias asmeninius rezultatus. Po pirmo atrankos etapo buvo atrinkta n ($1 \leq n \leq 10$) sportininkų. Jų geriausi rezultatai didėjimo tvarka surašyti faile *Duomenys1.txt*. Po antro atrankos etapo buvo atrinkta dar m ($1 \leq m \leq 10$) sportininkų. Jų geriausi rezultatai nerikiuoti surašyti faile *Duomenys2.txt*. Parašykite programą, kuri išrinktų geriausias 10 sportininkų, jų geriausių rezultatų sąrašas turi būti pateikiamas didėjimo tvarka, bei praneštų, jei rinktinė dar pilnai nesuformuota. Jei dviejų sportininkų rezultatai vienodi, į komandą patenka pirmą atrankos etapą įveikusi sportininkė.

Pradinių duomenų pirmame faile *Duomenys1.txt*: pirmoje eilutėje nurodytas sportininkų skaičius n . Antroje eilutėje įrašyta n realiųjų skaičių – sportininkų geriausi rezultatai metrais, išrikiuoti didėjimo tvarka.

Pradinių duomenų antrame faile *Duomenys2.txt*: pirmoje eilutėje įrašytas sportininkų skaičius m . Antroje eilutėje įrašyta m realiųjų skaičių – sportininkų geriausi rezultatai metrais.

Rezultatų failo „*Rezultatai.txt*“ pavyzdys pateiktas lentelėje.

<i>Duomenys1.txt</i>	<i>Duomenys2.txt</i>	<i>Rezultatai.txt</i>
6 4.26 4.56 4.98 4.99 5.02 5.10	7 4.97 4.21 5.23 5.00 4.32 5.12 4.75	Atrinktų sportininkų geriausių rezultatų sąrašas: 4.56 4.75 4.97 4.98 4.99 5.00 5.02 5.10 5.12 5.23
5 4.26 4.56 4.98 4.99 5.02	2 4.97 4.21	Komandai dar trūksta 3 sportininkų Atrinktų sportininkų geriausių rezultatų sąrašas: 4.21 4.26 4.56 4.97 4.98 4.99 5.02

Reikalavimai programai:

1. duomenims saugoti naudokite masyvus;
2. sukurkite pradinių duomenų skaitymo funkciją void C++ (procedūrą Pascal);
3. sukurkite tris funkcijas void C++ (procedūrą Pascal), atliekančias: 1) įterpimo, 2) šalinimo veiksmus bei 3) surandančią, kur reikia įterpti sportininkę;
4. sukurkite rezultatų rašymo į failą funkciją void C++ (procedūrą Pascal).

3. Žalgirio komandos 2013/2014 metų sezonas. Žalgirio krepšinio komandos sudėtį 2013/2014 metų sezoną sudaro komandos senbuviai (rungtyniavę praėjusį sezoną) ir naujokai. Parašykite programą, kokia ši sezoną komandos sudėtis. Žaidėjai turi būti išdėstyti pagal ūgį (įterpkite naujokų duomenis į senąjį žaidėjų sąrašą nekeisdami išdėstymo tvarkos).

Duoti du duomenų failai. Komandos senbuviai įrašyti faile *senbuviai.txt*, naujokai – *naujokai.txt*.

Failų pirmoje eilutėje nurodytas žaidėjų skaičius, antroje – žaidėjų ūgiai.

Rezultatų faile *komanda.txt* įrašykite 2013/2014 metų sezono Žalgirio krepšinio komandos krepšininkų ūgius.

Reikalavimai programai:

1. naudokite sveikųjų skaičių masyvus;
2. sukurkite pradinių duomenų vieną arba dvi skaitymo procedūras (funkcijas void C++);
3. sukurkite naujų narių įterpimo procedūrą (funkciją C++);
4. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

<i>senbuviai.txt</i>	<i>naujokai.txt</i>	<i>komanda.txt</i>
5 211 210 205 204 198	10 217 208 203 196 195 195 193 193 190 183	217 211 210 208 205 204 203 198 196 195 195 193 193 190 183

4. Parduotuvė. Maisto prekių parduotuvėje yra n ($1 < n < 200$) prekių. Ant kiekvienos prekės pakuotės užrašytas prekės galiojimo laikas. Pardavėjas kiekvieną dieną peržiūri visų prekių galiojimo laiką, ir, jei galiojimo laikas pasibaigęs, prekes išima iš prekybos. Parašykite programą, kuri iš pradinių duomenų pašalintų prekes, kurių galiojimo laikas pasibaigęs ir apskaičiuotų kelios prekių rūšys išimtos iš prekybos, kiek prekių vienetų išimta iš prekybos ir kokį nuostolį patirs parduotuvė.

Duomenų faile *Duomenys.txt* pirmoje eilutėje įrašytas prekių skaičius n ir data, kai pardavėjas tikrina prekių galiojimo laiką. Kitose failo eilutėse nurodytas prekių kiekis, prekės kaina, prekės galiojimo laikas.

Rezultatų faile *Rezultatai.txt* įrašykite:

- prekes, kurių galiojimo laikas tinkamas: prekių kiekį, kainą ir galiojimo laiką (duomenis pateikite lentele);
- kelių rūšių prekių baigėsi galiojimo laikas;
- kiek prekių reikės išimti iš prekybos;
- kokį nuostolį patirs parduotuvė.

Reikalavimai programai:

1. naudokite sveikųjų ir realiųjų skaičių masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkcijas void C++);
3. sukurkite šalinimo procedūrą (funkciją C++);
4. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

<i>Duomenys.txt</i>	<i>Rezultatai.txt</i>
16 20140224 10 12.45 20140221 5 20.45 20140315 15 10.95 20140425 7 4.50 20140627 8 15.85 20150121 6 4.56 20141228 10 12.30 20140312 12 15.00 20140225 2 30.42 20140223 11 8.69 20140315 30 7.89 20140315 4 9.69 20140817 5 6.98 20140521 8 8.78 20140624 6 14.98 20140228 12 2.18 20140225	<pre> Prekės prekyboje ----- Prekių kiekis Prekės kaina Prekės galiojimo laikas ----- 5 20.45 20140315 15 10.95 20140425 7 4.50 20140627 8 15.85 20150121 6 4.56 20141228 10 12.30 20140312 12 15.00 20140225 11 8.69 20140315 30 7.89 20140315 4 9.69 20140817 5 6.98 20140521 8 8.78 20140624 6 14.98 20140228 12 2.18 20140225 ----- Pasibaigė galiojimo laikas 2 rūšių prekių Išimti iš prekybos reikės 12 prekių Parduotuvė patirs 185.34 Lt nuostolį </pre>

13. Dvimatis masyvas

1. Didžiausio ir mažiausio elementų minimali suma. Duotas sveikųjų skaičių dvimatis masyvas $A(n,m)$, kur $(2 \leq n, m \leq 100)$. Eilutę, kurios didžiausio ir mažiausio elementų suma yra minimali, reikia pašalinti iš masyvo. Jei yra dvi vienodos minimalios sumos, šalinti pirmiau esančią eilutę.

Spausdinti pradinį ir pertvarkytą dvimačius masyvus lentelėmis, užrašant eilučių ir stulpelių numerius.

Pradinių duomenų faile *Duomenys.txt*:

Pirmoje eilutėje nurodytas eilučių kiekis n ir stulpelių kiekis m . Kitose eilutėse surašomos dvimačio masyvo reikšmės.

Rezultatų failo *Rezultatai.txt* pavyzdys pateiktas lentelėje.

<i>Duomenys</i>	<i>Rezultatai</i>
<pre> 5 6 0 5 -9 10 5 7 0 0 -2 -3 5 10 1 9 4 8 7 5 -5 -9 -2 -1 -1 -6 4 5 8 0 0 -4 </pre>	<pre> Pradinis dvimatis masyvas: 1 2 3 4 5 6 ----- 1 I 0 5 -9 10 5 7 2 I 0 0 -2 -3 5 10 3 I 1 9 4 8 7 5 4 I -5 -9 -2 -1 -1 -6 5 I 4 5 8 0 0 -4 Pakeistas dvimatis masyvas: 1 2 3 4 5 6 ----- 1 I 0 5 -9 10 5 7 2 I 0 0 -2 -3 5 10 3 I 1 9 4 8 7 5 4 I 4 5 8 0 0 -4 </pre>

Reikalavimai programai:

1. Duomenims saugoti naudokite dvimatį masyvą masyvus.
2. Sukurkite pradinių duomenų skaitymo funkciją void C++ (procedūrą Pascal).
3. Sukurkite dvi funkcijas void C++ (procedūrą Pascal): pirma randa eilučių didžiausio ir mažiausio elemento sumas, antroji šalina iš dvimačio masyvo vieną eilutę.
4. Sukurkite funkciją, gražinančią reikšmę per savo vardą C++, (funkciją Pascal), surandančią eilutės, kurios suma mažiausia, indeksą.
5. Sukurkite dvimačio masyvo reikšmių rašymo į failą funkciją void C++ (procedūrą Pascal).

2. Matematikos kontroliniai darbai. n ($1 \leq n \leq 30$) mokinių per pusmetį rašo m ($1 \leq m \leq 10$) matematikos kontrolinių darbų. Lentelėje (matricoje), sudarytoje iš n eilučių (1 eilutė – 1 mokyns) ir m stulpelių (1 stulpelis – 1 kontrolinis), yra surašyti įvertinimai dešimtbalėje sistemoje. Reikia rasti kiekvieno mokinio įvertinimų vidurkį ir kiekvieno kontrolinio darbo įvertinimų vidurkį. Vidurkius pateikite dviejų skaičių po kablelio tikslumu. Pateiktą lentelę reikia surikiuoti mokinių vidurkių mažėjimo tvarka.

Pradinių duomenų faile *Duomenys.txt*: pirmoje eilutėje nurodytas lentelės eilučių (t.y. mokinių) skaičius n ir stulpelių (t.y. kontrolinių darbų) skaičius m . Tolesnėse n eilučių yra surašyta po m įvertinimų.

Rezultatų failo *Rezultatai.txt* pavyzdys pateiktas lentelėje (spausdinami ir mokinių, ir kontrolinių darbų numeriai, įvertinimai, vidurkiai).

Pradiniai duomenys	Rezultatai							
5 7	1	2	3	4	5	6	7	vid
10 5 9 2 5 9 8	10	10	9	9	8	9	7	8.86
10 10 9 9 8 9 7	5	9	8	9	8	7	6	7.43
2 6 7 3 5 5 6	10	5	9	2	5	9	8	6.86
5 9 8 9 8 7 6	8	6	6	3	7	8	8	6.57
8 6 6 3 7 8 8	2	6	7	3	5	5	6	4.86
	vid	7.00	7.20	7.80	5.20	6.60	7.60	7.00

Reikalavimai programai:

1. Įvertinimams saugoti naudokite dvimatį masyvą, vidurkiams - vienmatį.
2. Sukurkite pradinių duomenų skaitymo funkciją void C++ (procedūrą Pascal).
3. Sukurkite dvi funkcijas void C++ (procedūras Pascal), skirtas vidurkiams dvimačio masyvo eilutėje ir stulpelyje rasti.
4. Sukurkite funkciją void C++ (procedūras Pascal), kuri surikiuoja duomenis mažėjimo tvarka.
5. Sukurkite rezultatų rašymo į failą funkciją void C++ (procedūrą Pascal).

3. Sandaugos įstrižainėse. Duota $n \times n$ dydžio natūraliųjų skaičių lentelė. Patikrinkite, ar šios lentelės pagrindinių įstrižainių skaičių sandaugos lygios. Raskite visas kvadratinės lentes (pradinės lentelės dalį), tenkinančias šią sąlygą. 1×1 dydžio lentelė taip pat gali būti sprendiniu. Duomenų failo *Duomenys.txt* pirmoje eilutėje įrašytas lentelės dydis – skaičius n ($2 \leq n \leq 8$), likusiose n eilučių pateikiami lentelę sudarantys skaičiai.

Rezultatų faile įrašykite rastas kvadratinės lentes, kurių įstrižainių sandaugos lygios.

Reikalavimai programai:

1. Duomenims saugoti naudokite daugiamatį masyvą.
2. Sukurkite pradinių duomenų skaitymo funkciją void C++ (procedūrą Pascal).
3. Sukurkite funkciją, skaičiuojančią įstrižainių sandaugas C++, (procedūrą Pascal).
4. Sukurkite rezultatų rašymo į failą funkciją void C++ (procedūrą Pascal).

Duomenys	Rezultatas
3 4 5 6 1 3 2 2 7 3	4 5 6 1 3 2 2 7 3 4
3 3 2 1 3 2 3 3 2 1	3 2 1 3 2 3 3 2 1 3 2 3 2 3

4. Magiškasis kvadratas. Magiškuoju kvadratu vadiname kvadratinę natūraliųjų skaičių lentelę, kurios kiekvieno stulpelio, kiekvienos eilutės ir abiejų įstrižainių sumos lygios.

Duota $n \times n$ kvadratinė natūraliųjų skaičių lentelė. Parašykite programą, kuri nustatytų, ar duotas kvadratas yra magiškasis.

Pradiniai duomenys pateikti pradinių duomenų faile *Duomenys.txt*. Pirmoje eilutėje įrašytas natūralusis skaičius n ($2 \leq n \leq 50$). Tolesnėse n eilučių – po n natūraliųjų skaičių (kvadratinė lentelė).

Pradiniai duomenys yra tokie, kad bet kurios eilutės, stulpelio ar įstrižainės suma neviršys long long (C++), maxlongint (Pascal).

Rezultatą įrašykite į tekstinį rezultatų failą *Rezultatai.txt*.

Jei duotoji lentelė yra magiškasis kvadratas, tai rezultatas turi būti pranešimas: magiškasis kvadratas, jei ne – nemagiškasis kvadratas.

Reikalavimai programai:

1. Duomenims saugoti naudokite dvimatį masyvą.
2. Sukurkite pradinių duomenų skaitymo funkciją void C++ (procedūrą Pascal).
3. Sukurkite funkciją, skaičiuojančią įstrižainių, eilučių ir stulpelių sumas C++, (procedūrą Pascal).
4. Sukurkite funkciją, patikrinančią ar įstrižainių, eilučių ir stulpelių sumos yra lygios.
5. Sukurkite rezultatų rašymo į failą funkciją void C++ (procedūrą Pascal).

<i>Duomenys</i>	<i>Rezultatai</i>
3 2 9 4 7 5 3 6 1 8	Magiškasis kvadratas

3 12 17 16 12 15 11 14 13 18	Nemagiškasis kvadratas
---------------------------------------	------------------------

5. Laivų mūšis. Sudarykite programą, kuri klasikinio žaidimo „Laivų mūšis“ ([http://en.wikipedia.org/wiki/Battleship_\(game\)](http://en.wikipedia.org/wiki/Battleship_(game))) žaidimo lauko matricoje atsitiktine tvarka išdėstytų vienviečius, dviviečius, triviečius ir keturviečius laivus. Žaidimo lango matricos ilgis ir plotis yra x langelių. Žaidimo lauke laivai gali būti talpinami tik iš kairės į dešinę. Jei langeliai užimti vienviečio laivo, jie žymimi 1, dviviečio – 22, triviečio - 333, keturviečio – 4444. Jei talpinant laivą į žaidimų lauką nustatoma, kad kuris nors langelis užimtas kito laivo – laivas nebekuriamas.

Duomenų failo *Duomenys.txt* pirmoje eilutėje įrašytas skaičius, nusakantis žaidimų lango ilgį ir plotį, antroje – reikiamų sukurti vienviečių, dviviečių, triviečių ir keturviečių laivų kiekiai.

Rezultatų faile *Rezultatai.txt* turi būti išvedama sukurto laivų mūšio lauko matrica, o taip pat kiek vienviečių, dviviečių, triviečių ir keturviečių laivų nepavyko sukurti.

Reikalavimai programai:

1. Sukurti funkcijas, pildančias laivus žaidimų lauke.
2. Sukurti duomenų failą nuskaitančias ir rezultatus išvedančias funkcijas.
3. Atsitiktinio skaičiaus generavimui galima naudoti funkciją `rand()` ;
4. Žaidimo laukui saugoti reikia naudoti dvimatį masyvą.

Pradinių duomenų pavyzdys	Galimi rezultatai (dėl funkcijos <code>random</code> panaudojimo, rezultatai kiekvieną kartą gali skirtis)
12 2 2 3 5	000000000000 000000333000 044444000000 000044440000 000000000000 000022000100 000000002200 000000444400 000000000000 000000000000 013330000000 000000000000 Nepavyko suformuoti: Vienviečiu: 0 Dviviečių: 0 Triviečių: 1 Keturviečių: 2

14. Įvairūs uždaviniai, kuriuose naudojamas masyvo duomenų tipas

1. Slėpynės. Vaikai (jų buvo n) susirinko kieme ir sutarė pažaisti tradicines slėpynes. Tačiau vos tik sustojo į eilę skaičiuotei, atbėgo dar vienas $(n+1)$ -asis vaikas. Vaikai sutiko priimti jį žaisti tik su sąlyga, kad jis pirmas liks nežiūrėti ir ieškoti pasislėpusių, tačiau jis privalo dalyvauti skaičiuotėje ir likti paskutinis po skaičiuotės.

Skaičiuotė pradedama nuo pirmo stovinčio vaiko. Jeigu pasiekiamas eilės galas, tai skaičiuoti pradedama vėl nuo eilės pradžios. Iš eilės išeina vaikas, kuriam tenka skaičiuotės paskutinis žodis. Skaičiuotė toliau tęsiama nuo toliau stovinčio vaiko. Paskutinis likęs vaikas ieškos pasislėpusių.

Parašykite programą, kuri patartų, kurioje eilės vietoje turi atsistoti atbėgęs vaikas, kad po skaičiuotės jis liktų paskutinis.

Pradinių duomenų faile *Duomenys.txt* yra surašoma:

Pirmoje eilutėje yra trys sveikieji skaičiai: pradinis vaikų skaičius n ($1 < n \leq 25$), skaičiuotėje esančių žodžių skaičius k ($1 < k \leq 50$), naujai atbėgusio vaiko numeris nr . Antroje eilutėje nurodyti vaikų numeriai tokia tvarka, kokia jie sustojo į eilę skaičiuotei prieš atbėgant naujokui. Vaikų numeriai nepasikartoja ir gali būti bet kokie sveiki skaičiai.

Rezultatų faile *Rezultatai.txt* turi būti $n+1$ skaičių seka – pradinė vaikų numerių seka su reikiamoje vietoje įterptu atbėgusio vaiko numeriu.

Reikalavimai programai:

1. Duomenims saugoti naudokite vienmatį masyvą.
2. Sukurkite pradinių duomenų skaitymo funkciją void C++ (procedūrą Pascal);
3. Sukurkite funkcijas void C++ (procedūrą Pascal): įterpiančią elementą į masyvą, šalinančią elementą iš masyvo ir surandančią vietą, kur reikia įterpti;
4. Sukurkite dvimačio masyvo reikšmių rašymo į failą funkciją void C++ (procedūrą Pascal).

Duomenys	Rezultatai
5 7 13	12 16 3 2 13 1
12 16 3 2 1	

2. Pakuotės. Parduotuvėje prekiaujama miltais, supakuotais po vieną ir po du kilogramus. Kiekvienas pirkėjas perka ne mažiau kaip 1 kg ir ne daugiau kaip 50 kg miltų. Pardavėjas pirmiausia nori išparduoti didesnes miltų pakuotes. Jei pirkėjas perka daugiau kaip 1 kg miltų, pardavėjas duoda pakuotes po 2 kg ir, jei reikia, 1 kg pakuotę (pvz. perkant 5 kg miltų, duodamos dvi 2 kg ir vien 1 kg pakuotė). Kai 2 kg pakuotės baigiasi, pardavėjas pardavinėja 1 kg pakuotes.

Parenkite programą, kuri apskaičiuotų:

1. kiek 1 kg ir kiek 2 kg miltų pakuočių pardavė pardavėjas;
2. keli pirkėjai nusipirko bent vieną miltų pakuotę;
3. kiek kilogramų miltų nusipirko paskutinis pirkėjas, dar gavęs bent vieną miltų pakuotę.

Pradinių duomenų faile *Duomenys.txt* yra surašyta:

- pirmoje eilutėje nurodyta, kiek miltų pakuočių yra parduotuvėje: $n1$ ($1 \leq n1 \leq 100$) – 1 kg ir $n2$ ($1 \leq n2 \leq 100$) – 2 kg.

- antroje eilutėje užrašytas pirkėjų skaičius n ($1 \leq n \leq 100$);
- tolesnėse n eilučių nurodyta, kiek kilogramų miltų pageidauja pirkti kiekvienas pirkėjas.

Rezultatų faile *Rezultatai.txt*:

- pirmoje eilutėje nurodyti, kiek 1 kg ir 2 kg pakuočių nupirkta (jei kurios rūšies pakuočių nėra parduota, rašyti nulį);
- antroje eilutėje – keli pirkėjai nupirko bent vieną miltų pakuotę;
- trečioje eilutėje – kelis kilogramus miltų nusipirko paskutinis pirkėjas, dar gavęs bent vieną miltų pakuotę.

Reikalavimai programai:

1. pirkėjų pageidavimams saugoti naudokite vienmačius skaičių masyvus.
2. sukurkite pradinių duomenų skaitymo funkciją void C++ (procedūrą Pascal);
3. sukurkite funkciją void C++ (procedūras Pascal) suskaičiuojančią reikiamus dydžius,
4. sukurkite rezultatų rašymo į failą funkciją void C++ (procedūrą Pascal).

Duomenys	Rezultatai
1 2	1 1
2	2
1	2
3	

3. Informacijos šifravimas. Jau senų senovėje žmonėms kilo poreikis apsaugoti svarbią informaciją nuo „nesavų“ žmonių akių, neleisti konkurentams ja pasinaudoti. Itin aktualu tai buvo karyboje, prekyboje – srityse, kur, sužinojus jų paslaptis, bet koks priešininkų žingsnis galėjo atnešti nesėkmę. Todėl buvo galvojama, kaip būtų galima užšifruoti informaciją, kad ji būtų neprieinama konkurentams.

Štai vienas iš šifravimo būdų: parenkamas žodis, vadinamas raktu. Teksto, kurį reikia užšifruoti, simboliai suskirstomi į grupes. Simbolių kiekis grupėje sutampa su duoto rakto ilgiu. Paskutiniojoje grupėje gali būti mažiau raidžių.

Pavyzdžiui, duotas raktas pieva ir tekstas, kuri reikia užšifruoti „Tik nedaugelis, apžiūrėję naująjį Vilniaus prekės ženklą, pasakytų, kad tai Gedimino pilies bokšto kontūras.“

p	i	e	v	a
4	3	2	5	1
T	i	k		n
e	d	a	u	g
e	l	i	s	,
	a	p	ž	i
ū	r	é	j	ę
	n	a	u	j
a	j	i		v
i	l	n	i	a
u	s		p	r
e	k	é	s	
ž	e	n	k	l
a	,		p	a
s	a	k	y	t
ų	,		k	a
d		t	a	i
	G	e	d	i
m	i	n	o	
p	i	l	i	e
s		b	o	k
š	t	o		k
o	n	t	ū	r
a	s	.		

Rakto raidės sunumeruojamos. Mažesnę numerį turi raidė, kuri lotyniškoje abėcėlėje eina pirmiau. Tuomet parašome pirmojo stulpelio simbolius, po to antrojo, ir t. t. Gauname užšifruotą tekstą:

ng,ięjVar lataii ekkrrkaipėaiñ èn k tenlbot.idlarnjlske,a, Gii
tnsTee ū aieuežasud mpsšoa usžju ipskpykadoio ū

Parašykite programą duotam tekstui užšifruoti.

Pradiniai duomenys įrašyti faile *Duomenys.txt*. Pirmoje eilutėje įrašytas raktas, antroje – pats tekstas. Raktas sudarytas tik iš mažųjų lotyniškų raidžių, jame nėra vienodų raidžių. Raktą sudaro ne daugiau kaip 7 simboliai, teksto ilgis neviršija 150. Tekstas gali būti sudarytas iš raidžių, tarpų, skyrybos ženklų.

Rezultatą spausdinkite faile *Rezultatas.txt*.

Reikalavimai programai:

1. naudokite masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite procedūrą (funkciją void C++), rakto raidžių numeriams rasti;
4. sukurkite procedūrą (funkciją void C++), užšifruojančią tekstą;
5. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

Duomenys	Rezultatai
pieva Tik nedaugelis, apžiūrėję naują Vilniaus prekės ženklą, pasakytų, kad tai Gedimino pilies bokšto kontūras.	ng,iejVar lataii ekrkaipéain én k tenlbot.idlarnjlske,a, Gii tnsTee ū aiuežasud mpsšoa usžju ipskpykadoio ū

4. Televizijos laidos reitingas. Nuo šio pavasario TV100 pradėjo transliuoti naują pavakario laidą paaugliams. Kūrėjams labai svarbus laidos reitingas, nes reitingas parodo, kokia visuomenės visumos dalis žiūrėjo tam tikrą laidą ir reklamą.

Reitingas skaičiuojamas po kiekvienos laidos. Situacija keičiasi, po vienos laidos buvęs aukštas reitingas po kitos laidos krenta ir atvirkščiai.

Žinoma, kad laidos reitingas buvo skaičiuojamas po n laidų. Parašykite programą, kuri rastų tokį ilgiausią laidų sąrašą, kad jame reitingai tiktai didėtų arba bent pasiliktų tokie patys. Jei sprendinių gali būti keletas, užtenka rasti vieną.

Pradiniai duomenys pateikiami duomenų faile *Duomenys.txt*. Pirmoje eilutėje įrašytas laidų skaičius n ($3 \leq n \leq 100$), tolesnėse n eilučių įrašyti atitinkamų laidų reitingai: pirmoje iš šių eilučių įrašytas reitingas po pirmosios laidos, antroje – po antrosios laidos ir t. t.

Rezultatai rašomi į failą *Rezultatai.txt*. Pirmoje eilutėje įrašykite rastojo ilgiausio laidų sąrašo ilgį, tolesnėse į šį sąrašą įeinančių laidų numerius.

Reikalavimai programai:

1. naudokite sveikųjų skaičių masyvą(-us);
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite procedūrą (funkciją void C) maksimaliam laidų kiekiui ir laidoms rasti;
4. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

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

5. Automobiliai. Prekybos įmonė darbu turi įsigijusi tam tikrą kiekį automobilių. Bėgant laikui automobiliai susidėvi, pradeda gesti, todėl įmonės vadovas remdamasis automobilių charakteristikomis, kurių sąrašą kiekvienų metų pradžioje sudaro vairuotojai, priima sprendimą kai kuriuos jų parduoti. Taip pat kiekvienais metais įmonei nuperkamas tam tikras kiekis naujų arba dėvėtų automobilių. Automobilio charakteristika apibūdinama jo pavadinimu, nuvažiuota rida ir dienų kiekiu, kuri automobiliu negalima buvo dirbti dėl remonto. Automobilis parduodamas tada, jei nuvažiuota rida arba remonto dienų kiekis viršija maksimaliai nustatytas reikšmes. Parašykite programą, leidžiančią papildyti įmonės automobilių sąrašą naujais automobiliais, o taip pat iš įmonės automobilių sąrašo pašalinti parduodamus automobilius.

Duomenų failo *Duomenys.txt* pirmoje eilutėje duota maksimali rida ir remonto dienų kiekis, kuriuos viršijus – automobilis yra parduodamas, antroje - įmonės turimų automobilių kiekis ir jų sąrašas. Toliau duomenų faile yra duodamas įmonės perkamų automobilių kiekis ir perkamų automobilių sąrašas.

Rezultatų faile *Rezultatai.txt* turi būti išvedamas po automobilių pirkimo ir pardavimo suformuotas naujas įmonės automobilių sąrašas, o taip pat išvedamas ribinės ridos automobilių sąrašas, kur ribinė rida – tai rida, didesnė arba lygi 85% maksimaliai nustatytos ridos.

Reikalavimai programai:

1. Sukurti funkcijas, papildančias ar šalinančias automobilių sąrašo elementus, o taip pat funkciją, sudarančią ribinės ridos automobilių sąrašą.
2. Sukurti duomenų skaitymo iš failo ir rezultatų rašymo į failą funkcijas.
3. Sąrašams saugoti reikia naudoti masyvus.

Duomenys	Rezultatai
45000 25	Naujas automobilių sąrašas:
7	Volkswagen_Polo 44000 24
Volkswagen_Polo 44000 24	Opel_Meriva 29000 24
Opel_Meriva 29000 24	Fiat_Panda 39000 23
Fiat_Panda 46000 5	Citroen_Berlingo 21000 24
Fiat_Panda 39000 23	Volkswagen_Polo 20000 0
Citroen_Berlingo 21000 24	Skoda_Fabia 40000 0
Opel_Astra 70000 26	Volkswagen_Golf 30000 0
Fiat_Ducato 90000 45	
3	Ribinės ridos automobiliai:
Volkswagen_Polo 20000 0	Volkswagen_Polo 44000 24
Skoda_Fabia 40000 0	Fiat_Panda 39000 23
Volkswagen_Golf 30000 0	Skoda_Fabia 40000 0

15. Simboliai

1. Balsės A ir E. Petriukas nutarė atlikti tyrimą, kuri balsė „a“ ar „e“ dažniau pasikartoja tekste. Pasirinkęs tekstą suskaičiavo, kelis kartus kiekviena balsė pasikartoja tekste. Parašykite programą, kuri suskaičiuotų, kelios tekste yra balsės „a“ ir kelios „e“, kuri balsė ir kiek procentų dažniau pasikartoja tekste. Pradiniai duomenys įrašyti faile *Duomenys.txt*. Rezultatai įrašomi į failą *Rezultatai.txt*.

Reikalavimai programai:

1. sukurkite pradinių duomenų skaitymo funkciją;
2. sukurkite balsių kiekio skaičiavimo funkciją;
3. sukurkite rezultatų rašymo į failą funkciją.

Duomenys	Rezultatai
Šiandien yra labai gražus orelis Kaune.	Raidė a pasikartoja tekste 6 kartus. Raidė e pasikartoja tekste 3 kartus. Raidė a pasikartoja 50 % dažniau.

2. Linksmas sukeitimas. Penktokai rašydami rašinėlius labai dažnai sukeičia gretimas raides vietomis. Tai pastebėjusi mokytoja nusprendė su vaikais pažaisti žaidimą, pagrįstą dviejų gretimų raidžių sukeitimu: kiekviename žodyje visos raidės suporuojamos ir poros raidės sukeičiamos vietomis. Paskutinė žodžio raidė gali likti ir be poros, jei žodyje yra nelyginis raidžių skaičius ir jina tuomet lieka savo vietoje. Pvz., sukeitus raides vietomis žodyje KOMPIUTERIS, gaunamas žodis OKPMUIETIRS.

Parašykite programą, kuri pagal aprašytą algoritmą iš duomenų faile *Duomenys.txt* įrašyto žodžio suformuotų ir į rezultatų failą *Rezultatai.txt* įrašytų naujai suformuotą žodį.

Reikalavimai programai:

1. naudokite char simbolius ar jų masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite dviejų gretimų raidžių sukeitimo procedūrą (funkciją void C++);
4. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

Duomenys	Rezultatai
informatika	niofmrtakia

3. Kodavimas. Telefono numerio skaitmenys (išskyrus 0 ir 1) koduojami trimis lotyniškos abėcėlės raidėmis:

- 2 ABC
- 3 DEF
- 4 GHI
- 5 JKL
- 6 MNO
- 7 PQR
- 8 STU
- 9 VWX

Septynių skaitmenų telefono numeriai yra siunčiami pasirinktu mainų kanalu. Kartais gavėjas dėl trikdžių gauna ne visai tikslus skaičių kodus. Iš trijų vieno kodo raidžių klaidinga būna tik viena.

Užduotis. Ištaisykite perdavimo klaidas, nurodykite klaidų skaičių ir atkoduokite telefono numerį.

Duomenys. Duomenų failo *Duom.txt* pirmoje eilutėje įrašytas perduodamų telefono numerių skaičius n . Likusiose n eilučių įrašyti perduoti telefono numerių kodai. Duomenys pateikti teisingai, t.y nėra aštuonių numerio skaitmenų kodų ir pan.

Rezultatai. Rezultatų faile *Rez.txt* išvedamas ištaisytas kodas. Išvedamas žodis *Klaidos:*, po to klaidų skaičius. Išvedamas žodis *Tel.nr.:*, po to – atkoduotas telefono numeris.

Duomenys	Rezultatai
3	ABCABCDEFDEFGHIGHIGHI Klaidos: 0 Tel. nr.:2233444
ABCABCDEFDEFGHIGHIGHI	ABCDEFMNOPQRSTUVWXYZ Klaidos: 4 Tel. nr.:2367898
AB B DEF N NO P RR S TUV V X S TU	VWXSTUPQRMNOJKLGHIDEF Klaidos: 7 Tel. nr.:9876543
VW Y ST X PR M MO K KLGH H DB F	

Reikalavimai programai:

1. naudokite char simbolius ar jų masyvus;
2. sukurkite funkciją, kuri nustato, ar kode nėra klaidų;
3. sukurkite kodą taisančią procedūrą arba funkciją;
4. sukurkite skaičiaus atkodavimo funkciją.

4. Kentauras. Žiniukui atsibodo žaisti šachmatais, todėl jis nutarė paskaityti Homero „Iliadą ir Odisėją“. Netrukus jis surado mišlingą personažą Kentaurą ir nutarė, kad būtent tokios figūros trūksta šachmatuose. Kentauras yra mitologinis graikų gyvūnas, turintis žirgo kūną ir žmogaus galvą bei liemenį.

Žiniukas norėjo turėti labai stiprią šachmatų figūrą, kurios ėjimai būtų panašūs į žirgo, tik langelių skaičius nebūtų ribojamas (žirgas gali eiti 2 langelius pirmyn ir vieną į šoną, arba 1 langelį pirmyn ir du į šoną).

Šachmatų figūros padėtis lentoje nusakoma raide ir skaičiumi. Paveikslėlyje pavaizduotos Kentauro padėtys G1 ir E5. Jeigu tai būtų jo šuolis, tai galima būtų apskaičiuoti to šuolio ilgį: 4 langeliai pirmyn ir 2 į šoną. Šuolio ilgis yra 6.

Parenkite programą, kuri apskaičiuotų ilgiausią Kentauro šulį.

- Pradinių duomenų faile *Duomenys.txt* įrašyta:

- pirmojoje eilutėje – užfiksuotų Kentauro padėčių šachmatų lentoje skaičius n ;
- tolesnėse n eilučių yra po du simbolius – raidę ir skaičių. Tai Kentauro padėtys šachmatų lentoje. Tarp raidės ir skaičiaus tarpo nėra.
- Apribojimai: $2 \leq n \leq 100$. Šachmatų lenta – standartinė.

Rezultatų faile *Rezultatai.txt* įrašykite vienintelį skaičių – ilgiausią Kentauro šuolį.

Reikalavimai programai:

- naudokite char simbolius;
- sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
- sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++);
- sukurkite funkciją, kuri skaičiuoja Kentauro vieno šuolio ilgį.

Duomenys	Rezultatai
3 A1 B4 G1	8

5. Būdvardžių linksniuotės. Tekstiniame faile *Duomenys.txt* yra pateikta n žodžių eilučių. Kiekvienoje iš n eilučių įrašytas būdvardžio vienaskaitos ir daugiskaitos vardininkas. Parašykite programą, kuri nustatytų kiekvieno būdvardžio linksniuotę ir rezultatų faile *Rezultatai.txt* pateiktų statistiką: po kiek buvo kiekvienos linksniuotės būdvardžių: pirmoje failo eilutėje – I, antroje – II, trečioje – III linksniuotės būdvardžių skaičius.

Lietuvių kalboje yra trys būdvardžių linksniuotės, žr. lentelę:

I linksniuotė		II linksniuotė		III linksniuotė	
Vyr. g.	Mot. g.	Vyr. g.	Mot. g.	Vyr. g.	Mot. g.
-as	-a	-us	-i	-is -ys	-ė

Reikalavimai programai:

1. sukurkite pradinių duomenų skaitymo funkciją;
2. sukurkite skaičiavimo funkciją;
3. sukurkite rezultatų rašymo į failą funkciją.

Duomenys	Rezultatai
5	2
Gražus gražūs	2
Naujas nauji	1
Senas seni	
Didelis dideli	
Jaukus jaukūs	

16. Simbolių eilutės

1. Žodžių paieška. Faile *Duomenys.txt* įrašytas tekstas. Pirmoje failo eilutėje įrašytas žodis, kurį norima rasti tekste, kitose eilutėse – tekstas, kuriame turi būti atliekama žodžio paieška. Reikia sudaryti algoritmą, kuris suskaičiuotų, kiek kartų tekste pasikartoja ieškomas žodis. Rezultatus pateikti *Rezultatai.txt* faile.

Reikalavimai programai:

1. Sukurkite pradinių duomenų skaitymo funkciją;
2. Sukurkite žodžio paieškos funkciją;
3. Sukurkite rezultatų įrašymo funkciją.

Duomenys	Rezultatai
Labai Labai labai seniai, vienoje šalyje, Limandijoje, gyveno labai mažas berniukas Adrianas. Šeima, kurioje jis gyveno, buvo labai didelė ir neturtinga. Net Kalėdų šventei šeima neturėjo pinigų duonai nupirkti.	4

2. Simbolių eilutės: Skirtingi Simboliai. Duotos dvi simbolių eilutės A ir B. Reikia sudaryti algoritmą randantį simbolius, kurie yra eilutėje A, bet kurių nėra eilutėje B. Tekstas sudarytas iš didžiųjų ir mažųjų raidžių (jos laikomos skirtingomis), bei skaitmenų. Simboliai gali kartotis. Jei visi simboliai kurie yra eilutėje A taip pat yra ir eilutėje B, spausdinti atsakymą “Taip”, kitu atveju spausdinti visus trūkstamus simbolius, atskirtus tarpais. Vienodų simbolių rezultatų faile spausdinti negalima.

Patarimas: Simbolių eilutes string galima naudoti kaip masyvus.

Kiekybiniai ribojimai. Eilutės nebus ilgesnės nei 100 simbolių ir trumpesnės nei 1 simbolis, jose nebus tarpų.

Reikalavimai programai:

1. Naudokite string simbolių eilutes/simbolių masyvus.
2. Pradinius duomenis įvesti iš failo (input.txt), rezultatus taip pat rašyti į failą (output.txt).
3. Sukurkite funkciją, kuri iš patikrina ar simbolis yra eilutėje/simbolių masyve.
4. Rezultatų faile visi simboliai turi būti unikalūs.

Duomenys	Rezultatai
abcdefg gaefa	b c d
gaefa abcdefg	Taip
aaaaaab b	a
1234 13579	2 4
1122334455 123456	Taip

3. Žodžiai palindromai. Palindromas yra žodis arba žodžių eilutė, kuri iš abiejų galų skaitoma vienodai. Pavyzdžiui, "SĖDĖK UŽU KĖDĖS".

Užduotis. Parašykite programą, kuri ištirtų n ($1 \leq n \leq 50$) teksto eilučių ir nustatytų, kurios iš jų yra palindromai.

Pastabos:

- Teksto eilutėse nėra skyrybos ženklų, o tarpai paliekami ten, kur jų reikia.
- Teksto eilutėse didžiosios ir mažosios raidės laikomos vienodomis.
- Simbolių skaičius eilutėje neviršija 80.

Pradiniai duomenys įrašyti faile *duom.txt*. Pirmoje failo eilutėje įrašytas teksto eilučių skaičius n . Žemiau įrašytos teksto eilutės.

Rezultatai rašomi į tekstinį failą *rez.txt*.

Reikalavimai programai:

1. Sudaryta procedūra (funkcija), kuri iš failo perskaito teksto eilutes.
2. Sudaryta funkcija, kuri iš teksto eilutės pašalina tarpo ženklus.
3. Sudaryta funkcija, kuri nustato, ar teksto eilutė yra palindromas.
4. Sudaryta procedūra (funkcija) kuri rašo rezultatus į tekstinį failą.

Duomenys	Rezultatai
6 Kol einu šunie lok Aš einu Madam Argi ten ne TIGRA Mama nori Ir Ona nori	Kol einu šunie lok Madam Argi ten ne TIGRA Ir Ona nori

4. Kalėdinė Eglutė. Prieš pat Kalėdas Jurgiuko tėtis atnešė į namus gražią eglutę. Jurgiukas kaip tik kūrė sveikinimus savo draugams. Sveikinimų buvo sugalvojęs daug ir labai gražių, tačiau norėjosi išrasti nematytą sveikinimo pateikimo formą. Jam staiga kilo idėja Kalėdų sveikinimus surašyti eglute. Tam reikia sugalvoti taisykles, pagal kurias iš paprasto teksto gautume eglutę:

- Pirmojoje eglutės eilutėje bus vienas žodis.
- Kitose – kaskart vis vienu žodžiu daugiau, kol užteks teksto duotojoje eilutėje.
- Likusi nepilna duotosios eilutės dalis sudarys naują eilutę.
- Taip pertvarkoma kiekviena duotojo teksto eilutė.
- Be to, tekstas pertvarkytose eilutėse patraukiamas tai į vieną, tai į kitą pusę taip, kad eglutės kamienui liktų tik vienas simbolis (žiūrėkite pavyzdžius).

Jurgiukas prisiminė, kad visus sveikinimus turi tekstinuose failuose. Kaip būtų gerai parašyti programą, kuri už jį visus sveikinimus surašytų eglute. Padėkite Jurgiukui, juk iki Kalėdų jau ne tiek daug laiko beliko.

Duomenys. Pirmojoje duomenų failo *Duomenys.txt* eilutėje yra teksto eilučių skaičius n . Likusiose n eilučių yra tekstas. Tekste gali būti skyrybos ženklai, kurie pagal rašybos taisykles glaudžiami prie žodžių pabaigos.

Rezultatai. Tekstas spausdinamas į tekstinį failą *Rezultatai.txt* eglutės forma. Kairėje pusėje užpildant reikiamu tarpo simbolių skaičiumi.

Kiekybiniai ribojimai. Duotasis tekstas sudarytas ne daugiau kaip iš 10 ($n \leq 10$) eilučių. Pertvarkytas tekstas turėtų užimti ne daugiau kaip 100 eilučių. Kiekvienoje duotojo teksto eilutėje ne daugiau kaip 255 simboliai.

Reikalavimai programai:

1. naudokite *string* simbolių eilutes/eilučių masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++);
4. sukurkite funkciją, kuri iš duoto teksto suformuoja eglute išdėstytą tekstą.

Duomenys	Rezultatai
<pre>4 Ei, drauguži, žiemos vidury Roges jei turi, tai gerai! Į kalną aukščiausią, patį balčiausią Įkopt jei gali, tai gerai!</pre>	<pre> Ei, drauguži, žiemos vidury Roges jei turi, tai gerai! Į kalną aukščiausią, patį balčiausią Įkopt jei gali, tai gerai!</pre>

<pre>2 A b c d e f g h i j k O mano eglutė gražesnė!</pre>	<pre> A b c d e f g h i j O mano eglutė gražesnė!</pre>
--	--

5. Olimpinės plaukimo varžybos. Olimpinėse žaidynėse individualių sporto šakų rezultatai skelbiami taip: sportininko vardas ir pavardė, šalies, kuriai atstovauja sportininkas, santrumpa ir sportininko pasiektas rezultatas.

Tarptautinis Olimpinis Komitetas oficialiai naudoja šalių pavadinimų santrumpas iš trijų raidžių, pvz.: LTU – Lietuvos Respublika, PNG – Papua Naujoji Gvinėja ir t.t.

Lina kūno kultūros pamokai turi parengti statistinius duomenis apie Londono olimpiados plaukimo varžybas. Interneto svetainėje <http://www.olympic.org/olympic-results/london-2012/swimming> ji surado visų plaukimo rungčių rezultatus. Surastus rezultatus mergina planuoja perkelti į pateiktą. Savo sumanymu Lina pasidalino su klasės draugais. Draugai jai patarė į pateiktą dėti ne šalies santrumpą, o pilną pavadinimą. Kad darbas vyktų sparčiau, santrumpos keitimui pilnu pavadinimu reikia sukurti programą.

Tekstinio failo *salys.txt* pirmoje eilutėje Lina įrašė šalių skaičių *n*. Tolesnėse *n* failo eilučių įrašyta: šalies santrumpa, kurią sudaro trys raidės ir pilnas šalies pavadinimas, kuri gali sudaryti iki 7 žodžių (žr., http://lt.wikipedia.org/wiki/Sarasas:Šalių_sarasas), šalys išdėstytos atsitiktine tvarka.

Kitame tekstiniame faile *plaukikai.txt* surašyti vienos rungties 8 geriausių plaukikų vardai ir pavardės (skiriama 20 pozicijų), šalių santrumpos ir pasiekti rezultatai. Vienam plaukikui skiriama viena eilutė.

Rezultatų faile *galutinis_sarasas.txt* reikia suformuoti naują sąrašą, kuriame būtų plaukikų vardai ir pavardės (skiriama 20 pozicijų), pilni šalių pavadinimai ir pasiekti rezultatai.

Reikalavimai programai:

1. naudokite simbolių eilutes ir simbolių eilučių masyvus;
2. sukurkite pradinių duomenų apie šalis skaitymo procedūrą (funkciją void C++);
3. sukurkite pradinių duomenų apie plaukikų rezultatus skaitymo procedūrą (funkciją void C++);
4. sukurkite funkciją, kuri iš santrumpos gražina pilną šalies pavadinimą;
5. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

Tekstinio failo *salys.txt* fragmentas (failą su visų šalių pavadinimais parsisiųsite spragtelėję [nuorodą](#))

202

AFG Afganistano Islamo Valstybė
IRL Airijos Respublika
ALB Albanijos Respublika
ALG Alžyro Liaudies Demokratinė Respublika
ASA Rytų Samoa
....
LTU Lietuvos Respublika
....
VIE Vietnamo Socialistinė Respublika
GER Vokietijos Federacinė Respublika
ZAM Zambijos Respublika
ZIM Zimbabvės Respublika
CPV Žaliojo Kyšulio Respublika

Duomenys			Rezultatai		
Ruta MEILUTYTE	LTU	1:05.47	Ruta MEILUTYTE	Lietuvos Respublika	1:05.47
Rebecca SONI	USA	1:05.55	Rebecca SONI	Jungtinės Amerikos Valstijos	1:05.55
Satomi SUZUKI	JPN	1:06.46	Satomi SUZUKI	Japonija	1:06.46
Alia ATKINSON	JAM	1:06.93	Alia ATKINSON	Jamaika	1:06.93
Leisel JONES	AUS	1:06.95	Leisel JONES	Australijos Sandrauga	1:06.95
Breeja LARSON	USA	1:06.96	Breeja LARSON	Jungtinės Amerikos Valstijos	1:06.96
Iuliia EFIMOVA	RUS	1:06.98	Iuliia EFIMOVA	Rusijos Federacija	1:06.98
Rikke PEDERSEN	DEN	1:07.55	Rikke PEDERSEN	Danijos Karalystė	1:07.55

17. Struktūros (įrašai). Struktūrų (įrašų) masyvai

1. Oro temperatūros matavimai. Faile *Duomenys.txt* pateikti oro temperatūros matavimai. Pirmoje eilutėje įrašyti 2 sveikieji skaičiai: kelių mėnesių duomenys ir kiek kartų matuota oro temperatūra kiekvieną mėnesį. Tolesnėse eilutėse pateikti kiekvieno mėnesio matavimai. Parašykite programą, skaičiuojančią kiekvieno mėnesio vidutinę temperatūrą ir bendrą vidutinę temperatūrą.

Reikalavimai programai:

1. duomenims saugoti naudokite įrašo (struktūros) duomenų tipą ir įrašų (struktūrų) masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite vidurkio skaičiavimo funkciją;
4. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

Duomenys		Rezultatai	
3	6	Sausis	-10.13
Sausis	-11.5 -10.3 -12 -6.5 -5 -15.5	Vasaris	-9.22
Vasaris	-16 -14.5 -11 -10.3 -3.5 0	Kovas	3.37
Kovas	-3 -2.5 0 5 9.5 11.2	Bendras matavimų vidurkis:	-5.33

2. Įnoringiausia princesė. Karalius turėjo dukrą. Niekaip jis negalėjo jos išleisti nors už kokio princo, nes dukra vis rasdavo siūlomame prince ypatybę, dėl kurios jį atstumdavo.

Kartą karalius nutarė išleisti savo dukrą už aukščiausio ūgio princo. Tačiau princesė į karaliaus norą atsakė: „Taip, tekėsiu. Mano vyras bus labai aukštas, tačiau ne pats aukščiausias, privalo bendrauti mažiausiai trimis užsienio kalbomis ir mokėti fechtuotis“. Buvo surašyti pretendentų duomenys ir pateikti Karaliui. Karalius iš karto atmetė nemokančius fechtuotis ir negalinčius bendrauti bent trimis užsienio kalbomis, tačiau sąrašas vis tiek liko išpūdingo dydžio, nes princesė buvo nepaprastai graži. Už kurio princo tekėtų princesė? Parašykite programą, kuri spręstų šį uždavinį.

Duomenys. Pirmojoje duomenų failo eilutėje yra princų skaičius n , o kitose n eilučių - jų duomenys: vardas (iki 24 simbolių, gali būti tarpo simbolių) ir ūgis (pėdomis ir coliais).

Rezultatai. Spausdinamas tinkamo princo vardas arba skaičius 0, jei nei vienas princas netenkina princesės iškeltų sąlygų.

Kiekybiniai ribojimai. Pretenduojančių princų gali būti nuo 1 iki 50.

Reikalavimai programai:

1. naudokite string simbolių eilutes arba char masyvus;
2. sudarykite struktūrą ir masyvą, skirtą princo duomenims saugoti;
3. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
4. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++);
5. sukurkite funkciją, randančią tinkamiausią princa.

<i>princai-duom.txt</i>	<i>princai-rez.txt</i>
5 Joachimas Hamburgietis 6 5 Kurtas Roterdamiētis 5 3 Horstas Berlynietis 5 7 Otto Bremenietis 5 2 Hansas Liubekietis 5 3	Horstas Berlynietis
1 Joachimas Hamburgietis 6 5	0

3. Padangų žymėjimas. Nepriklausomai nuo to, kas gamina padangas, pagal įstatymus jų šone būtinai turi būti standartinis žymėjimas. Pavyzdžiui, užrašas 185 65 HR13 šifruojamas taip: 185 – Atstumas (milimetrais) nuo vieno padangos krašto iki kito, kai padanga neapkrauta. Šis dydis vadinamas padangos pločiu.

65 – Šoninės sienelės aukštis, išreikštas procentais nuo padangos pločio, kai padanga neapkrauta.

H — Greičio indeksas. Jis rodo, koku maksimaliu greičiu galima važiuoti nepertraukiamą 10 minučių laikotarpį. Viršijus šią ribą, padanga, veikiamą išcentrinės jėgos, gali plyšti. (H – greitis 210 km/h)

R – Ši raidė rodo, kad padangos konstrukcija – radialinė. Jei raidės „R“ nėra, padangos konstrukcija yra diagonalinė.

13 — Šis colinis dydis rodo, kokio skersmens ratlankiams skirta padanga.

Klientas nurodo padangų atstumą, aukštį, greičio indeksą, padangos konstrukciją ir ratlankio skersmenį. Sudarykite programą, kuri suskaičiuotų kiek sandėlyje yra klientui reikalingų padangų ir kiek jos visos kainuotų.

Duomenų faile *duom.txt* pirmoje eilutėje įrašyti kliento padangos duomenys: atstumas, aukštis, greičio indeksas, padangos konstrukcija ir ratlankio skersmuo.

Antroje eilutėje sandėliuojamų padangų skaičius n ($1 \leq n \leq 500$).

Kitose failo eilutėse turimų padangų duomenys.

Rezultatai. Pirmoje failo *rez.txt* eilutėje už teksto „Tinkamų padangų skaičius:“ rašomas rastų tinkamų padangų skaičius. Antroje failo eilutėje už teksto „Jos kainuotų:“ įrašoma kokią pinigų sumą kainuotų tinkamos padangos ir valiutos simbolis „Lt“. Jei tinkamų padangų nėra, tuomet rezultatų faile turi būti įrašytas žodis Nėra.

<i>duom.txt</i>	<i>rez.txt</i>
185 65 HR13 5 185 65 HR13 201.50 225 45 H 17 180.50 185 65 HR13 178.00 225 45 HR17 198.50 185 65 HR13 199.99	Tinkamų padangų skaičius: 3 Jos kainuotų: 579.49 Lt

Reikalavimai programai:

1. duomenims saugoti naudokite įrašo (struktūros) duomenų tipą ir įrašų (struktūrų) masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite funkciją, nustatančią, ar du įrašai sutampa;
4. sukurkite procedūrą (funkciją void C++), skaičiuojančią tinkamų padangų kiekį ir bendrą jų kainą;
5. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

18. Veiksmai su struktūrų (įrašų) masyvų elementais

1. Prekyba elektronikos prekėmis. Parduotuvė prekiauja buitine elektronika. Parduodama n ($3 \leq n \leq 100$) modelių aparatų. Modelis nusakomas numeriu $[1..n]$, pavadinimu (18 simbolių) ir kaina. Atlikta p ($3 \leq p \leq 1000$) pardavimų fiksuojant modelį ir parduotų aparatų skaičių. Apskaičiuokite: kiek vienetų parduota populiariausio modelio; kiek surinkta pinigų už populiariausią modelį; kiek vienetų parduota pelningiausio modelio; kiek buvo surinkta už pelningiausią modelį. Kuris modelis yra mažiausiai populiarus? Kiek parduota jo vienetų ir kiek už jį buvo surinkta pinigų. Net ir mažiausiame duomenų faile nebus vienodai vertinamų modelių.

Duomenys. Duomenys tekstiniame faile *Duomenys.txt* pateikti tokiu būdu:

- pirmojoje eilutėje įrašytas prekių skaičius n ;
- kitose n eilučių įrašyta prekės numeris, pavadinimas ir kaina;
- po to įrašytas pardavimų skaičius p ;
- toliau yra dar p duomenų eilučių. Kiekvienoje po du skaičius: prekės numeris ir parduotas tos prekės kiekis.

Rezultatai. Rezultatai rašomi į tekstinį failą *Rezultatai.txt* blokais. Iš viso turėtų būti suformuoti trys analogiški blokai atitinkamai populiariausiai, pelningiausiai ir mažiausiai populiariai prekei.

Kiekybiniai ribojimai. Skirtingų prekių skaičius nėra mažesnis už 3 ir neviršija 100. Pardavimų skaičius nėra mažesnis už 3 ir neviršija 1000. Kainos ir kiekiai nėra labai dideli skaičiai, nereikia imtis specialių priemonių, kad išvengtų perpildymo, ar galimai susikaupusių skaičiavimo realiaisiais skaičiais paklaidų įvertinimo.

Reikalavimai programai:

1. naudokite struktūras (įrašus) ir jų masyvus, sveikuosius ir realiuosius skaičius;
2. sudarykite struktūrą (įrašą) ir masyvą, skirtą prekės duomenims saugoti;
3. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
4. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++);
5. sukurkite funkcijas, randančias populiariausią, pelningiausią ir mažiausiai populiarią prekę.

Duomenys	Rezultatai
3	Populiariausias: Samsunk XT
2 Samsunk XT Pro 789.99	Pro
6 Sonia Bony M 1237.89	Parduota: 16 vnt.
3 Celestia No 1 666.99	Surinkta: 12639.84 Lt
5	
2 7	Pelningiausias: Sonia Bony M
6 4	Parduota: 11 vnt.
3 1	Surinkta: 13616.79 Lt
2 9	
6 7	Mažiausiai populiarius:
	Celestia No 1
	Parduota: 1 vnt.
	Surinkta: 666.99 Lt

2. Gimtadieniai. Auklėtoja savo mokiniams pasiūlė, bendraklasius suskirstyti į grupes – rudens, žiemos, pavasario, vasaros – ir gimtadienius švęsti vieną kartą per ketvirtį.

Sudarykite programą, kuri padėtų suskirstyti mokinius į grupes pagal gimimo datą: rudens, žiemos, pavasario ir vasaros. Taip pat rastų vyriausią ir jauniausią klasės mokinį.

Duomenų failo *mokiniai.txt* pirmoje eilutėje įrašytas klasės mokinių skaičius n ($5 \leq n \leq 36$). Kitose n eilučių įrašytas mokinio vardas ir pavardė (tam skirti 25 simboliai) ir gimimo data: metai, mėnuo ir diena – atskirti brūkšneliais.

Rezultatų faile *grupes.txt* įrašykite tekstą: „Rudens grupė:“ ir žemiau pateikite mokinių sąrašą su gimimo datomis. Toliau įrašykite tekstą: „Žiemos grupė:“ ir žemiau pateikite mokinių sąrašą su gimimo datomis. Ir t.t. Jeigu kurioje nors grupėje nebus mokinių vietoj sąrašo rašykite tris brūkšnius. Grupių gale parašykite „Vyriausias klasės mokinytis:“ ir jo vardą bei pavardę. Paskutinėje eilutėje – „Jauniausias klasės mokinytis:“ ir jo vardą bei pavardę.

Duomenys		Rezultatai	
10		Rudenio grupė	
Kiaunytė Lapytė	2000-12-2	Lokys Pikšrys	2000-10-5
Lokys Pikšrys	2000-10-0	Vilija Ženislovė	2000-9-3
Sukas Barsukas	2001-01-1		
Kiškė Parimiškė	2000-08-0	Žiemos grupė	
Vilkas Violetinis	2001-02-1	Kiaunytė Lapytė	2000-12-23
Vikšras Rajūnas	2000-07-2	Sukas Barsukas	2001-1-16
Vilija Ženislovė	2000-09-0	Vilkas Violetinis	2001-2-15
Jonas Ilgakakliūnas	2001-01-1	Jonas Ilgakakliūnas	2001-1-10
Renaldas Švilpikutis	2000-06-0	Kiaunytė Pameilutytė	2000-12-14
Kiaunytė Pameilutytė	2000-12-1		
		Pavasario grupė	
		- - -	
		Vasaros grupė	
		Kiškė Parimiškė	2000-8-5
		Vikšras Rajūnas	2000-7-23
		Renaldas Švilpikutis	2000-6-4
		Vyriausias klasės mokinys: Renaldas Švilpikutis	
		Vyriausias klasės mokinys: Vilkas Violetinis	

Reikalavimai programai:

1. Sukurta funkcija (procedūra) duomenims iš failo nuskaityti. Duomenys skaitomi į struktūros (įrašo) tipo masyvą.
2. Sukurta funkcija (procedūra) duomenims atrinkti į vieną grupę. Ta pati funkcija (procedūra) naudojama ir kitoms grupėms formuoti. Formuojamos grupės įrašomos į naują struktūros (įrašo) tipo masyvą.
3. Sukurta funkcija vyriausiam mokiniui rasti.
4. Sukurta funkcija jauniausiam mokiniui rasti.
5. Sukurta funkcija (procedūra) rezultatams į duomenų failą įrašyti.

3. Mokinių vidurkiai. Faile *Duomenys.txt* pateiktas mokinių sąrašas su mokomųjų dalykų vidurkiais. Pirmoje eilutėje nurodytas mokinių skaičius. Kitose eilutėse - mokinio vardas, pavardė, klasė, dalykų skaičius ir likę skaičiai – tų dalykų vidurkiai. Sukurkite programą, kuri suskaičiuotų kiekvieno mokinio visų dalykų vidurkį (apvalinant iki 1 skaičiaus po kablelio ir papildomai pateikiant suapvalinus iki sveikosios dalies) bei sugrupuotų mokinius pagal klases. Sąrašo pabaigoje pateikti dviejų mokinių, turinčių didžiausią ir mažiausią vidurkius, vardus, pavardes ir klases. Jei yra po kelis mokinius, turinčius didžiausią ir mažiausią vidurkius, tuomet reikia pateikti pirmųjų sąrašė esančių mokinių su atitinkamais vidurkiais, duomenis. Rezultatus pateikti faile *rezultatai.txt*.

Reikalavimai programai:

1. duomenims saugoti naudoti struktūras ir struktūrų masyvus;
2. galutiniams rezultatams saugoti sukurti naują struktūrų masyvą;
3. sukurkite pradinių duomenų iš failo skaitymo funkciją;
4. sukurkite vidurkio skaičiavimo funkciją;
5. sukurkite rezultatų įrašymo į failą funkciją.

Duomenys	Rezultatai
8	
Petras Rūstusis 5a 6 8 7 8 8 9 9	Klasė 5a:
Jonas Juodasis 4b 5 9 9 8 10 10	Petras Rūstusis 8.2 8
Ona Balsytė 5a 6 9 9 10 9 9 10	Ona Balsytė 9.3 9
Juozas Klonis 4b 5 10 10 9 9 9	Adelė Luknytė 8.3 8
Žydrūnas Melsvutis 4a 5 9 8 8 8 9	Algis Beržinis 9.2 9
Adelė Luknytė 5a 6 7 8 10 8 8 9	
Algis Beržinis 5a 6 9 9 8 10 10 9	Klasė 4b:
Jonė Juodytė 4a 5 9 9 9 8 10	Jonas Juodasis 9.2 9
	Juozas Klonis 9.4 9
	Klasė 4a:
	Žydrūnas Melsvutis 8.4 8
	Jonė Juodytė 9 9
	Didžiausias vidurkis: Juozas Klonis iš 4b
	Mažiausias vidurkis: Petras Rūstusis iš 5a

4. Šviesuolių miestelio skaitytojai. „Šviesuolių“ miestelį aptarnaujantis laiškininkas ne tik išnešioja gyventojų gautus laiškus, bet ir jų prenumeruojamą spaudą. Metų pradžioje laiškininkas gavo leidimą organizuoti akciją „Prenumeruok didesnę kiekį – mokėsi tris kartus pigiau“. „Šviesuolių“ miestelio gyventojai ne tik šviesuoliai, bet ir labai praktiški žmonės sugalvojo, kad žmogus, užsiprenumeravęs kelis tokio paties pavadinimo leidinius, galės vieną pasilikti sau, o kitus išdalinti kaimynams, kitas žmogus prenumeruoja kito pavadinimo kelis leidinius, vieną pasilieka sau, o kitus išdalina kaimynams ir t.t. Kaip tarė, taip padarė.

Pirmoje tekstinio failo *sviesuoliai.txt* eilutėje įrašytas prenumeratorių skaičius n ($3 \leq n \leq 20$) ir prenumeruojamų leidinių skaičius k ($3 \leq k \leq 10$). Antroje failo eilutėje surašytos leidinių kainos, viena nuo kitos atskirtos tarpais. Kainos yra realieji skaičiai. Toliau faile yra n eilučių, kuriose išvardinti miestelio gyventojų vardai (skiriama 15 pozicijų), jų prenumeruojamų leidinių kiekiai nuo pirmojo iki k -tojo. Jei kurio nors leidinio žmogus neprenumeruoja, toje vietoje įrašytas nulis. Parašykite programą, kuri tekstiniam faile *ataskaita.txt* surašytų:

- pirmosiose n failo eilučių surašoma, už kokią pinigų sumą kiekvienas žmogus užsiprenumeravo leidinių;
- už kokią pinigų sumą leidinių užsiprenumeravo visi „Šviesuolių“ miestelio gyventojai;
- kuris „Šviesuolių“ miestelio gyventojas užsiprenumeravo daugiausiai leidinių;
- kuris „Šviesuolių“ miestelio gyventojas užsiprenumeravo leidinių už mažiausią pinigų sumą.

Reikalavimai programai:

1. naudokite įrašo (struktūros) duomenų tipą ir masyvus su įrašo (struktūros) tipo duomenimis;
2. turi būti funkcija, skaičiuojanti vieno žmogaus išleistą leidiniams pinigų sumą;
3. turi būti funkcija, skaičiuojanti visą leidiniams išleistą pinigų sumą;
4. turi būti funkcija, surandanti, kuris miestelio gyventojas užsiprenumeravo daugiausiai leidinių;
5. turi būti funkcija, surandanti, kuris „Šviesuolių“ miestelio gyventojas užsiprenumeravo leidinių už mažiausią pinigų sumą;
6. turi būti funkcija, kuri rezultatų faile spausdina, už kokią pinigų sumą kiekvienas žmogus užsiprenumeravo leidinių.

Duomenys	Rezultatai
5 4	Tomas 8.00
2.00 3.00 2.00 2.00	Romas 5.00
Tomas 2 0 2 0	Rimas 18.00
Romas 0 1 1 0	Simas 7.00
Rimas 2 2 1 3	Tadas 7.00
Simas 0 1 0 2	45.00
Tadas 1 1 1 0	Rimas 8
	Romas 5.00

19. Struktūrų (įrašų) masyvų elementų rikiavimas

1. Populiariausios programavimo kalbos. TIOBE Software svetainėje (<http://www.tiobe.com>) skelbiami duomenys apie populiariausias programavimo kalbas: pavadinimas (skiriama 20 pozicijų), vieta 2014 metų vasario mėnesį, vieta 2013 metų vasario mėnesį, reitingas procentais, reitingo pokytis, procentais.

Parenkite programą, kuri surikiuotų kalbas reitingo pokyčio didėjimo tvarka.

Pradinių duomenų faile *Duomenys.txt* įrašyta:

- pirmoje eilutėje – programavimo kalbų skaičius n ;
- tolesnėse n eilučių – informacija apie kiekvieną programavimo kalbą: pavadinimas, vieta 2014 metų vasario mėnesį, vieta 2013 metų vasario mėnesį, reitingas ir reitingo pokytis.

Rezultatų faile *Rezultatai.txt* įrašyti kalbas, surikiuotas reitingo pokyčio didėjimo tvarka. Jei kalbų reitingo pokytis yra vienodas, jas palikti išdėstyti taip, kaip pradiniame sąrašė.

Reikalavimai programai:

1. naudokite įrašus (struktūras) ir masyvus su įrašo (struktūros) tipo elementais;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++);
4. sukurkite funkciją, rikiuojančią programavimo kalbas populiarumo pokyčio didėjimo tvarka.

<i>Duomenys.txt</i>	<i>Rezultatai.txt</i>
6	C++ 4 4 6.892 -1.87
C 1 2 18.334 1.25	Java 2 1 17.316 -1.07
Java 2 1 17.316 -1.07	PHP 6 6 4.219 -0.85
Objective-C 3 3 11.341 1.54	C# 5 5 6.450 -0.23
C++ 4 4 6.892 -1.87	C 1 2 18.334 1.25
C# 5 5 6.450 -0.23	Objective-C 3 3 11.341 1.54
PHP 6 6 4.219 -0.85	

2. Įdomiausi filmai. „Oskaro“ apdovanojimas teikiamas už nuopelnus kine. Nuo 1986 m. apdovanojimo ceremonija yra transliuojama per televiziją. Internete Adomas surado informaciją apie „Oskaro“ nominacijas pelniusius geriausius filmus ir nusprendė peržiūrėti apdovanojimų ceremonijų transliacijų įrašus. Sukurkite programą, kuri filmus surikiuotų pagal tokias taisykles: apdovanojimų ceremonijų transliacijos laiko didėjimo tvarka, o jei kelių filmų apdovanojimų ceremonijos transliacijos laikai sutampa, tuomet pagal žiūrovų skaičių mažėjimo tvarka.

Pradinių duomenų faile *Duomenys.txt*:

- pirmoje eilutėje įrašytas apdovanojimų ceremonijų skaičius n;
- tolesnėse n eilučių įrašyta informacija apie vieną apdovanojimų ceremoniją: metai, filmo pavadinimas (gali sudaryti keli žodžiai, skiriamos 25 pozicijos), transliacijos laikas valandomis ir minutėmis (du sveikieji skaičiai), žiūrovų skaičius milijonais (sveikasis skaičius);

Rezultatų faile *Rezultatai.txt* turi būti įrašytas laikantis užduoties reikalavimų surikiuotų filmų sąrašas. Vienam filmui skiriama viena eilutė.

Reikalavimai programai:

1. naudokite struktūras (įrašus) ir jų masyvus;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite rikiavimo pagal du raktus procedūrą (funkciją void C++);
4. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).

<i>Duomenys.txt</i>	<i>Rezultatai.txt</i>
5	2001 Gladiatorius 3 23 43
1988 Paskutinis imperatorius 3 33 42	1992 Avinėlių tylėjimas 3 33 44
1991 Šokis su vilkais 3 35 43	1988 Paskutinis imperatorius 3 33 42
1992 Avinėlių tylėjimas 3 33 44	2006 Avarija 3 33 39
2001 Gladiatorius 3 23 43	1991 Šokis su vilkais 3 35 43
2006 Avarija 3 33 39	

3. Skaniausi saldainiai. Saulius yra saldainių ekspertas. Jis ragauja įvairių gamintojų produkciją ir įvertina saldainių skonį 5 balų skalėje (1 – neskanus; 2 – iš bėdos galima valgyti; 3 – galima valgyti, bet yra ir skanesnių; 4 – skanus; 5 – labai skanus).

Parenkite programą, kuri Sauliui pateiktų kiekvieno gamintojo saldainių sąrašą skanumo didėjimo tvarka.

Pradinių duomenų faile *Duomenys.txt* įrašyta:

- pirmoje eilutėje – saldainių gamintojų skaičius n ;
- toliau surašyti duomenys apie kiekvieną gamintoją:
 - vienoje eilutėje įrašytas gamintojo pavadinimas (skiriama 25 pozicijos) ir saldainių rūšių skaičius r ;
 - tolesnėse r eilučių įrašytas saldainių rūšies pavadinimas (skiriama po 20 pozicijų) ir sveikasis skaičius nuo 1 iki 5, nurodantis saldainio skanumą.

Rezultatų faile *Rezultatai.txt* turi būti įrašyta:

- kiekvieno gamintojo pavadinimas;
- to gamintojo saldainiai, išrikiuoti skanumo didėjimo tvarka. Jei yra kelios to paties skanumo saldainių rūšys, tai jos gali būti išdėstytos bet kuria tvarka.

Reikalavimai programai:

1. naudokite įrašus (struktūras) ir masyvus su įrašo (struktūros) tipo elementais;
2. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
3. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++);
4. sukurkite funkciją, rikiuojančią saldainius skanumo didėjimo tvarka.

<i>Duomenys.txt</i>	<i>Rezultatai.txt</i>
3	Rūta
Rūta 4	Auksinis raktelis 3
Auksinis raktelis 3	Mersi 3
Mersi 3	Žara 3
Čiurlionis 4	Čiurlionis 4
Žara 3	Pergalė
Pergalė 5	Kregždutė 3
Nomeda 5	Sonata 4
Sostinės 5	Nomeda 5
Vilnius 5	Sostinės 5
Sonata 4	Vilnius 5
Kregždutė 3	Mondelez International
Mondelez International 3	Milka 3
Manija 4	Manija 4
Karūna 5	Karūna 5
Milka 3	

4. Geriausi draugai. Kartą Mikei Pūkuotukui besisvečiuojant pas Triušį kilo beprotiška idėja – nustatyti, kas yra geriausi jo draugai. Vidinis balsas Mikei seniai buvo pasakęs geriausio draugo vardą (tai, žinoma, buvo Knysliukas), tačiau yra dar matematika ir statistika, kurios susuko Mikei ir Triušiu protą. Triušis neblogai skaičiavo ir programavo, todėl sutiko sukurti programą, kuri tai apskaičiuotų. Pirmiausia reikėjo paruošti duomenų failą. Mikė žinojo, kad turi n draugų ($3 \leq n \leq 50$). Taip pat gerai žinojo jų vardus, kiek kartų jis pats mielai dalyvavo jų gimtadieniuose ir kiek kartų kiekvienas iš draugų švenė Mikės gimtadienį. Taip pat Mikė, būdamas savanaudis ir smaližius, puikiai atsiminė, kiek medaus puodynėlių jam padovanojo kiekvienas draugas. Be to Mikė turėjo gėdingai prisiminti, kad ne pas visus draugus į svečius eidavo su dovana. Šie duomenys ir lėmė draugiškumo skaičiavimo ypatybes.

Ar Jūs nenorėtumėt sukurti programą, kuri spręstų šį uždavinį?

Duomenys. Duomenys tekstiniam failui *Duomenys.txt*:

- pirmoje eilutėje įrašytas Mikės draugų skaičius n ;
- kitose n eilučių įrašyta šitokia informacija:
 - dv – draugo vardas (16 simbolių, gali būti tarpų);
 - dg – kiek kartų Mikė šventė draugo gimtadienį;
 - md – kiek kartų Mikė draugui dovanojo dovaną;
 - mg – kiek kartų draugas buvo Mikės gimtadienyje;
 - dd – kiek medaus puodynėlių draugas padovanojo Mikei.

Draugai duomenų failu surašyti pagal abėcėlę.

Skaičiavimai. Pirmiausia draugiškumą vertina pagal formulę $d1 = \max(mg, dd) + \min(dg, md)$. Kuo didesnis šis skaičius – tuo geresnis draugas. Šis dydis rodo draugų dosnumą ir pagarbą (pagal Triušį). Jei pasirodytų, kad kelių draugų apskaičiuotos reikšmės yra vienodos, tuomet skaičiuojamas kitas dydis pagal formulę $d2 = | \max(mg, dd) - \min(dg, md) |$. Kuo šio parametro reikšmė yra mažesnė – tuo draugas yra geresnis. Šis dydis parodo savitarpio supratimą (pagal Mikę). Jei abudu dviejų ar kelių draugų dydžiai $d1$ ir $d2$ yra vienodi, tai draugų spausdinimo tvarka nėra svarbi.

Rezultatai. Rezultatai rašomi į tekstinį failą *Rezultatai.txt*. Draugų sąrašas pradamas geriausiu draugu, o toliau – draugiškumo mažėjimo tvarka. Šalia draugo vardo taip pat reikia spausdinti dydžius $d1$ ir $d2$.

Kiekybiniai apribojimai. Draugų skaičius nėra mažesnis už 3 ir neviršija 50. Mikės amžius gali būti tik vienženklis arba dviženklis skaičius. Spausdinant rezultatus vardams skiriama 16 pozicijų, dydžiui $d1$ skiriama 4, o dydžiui $d2$ – 3 pozicijos.

Reikalavimai programai:

1. naudokite struktūras (įrašus) ir jų masyvus, sveikuosius skaičius;
2. sudarykite struktūrą (įrašą) ir masyvą, skirtą draugo duomenims saugoti;
3. sukurkite pradinių duomenų skaitymo procedūrą (funkciją void C++);
4. sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++);
5. sukurkite funkcijas/procedūras, skaičiuojančias draugiškumą, rikiuojančias draugų sąrašą.

<i>Duomenys.txt</i>	<i>Rezultatai.txt</i>
7	Geriausi draugai:
Asiliukas 8 8 5 6	Triušis 15 3
Knysliukas 7 7 7 5	Knysliukas 14 0
Lokis Tokis 3 2 3 2	Asiliukas 14 2
Pelėda 6 6 7 7	Pelėda 13 1
Šarka 2 2 2 1	Šernas 5 1
Šernas 3 2 3 2	Lokis Tokis 5 1
Triušis 7 6 7 9	Šarka 4 0

20. Struktūrų (įrašų) masyvų elementų šalinimas

1. Prekyba buitine technika. Duotas buitinės technikos parduotuvės sandėlio prekių sąrašas *Sandelis.txt* faile. Pirmoje eilutėje pateiktas bendras prekių kiekis, kitose – prekės pavadinimas (25 simboliai), prekės kodas, vienetų skaičius ir vieneto kaina. Kitame faile *Uzsakymai.txt* pateiktas klientų užsakytų prekių sąrašas. Pirmoje eilutėje pateiktas prekių sąrašas, kitose – prekės kodas ir užsakytas vienetų skaičius. Parašykite programą, kuri pateiktą informaciją apie sandėlyje likusias prekes *SandelisRez.txt* rezultatų faile ir apie prekes, kurių sandėlyje trūksta *UzsakymaiRez.txt* faile (jei prekės kiekis yra neigiamas, tuomet prekę reikia užsakyti iš tiekėjo). Rezultatus pateikti tokia forma, kaip nurodyta pavyzdyje.

Reikalavimai programai:

1. Duomenims saugoti naudokite duomenų struktūras.
2. Sukurkite pradinių duomenų skaitymo funkciją.
3. Sukurkite funkciją, kuri šalintų įrašus.
4. Sukurkite sumos skaičiavimo funkciją.
5. Sukurkite rezultatų įrašymo funkciją.

Duomenų failas <i>Sandelis.txt</i>	Duomenų failas <i>Uzsakymai.txt</i>
12	7
Whirlpool šaldytuvas 418 10 1199.00	701 20
Beko šaldytuvas 431 18 1599.99	603 15
Lorenz el. krosnelė 021 26 499.00	124 15
Cook aerogrilis 023 11 359.00	418 10
Aurum el.virdulys 718 54 129.99	615 19
Petra virdulys juodas 701 19 159.00	122 14
Gastroback sulčiaspaudė 219 11 219.00	021 14
Philips sulčiaspaudė 225 12 199.99	
First kavamalė 603 36 99.99	
DeLonghi kavamalė 615 22 109.00	
Bosch džiov. Mašina 124 12 1899.00	
Panasonic džiov. mašina 122 12 1559.00	

Rezultatų failas SandelisRez.txt		Rezultatų failas UžsakymaiRez.txt	
Beko šaldytuvas	431 18 1599.99	Petra virdulys juodas	701 1
Lorenz el. krosnelė	021 12 499.00	Bosch džiov. Mašina	124 3
Cook aerogrilis	023 11 359.00	Panasonic džiov. mašina	122 2
Aurum el.virdulys	718 54 129.99	Sumokėti tiekėjams	8974.00
Gastroback sulčiaspaudė	219 11 219.00		
Philips sulčiaspaudė	225 12 199.99		
First kavamalė	603 21 99.99		
DeLonghi kavamalė	615 3 109.00		

2. Registracija. Registracijos interneto svetainėje priimamos tik patikrinus asmens kredito kortelės numerį. Jei asmuo pateikia neteisingą informaciją, tai jo registracija yra naikinama. Kredito kortelės numeruojamos vieningai – numerį sudaro 13-16 skaitmenų, iš kurių du ar trys pirmi nusako kortelės tipą, 6 skaičiai nusako kortelę išdavusį banką ar įstaigą (BIN), o likę skaičiai parenkami pagal tam tikras taisykles – Luhn'o formulę. Luhn'o formulė – algoritmas, kuris leidžia patikrinti, ar mokėjimo kortelės identifikacinis numeris yra teisingas: ar vedant kortelės numerį nepadaryta klaida, ar numeris nėra atsitiktinai sugalvotas.

Luhn'o formulė: kas antras skaitmuo (antras, ketvirtas ir t.t.) dauginamas iš 2, jei rezultatas dviženklis, abu skaitmenys sudedami. Gauta nauja skaitmenų eilutė sudedama ir, jeigu sumos paskutinis skaitmuo yra 0, numeris yra teisingas.

Pavyzdžiui, identifikacinis kortelės numeris 49927398716 yra teisingas, nes atlikus algoritmą, skaitmenų suma yra lygi 70.

Numeris	4	9	9	2	7	3	9	8	7	1	6
Kas antras skaitmuo dauginamas iš 2		9*2 18		2*2 4		3*2 6		8*2 16		1*2 2	
Nauji numerio skaitmenys	4	9	9	4	7	6	9	7	7	2	6

$$4 + 9 + 9 + 4 + 7 + 6 + 9 + 7 + 7 + 2 + 6 = 70$$

Duomenų faile *reg.txt* pateikti n ($1 < n < 100$) asmenų registracijos duomenys: asmens vardas ir pavardė (25 simboliai) bei kortelės identifikacinis numeris (11 skaitmenų).

Sudarykite programą, kuri naikintų neteisingai nurodžiusių kortelės numerius asmenų registraciją ir šiuos asmenis įrašytų į pašalintų asmenų sąrašą.

Rezultatų failo *rez.txt* pirmoje eilutėje spausdinkite „Geras sąrašas“ ir teisingus duomenys pateikusių asmenų sąrašą. Jeigu tokių asmenų neliko, tai parašykite teksto eilutę „Duomenų nėra“. Po to, spausdinkite „Pašalintų sąrašas“ ir pašalintų asmenų sąrašą. Jeigu tokių asmenų nėra, tai parašykite teksto eilutę „Duomenų nėra“.

<i>reg.txt</i>	<i>rez.txt</i>
5 Vilkas Žiaurusis 49927398716 Lapė Gudrutė 10923247234 Kiškis Naivuolis 12525447558 Meškis Lepeškis 32587413698 Barsukas Sukas 12123212123	Geras sąrašas ----- Vilkas Žiaurusis 49927398716 Kiškis Naivuolis 12525447558 Barsukas Sukas 12123212123 Pašalintų sąrašas ----- Meškis Lepeškis 32587413698 Lapė Gudrutė 10923247234
2 Lapė Gudrutė 10923247234 Meškis Lepeškis 32587413698	Geras sąrašas ----- Duomenų nėra Pašalintų sąrašas ----- Meškis Lepeškis 32587413698 Lapė Gudrutė 10923247234

Reikalavimai programai

1. Naudoti struktūrų (įrašų) masyvus.
2. Sudaryti duomenų skaitymo iš failo funkciją (procedūrą).
3. Sudaryti kortelės numerio teisingumo nustatymo funkciją.
4. Sudaryti struktūros (įrašo) šalinimo iš struktūrų (įrašų) masyvo funkciją (procedūrą).
5. Sudaryti rezultatų rašymo į failą funkciją (procedūrą).

3. Muzikos albumų parduotuvė. Parduotuvė prekiauja muzikos įrašų albumais. Kiekvienas albumas nusakomas pavadinimu (16 simbolių), autoriumi (24 simboliai), kaina (sveikasis skaičius) ir kiekiu (sveikasis skaičius).

Duomenys. Duomenys yra pateikti tekstiniame faile *duom.txt*. Pirmojoje jo eilutėje yra skirtingų albumų skaičius n ($1 \leq n \leq 100$), o kitose n eilučių išvardinti kiekvieno albumo duomenys.

Kitame tekstiniame faile *operacijos.txt* surašyta informacija apie albumų pirkimą ir jų tiekimą.

Šio failo pirmojoje eilutėje yra operacijų skaičius k . Kitose k eilučių pateikta informacija apie albumų tiekimą užrašyta tokia pačia tvarka, kaip ir faile *duom.txt*, tik pradžioje yra pliuso ženklas ir tarpo ženklas. Plusas reiškia, kad albumai buvo gauti. Jei parduotuvėje tokių albumų būta, tai tik padidinamas tokių albumų skaičius, jai nebuvo – albumas įrašomas į albumų sąrašą pabaigoje.

Pardavimas yra fiksuojamas minuso ženklu, po kurio yra tarpas, o toliau nurodomas albumo pavadinimas ir pirkėjų pageidautas albumų skaičius.

Skaičiavimai. Suprantama, kad pirkėjų norai gali būti didesni už parduotuvės galimybes, todėl parduodama tiek, kiek yra tuo momentu. Jei tokių albumų dar kažkiek lieka, tai atitinkamai mažinamas jų skaičius sąrašė. Jei albumų nelieka – toks albumas pašalinamas iš sąrašo.

To paties pavadinimo ir to paties autoriaus albumai gali būti skirtingų kainų. Tuomet jie fiksuojami sąrašė atskirose eilutėse.

Skirtingų autorių albumų pavadinimai skiriasi.

Pirkėjas suinteresuotas jį dominantį albumą nusipirkti už kuo mažesnę kainą, jei tik yra

galimybė. Klientas visada teišus. Pardavėjai maloniai aptarnauja.

Rezultatai. Reikia apskaičiuoti, kokiai sumai parduota albumų ir atspausdinti likusių albumų sąrašą, pateikiant informaciją apie albumus tokiu pačiu pavidalu, kaip duomenų faile *duom.txt*. Rezultatus rašome į tekstinį failą *ataskaita.txt*.

Kiekybiniai apribojimai. Albumų skaičius programos vykdymo eigoje neviršija 100.

Reikalavimai programai:

1. Naudokite struktūras ir struktūrų masyvus, sveikuosius skaičius.
2. Sudarykite struktūrą, skirtą saugoti albumo informacijai.
3. Sukurkite procedūrą (funkciją void C++) skirtą skaityti duomenis iš failo *duom.txt*.
4. Sukurkite procedūrą/funkciją (funkciją C++) skirtą apdoroti likusiems duomenims iš failo *operacijos.txt*;
5. Sukurkite rezultatų rašymo į failą procedūrą (funkciją void C++).
6. Sukurkite funkciją, kuri rastų pigiausią albumą sąrašė.
7. Sukurkite funkciją, kuri pašalintų albumą iš sąrašo.

Duomenys

<i>Duomenys.txt</i>			
3			
Pabėgimas	A. Mamontovas	25	4
Tadas Blinda	A. Mamontovas	22	3
Šiam pasauly	G. Paškevičius	18	5

<i>operacijos.txt</i>			
10			
+ Tranzas	G. Paškevičius	26	5
- Tadas Blinda	1		
+ Beribiam danguje	G. Paškevičius	17	5
- Tadas Blinda	3		
- Tranzas	4		
+ Tadas Blinda	A. Mamontovas	22	2
+ Tadas Blinda	A. Mamontovas	18	2
- Tadas Blinda	1		
- Šiam pasauly	2		
- Tadas Blinda	10		

Rezultatai

<i>ataskaita.txt</i>			
Parduota už 286 Lt			
Prekių likučiai:			
Pabėgimas	A. Mamontovas	25	4
Šiam pasauly	G. Paškevičius	18	3
Tranzas	G. Paškevičius	26	1
Beribiam danguje	G. Paškevičius	17	5

Reikalavimai programai:

1. Sąrašui saugoti naudokite struktūrų masyvus.
2. Klasės mokinių sąrašą formuokite tokiu būdu: sąrašas sutvarkomas masyvo elementų įterpimo ir šalinimo būdu prioritetų skaitmenų didėjimo tvarka (1,1,2,2,2,3,...). Nustatytą mokinių skaičių viršijantys mokiniai yra pašalinami iš sąrašo.
3. Sukurti funkcijas, sutvarkančias, papildančias ar šalinančias mokinių sąrašo masyvo elementus. Galima kurti papildomas funkcijas.
4. Sukurti duomenų skaitymo iš failo ir rezultatų rašymo į failą funkcijas.

Duomenys	Rezultatai
10 13 Petras_Petraitis 2 Jonas_Jonaitis 1 Andrijauskaitė_Agnė 1 Balsytė_Giedrė 3 Bružas_Darius 2 Eičinaitė_Ema 2 Gulbinas_Kęstas 1 Kumponas_Ervidas 1 Lapšytė_Vilija 3 Pranauskis_Pranas 3 Pranauskytė_Meda 2 Vasaris_Liudas 2 Žiulpauskas_Ažuolas 1	Sudarytos klasės mokinių sąrašas: Žiulpauskas_Ažuolas 1 Kumponas_Ervidas 1 Gulbinas_Kęstas 1 Andrijauskaitė_Agnė 1 Jonas_Jonaitis 1 Vasaris_Liudas 2 Pranauskytė_Meda 2 Eičinaitė_Ema 2 Bružas_Darius 2 Petras_Petraitis 2

3. Krepšinis. Prieš krepšinio varžybas komandos vadovas registruoja 12 žaidėjų. Krepšinio varžybų metu sekretorius fiksuoja kas (žaidėjo numeris) ir kiek pelno taškų (taškų skaičius). Jeigu žaidėjas metė, tačiau nepataikė – rašomas 0.

Parašykite programą, kuri lentelėje pateiktą tokią varžybų statistiką: žaidėjo numerį, jo vardą, pavardę, įmetų baudų skaičių (po 1 tašką), dvitaškių skaičių (2), tritaškių skaičių (3), iš viso žaidėjo surinktų taškų skaičių ir bendrą metimų tikslumą procentais (0 reiškia, jog žaidėjas metė, tačiau nepataikė). Taip pat nustatyti: pelniusį daugiausia taškų, įmetusį daugiausia tritaškių ir taikliausią žaidėją.

Faile *zaidėjai.txt* pateiktas 12-kos registruotų žaidėjų sąrašas: atskirose eilutėse įrašomas žaidėjo numeris, jo vardas ir pavardė (25 simboliai).

Duomenų faile *taskai.txt* saugoma tokia informacija: žaidėjo numeris (nuo 1 iki 100) ir pelnytų taškų skaičius (0, 1, 2 arba 3), tam skiriama viena teksto eilutė. Faile gali būti ne daugiau kaip 100-as įrašų.

Rezultatų faile *suvestine.txt* pateikiama lentelė:

Nr.	Vardas Pavardė	Baudų metimų skaičius	Dvitaškių metimų skaičius	Tritaškių metimų skaičius	Surinkti taškai	Nepataikytų metimų skaičius	Tikslumas
-----	-------------------	-----------------------------	---------------------------------	---------------------------------	--------------------	-----------------------------------	-----------

Suvestinėje žaidėjai išrikiuoti nuo didžiausią numerį turinčio žaidėjo iki mažiausią numerį turinčio žaidėjo. Po lentele pateikite geriausius žaidėjus: numerį, vardą, pavardę ir geriausią rodyklį. Jeigu yra keli geriausi žaidėjai, spausdinti vieną iš jų.

Reikalavimai programai:

1. duomenims saugoti sukurti įrašo tipo masyvą(-us);
2. sukurti vieną arba dvi duomenų skaitymo iš dviejų failų funkcijas (procedūras);
3. sukurti funkciją, kuri skaičiuoja surinktus taškus;
4. sukurti funkciją, kuri skaičiuoja metimų tikslumą;
5. sukurti funkciją (procedūrą) rikiuoti įrašų masyvą pagal žaidėjo numerius;
6. sukurti geriausių žaidėjų atrankos funkciją (procedūrą);
7. sukurti rezultatų rašymo į failą funkciją (procedūrą).

<i>zaidejai.txt</i>	<i>taskai.txt</i>
12 Aldevinas Siriūnaitis	12 0
13 Darius Gricius	12 1
2 Donatas Drimeikis	12 1
4 Giedrius Purauskas	11 3
10 Henrikas Paulauskas	9 2
11 Justas Ališauskas	16 2
9 Justinas Vitkus	33 2
23 Mindaugas Lukonaitis	9 0
3 Robertinas Vilimas	4 0
26 Tomas Lukonaitis	4 1
88 Tomas Samalionis	11 2
33 Valdas Dobilas	16 3
	10 2
	9 0
	11 3
	9 2
	23 3
	33 3
	10 1
	11 0
	9 3
	33 0
	23 2
	23 0
	9 2
	9 1
	9 1

Rezultatų failas *suvestine.txt*

```
*****
|Nr.| Vardas Pavardė          |Baudos | Dvit. | Trit. | Taškai |Nepat.|Tikslumas|
*****
| 2 | Donatas Drimeikis       | 0      | 0     | 0     | 0      | 0     | 0      |
| 3 | Robertinas Vilimas     | 0      | 0     | 0     | 0      | 0     | 0      |
| 4 | Giedrius Purauskas    | 1      | 0     | 0     | 1      | 1     | 50     |
| 9 | Justinas Vitkus        | 2      | 3     | 1     | 11     | 2     | 75     |
|10 | Henrikas Paulauskas   | 1      | 1     | 0     | 3      | 0     | 100    |
|11 | Justas Ališauskas     | 0      | 1     | 2     | 8      | 1     | 75     |
|12 | Aldevinas Siriūnaitis | 2      | 0     | 0     | 2      | 1     | 67     |
|13 | Darius Gricius        | 0      | 0     | 0     | 0      | 0     | 0      |
|23 | Mindaugas Lukonaitis  | 0      | 1     | 1     | 5      | 1     | 67     |
|26 | Tomas Lukonaitis     | 0      | 0     | 0     | 0      | 0     | 0      |
|33 | Valdas Dobilas       | 0      | 1     | 1     | 5      | 1     | 67     |
|88 | Tomas Samalionis     | 0      | 0     | 0     | 0      | 0     | 0      |
*****
```

```
          Daugiausia taškų pelnė: 9          Justinas Vitkus          11
Daugiausia įmetė tritaškių: 11         Justas Ališauskas        2
          Taikliausias: 10              Henrikas Paulauskas     100
```

4. Orientacinio sporto varžybos. Miške paslepiami kontroliniai punktai, kuriuos sportininkai privalo surasti. Kiekviename punkte paliktas varžybų blankas ir rašymo priemonė. Suradęs punktą sportininkas tame blanke užrašo punkto radimo laiką ir savo numerį, o savo blanke – tą patį laiką ir surasto punkto numerį. Be to, teisėjai fiksuoja sportininko numerį, jo starto laiką ir finišo laiką. Starto ir finišo laiką užsirašo ir sportininkas savo blanke. Punktus sportininkas lanko bet kokia tvarka, tik paskutinis punktas turi būti finišas. Startas neįeina į punktų skaičių, jame tik fiksuojamas starto laikas. Sportininkai gauna kompasus ir žemėlapius, kuriuose pažymėti visi punktai, taip pat ir startas. Jeigu sportininkas neranda finišo punkto, tai jis turi grįžti į startą, o vietoje finišo laiko įrašomas grįžimo į startą laikas. Tokiu atveju jo rezultatai nėra skaičiuojami. Punktai numeruojami iš eilės pradedant nuo vieneto. Punktas su didžiausiu numeriu yra finišas. Punktuose esantys varžybų blankai yra pateikiami teisėjų kolegijai. Varžybų nugalėtojai nustatomi pirmiausia atsižvelgiant į rastų punktų skaičių. Didesnis rastų punktų skaičius garantuoja aukštesnę vietą. Jei keli sportininkai rado vienodą punktų skaičių, tai tuomet tikrinama, kuris iš jų trasoje užtruko trumpiau. Nėra dviejų sportininkų, kurie turėtų du vienodus pagrindinius kriterijus – vienodą punktų skaičių ir bėgimo trasoje trukmę. Trasą įveikia bent vienas sportininkas. Jeigu sportininkas nesutinka su teisėjų kolegijos sprendimu, jis gali pateikti apeliaciją su savo blanku ir palyginti savo ir teisėjų duomenis.

Duomenys ir kiekybiniai apribojimai. Tekstiniame faile *orient-duomenys.txt* duota informacija, kurią turi teisėjų kolegija.

Pirmoje failo eilutėje duotas sportininkų skaičius n ($2 \leq n \leq 100$). Sportininkai numeruojami skaičiais iš eilės nuo 1 iki n pagal surašymo tvarką duomenų faile. Tolesnėse n eilučių yra duoti sportininkų vardai ir pavardės (keli žodžiai, iš viso iki 16 simbolių) starto laikas ir finišo laikas išreikšti valandomis minutėmis ir sekundėmis. Bet kurio sportininko laikas trasoje neviršija 8 valandų.

Toliau duotas punktų skaičius p ($2 \leq p \leq 20$). Kitose p eilučių surašyti tokie duomenys:

Punkto numeris, punktą radusių sportininkų skaičius ir sportininkų numeriai, atskirti tarpais.

Rezultatai. Rezultatai spausdinami tekstiniame faile *orient-rezultatai.txt*. Reikia apskaičiuoti, kuriam sportininkui skiriama pirmoji vieta, atspausdinti jo vardą ir pavardę. Toliau spausdinti visą trasą įveikusią sportininkų sąrašą rezultato blogėjimo tvarka: vardas, pavardė surastų punktų

skaičius ir sugaištas laikas trasoje. Nepasiekę finišo sportininkai nespausdinami.

Rezultatai spausdinami tokiu pavidalu:

Vardas ir pavardė – kaip duota tekstiniame faile, po to 1 tarpas.

Surastų punktų skaičiui skiriamos 2 pozicijos, po to 1 tarpas.

Laikas tiek duomenų, tiek rezultatų faile užrašytas naudojant dvitaškį.

Valandų skaičiui skiriamos dvi pozicijos.

Minutės ir sekundės spausdinamos dvizenkliais skaičiais. Jei skaičius vienaženklis – priekyje pridedamas nulis.

Reikalavimai programai:

1. Aprašykite struktūrą, kuri tiktų saugoti vieno sportininko duomenis. Gali būti naudojamos ir kitos struktūros, pavyzdžiui, saugoti laikui.
2. Programoje panaudokite struktūrų masyvą.
3. Parašykite skaitymo funkciją (procedūrą). Skaitymo funkcija gali atlikti tam tikrą skaitomų duomenų apdorojimą.
4. Parašykite kitas duomenų apdorojimo funkcijas (procedūras).
5. Parašykite funkciją (procedūrą), kuri spausdina rezultatus.

Duomenys <i>orient-duomenys.txt</i>	Paaškinimai
4 Jonas Gugis 8:45:00 10:11:25 Petras Kukis 8:50:00 10:06:11 Balys Tomkus 8:55:00 11:34:56 Algis Jonkus 9:00:00 10:08:55	Sportininkų skaičius Vardas, pavardė starto ir finišo laikas
5 1 3 1 3 4 3 4 2 1 3 4 2 3 1 3 4 5 3 1 3 4 4 4 1 2 3 4	Punktų skaičius Pirmą punktą aplankė 3 sportininkai, kurių numeriai 1, 3 ir 4 Finišas yra 5 punktas Petras Kukis finišo nepasiekė

Rezultatai <i>orient-rezultatai.txt</i>
Algis Jonkus I vieta
Algis Jonkus 5 1:08:55
Jonas Gugis 5 1:26:25
Balys Tomkus 5 2:39:56