

## 8 savaitė

### Minimalios ir maksimalios reikšmės paieška masyve

#### 1 lygis

#### 1. Atostogų kelionės (iš knygos „Informacinių technologijų brandos egzaminui. Užduočių pavyzdžiai)

Šeima, kurią sudaro  $s$  suaugusiųjų ir  $v$  vaikų, keliauja iš Varšuvos į Budapeštą traukiniu be persėdimų. Žinomos bilietų kainos vykstant kiekvienu to maršruto traukiniu. Be to, perkant daugiau nei  $k$  bilietų, visų bilietų kainoms taikoma  $p$  procentų nuolaida. Parenkite programą, kuri apskaičiuotų, kiek šeimai kainuos kelionė kiekvienu nurodyto maršruto traukiniu ir kuriuo traukiniu keliauti apsimoka labiausiai.

Pradinių duomenų failo **atostogos\_data.txt**:

- pirmoje eilutėje nurodytas keliaujančių suaugusiųjų  $s$  ir vaikų  $v$  skaičius;
- antroje eilutėje yra bilietų, kuriems taikoma nuolaida, skaičius  $k$ ;
- trečioje eilutėje nurodytas traukinių, važiuojančių maršrutu Varšuva-Budapeštas, skaičius  $n$ ;
- tolesnėse  $n$  eilučių įrašyti duomenys, keliaujant kiekvienu nurodyto maršruto traukiniu: po du realiuosius skaičius – bilieto kaina suaugusiajam bei vaikui, ir sveikasis skaičius – nuolaida  $p$ .

Pirmosiose  $n$  rezultatų failo **atostogos\_res.txt** eilučių pateikite traukinio eilės numerį ir šeimos kelionės šiuo traukiniu išlaidas. Paskutinėje eilutėje nurodykite, kuriuo traukiniu šeimai keliauti labiausiai apsimoka. Jei yra keli tokie traukiniai, tai įrašykite paskutinio iš eilės tinkamo traukinio numerį.

#### Reikalavimai

- Naudokite sveikųjų ir realiųjų skaičių masyvus.
- Sukurkite pradinių duomenų skaitymo funkciją void.
- Sukurkite rezultatų rašymo į failą funkciją void.
- Sukurkite funkciją, skaičiuojančią, kiek kainuos šeimai kelionė vienu traukiniu.
- Sukurkite traukinio, kuriuo šeimai keliauti pigiausia, paieškos funkciją.

atostogos_data.txt	atostogos_res.txt
3 4	1 685.44
3	2 740.35
4	3 670.21
147 80.15 10	4 696.80
163 95.50 15	3
147 80.15 12	
163 95.50 20	

## 2. Pirkinys (iš knygos „Informacinių technologijų valstybiniam brandos egzaminui: užduočių pavyzdžiai“)

Šeima ruošiasi pirkti automobilį. Tėvas išrinko n automobilių ir surašė jų duomenis: pagaminimo metus, vietų skaičių, pardavimo kainą ir automobilio priežiūros išlaidas. Parenkite programą, kuri iš sąrašo išrinktų šeimai tinkamus automobilius.

Pirmoje pradinių duomenų failo **pirkinys\_data.txt** eilutėje įrašytas tėvo pasirinktų automobilių skaičius n. Tolesnėse n eilučių surašyti duomenys apie automobilius: pagaminimo metai, vietų skaičius, pardavimo kaina ir automobilio priežiūros išlaidos.

Rezultatų failo **pirkinys\_res.txt** atskirose eilutėse surašykite duomenis šių automobilių:

- naujausio;
- turinčio daugiausiai vietų;
- kainuojančio mažiausiai;
- kurio priežiūra pigiausia.

Jei yra keli tinkami automobiliai, tai reikia pateikti jų visų duomenis.

### Reikalavimai

- Naudokite sveikųjų ir realiųjų skaičių masyvus.
- Sukurkite pradinių duomenų skaitymo funkciją void.
- Sukurkite didžiausios reikšmės paieškos funkciją ir ją panaudokite naujausiam ir daugiausiai vietų turinčiam automobiliui rasti.
- Sukurkite mažiausios reikšmės paieškos funkciją ir ją panaudokite ieškodami automobilio, kuris būtų pigiausias ir kurio priežiūra kainuotų mažiausiai.
- Sukurkite rezultatų rašymo į failą funkciją void.

pirkinys_data.txt	pirkinys_res.txt
4	2010 5 24000.00 7520.12
2008 5 22000.00 5400.55	2009 7 30385.00 3254.56
2010 5 24000.00 7520.12	2008 6 21900.00 4512.50
2009 7 30385.00 3254.56	2009 7 30385.00 3254.56
2008 6 21900.00 4512.50	

## 3. Sportininkai (iš knygos „Informacinių technologijų valstybiniam brandos egzaminui: užduočių pavyzdžiai“)

Tarptautinis Olimpinis Komitetas (TOK) sportininkams rekomenduoja maisto sudėtinių dalių pasiskirstymą per parą: angliavandeniai turėtų sudaryti **a** procentų, riebalai – **r** procentų, baltymai – **b** procentų gaunamų kilokalorijų (kcal). Žinomi sportininkų duomenys: ūgis, masė ir suvartojamas energijos kiekis kcal per parą.

Parašykite programą, kuri rastų aukščiausio, žemiausio, lengviausio ir sunkiausio sportininko masę, ūgį, suvartojamos energijos kiekį kcal per parą ir kūno masės indeksą (KMI). Sportininko kūno masės indeksas (KMI), apskaičiuojamas pagal Adolphe Quetelet formulę:

$$KMI = \text{masė (kg)} / \text{ūgis}^2 (\text{cm}^2).$$

Jeigu yra keli vienodai aukščiausi (žemiausi, sunkiausi, lengviausi) sportininkai, tai spausdinami visų duomenys.

Pradinių duomenų faile **sportininkai\_data.txt**:

- pirmoje eilutėje įrašyti trys sveikieji skaičiai **a**, **r**, **b** – rekomenduojamas angliavandenių, baltymų ir riebalų kiekis procentais;
- antroje failo eilutėje įrašytas krepšinio komandos sportininkų skaičius **n**;
- tolesnėse **n** eilučių įrašyta po du realiuosius skaičius: sportininko kūno masė (kg), ūgis (m) ir sveikasis skaičius – per parą suvartojamas energijos kiekis (kcal).

Rezultatų faile **sportininkai\_res.txt** turi būti įrašyta:

- pirmosiose **n** eilučių – kiekvieno sportininko eilės numeris ir kūno masės indeksas (KMI), apvalinamas iki sveikąjį skaičių;
- tolesnėse eilutėse turi būti spausdinama:
  - aukščiausio sportininko duomenys: eilės numeris, masė, ūgis, suvartojamas energijos kiekis, KMI. Jei yra keli tokie sportininkai, tuomet turi būti spausdinami visų sportininkų duomenys;
  - žemiausio sportininko duomenys: eilės numeris, masė, ūgis, suvartojamas energijos kiekis, KMI. Jei yra keli tokie sportininkai, tuomet turi būti spausdinami visų sportininkų duomenys;
  - lengviausio sportininko duomenys: eilės numeris, masė, ūgis, suvartojamas energijos kiekis, KMI. Jei yra keli tokie sportininkai, tuomet turi būti spausdinami visų sportininkų duomenys;
  - sunkiausio sportininko duomenys: eilės numeris, masė, ūgis, suvartojamas energijos kiekis, KMI. Jei yra keli tokie sportininkai, tuomet turi būti spausdinami visų sportininkų duomenys.

## Reikalavimai programai

- naudokite sveikųjų ir realiųjų skaičių masyvus;
- sukurkite pradinių duomenų skaitymo funkciją `void`;
- sukurkite didžiausios reikšmės paieškos funkciją. Ją panaudokite ieškodami aukščiausio ir sunkiausio sportininkų;
- sukurkite mažiausios reikšmės paieškos funkciją. Ją panaudokite ieškodami žemiausio ir lengviausio sportininkų;
- sukurkite rezultatų rašymo į failą funkciją `void C++`.

<b>sportininkai_data.txt</b>	<b>sportininkai_res.txt</b>
60 25 15	1 24
12	2 25
82.00 1.85 3608	3 24
98.00 1.98 4073	4 22
92.00 1.95 3893	5 29
80.00 1.90 3541	6 26
125.00 2.08 4853	7 25
107.00 2.04 4358	8 24
107.00 2.05 4367	9 26
97.00 2.03 4094	10 23
117.00 2.11 4677	11 24
98.00 2.05 4138	12 24

93.00 1.98 3946	9 117.00 2.11 4677 26
85.00 1.88 3650	1 82.00 1.85 3608 24
	4 80.00 1.90 3541 22
	5 125.00 2.08 4853 29

## 2 lygis

### 4. Giliukai

Klasės mokinių  $n$  ( $1 \leq n \leq 30$ ) dalyvavo zoologijos sodo organizuotoje akcijoje: rinko giliukus. Kiekvieno mokinio atneštų giliukų svoris buvo įrašytas į sąrašą. Svorių sąrašas kilogramais sudarytas pagal mokinių klasės abėcėlinį sąrašą (t.y pirmo mokinio giliukų svoris parašytas pirmas, antro – antras ir t.t.). Jei mokinys neatnešė giliukų – rašomas 0. Klasės vadovui prireikė parengti ataskaitą apie daugiausiai ir mažiausiai giliukų surinkusius mokinius ir kiek kilogramų giliukų vidutiniškai surinko vienas klasės mokinių (tie mokiniai, kurie nedalyvavo akcijoje, atliekant skaičiavimus nevertinami). Jei keli mokiniai atnešė po vienodai daugiausiai ir vienodai mažiausiai giliukų, turi būti spausdinami jų visų numeriai. Pradinių duomenų faile **giliukai\_data.txt** pirmoje eilutėje nurodomas klasės mokinių skaičius  $n$ . Antroje eilutėje  $n$  realiųjų skaičių – atneštų giliukų svoriai kilogramais.

Rezultatų failo **giliukai\_res.txt** pirmoje eilutėje įrašykite didžiausią giliukų svorį trijų skaitmenų po kablelio tikslumu ir išvardinkite juos atnešusių mokinių numerius atskirdami juos vieną nuo kito tarpais.

Antroje eilutėje įrašykite mažiausią giliukų svorį trijų skaitmenų po kablelio tikslumu ir išvardinkite juos atnešusių mokinių numerius atskirdami juos vieną nuo kito tarpais.

Trečioje eilutėje įrašykite vidutinį giliukų svorį trijų skaitmenų po kablelio tikslumu.

<b>giliukai_data.txt</b>	<b>giliukai_res.txt</b>
10	10.000 7
2.56 1.598 0 5.12 0 3.012 10 4.789 2.569 0	1.598 2
	2.965

### Reikalavimai

- Duomenims saugoti naudokite masyvus.
- Sukurkite pradinių duomenų skaitymo funkciją `void()`.
- Sukurkite funkcijas `void()`, randančią didžiausio arba mažiausio svorio dydį ir mokinio numerį.
- Sukurkite funkciją vidurkio skaičiavimui, gražinančią reikšmę per funkcijos vardą.
- Sukurkite rezultatų rašymo į failą funkciją `void()`.

## 5. Žaidimas „Protų mūšis“

„Ažuolo“ gimnazijoje buvo organizuojamas žaidimas „Protų mūšis“. Žaidimui užsiregistravo visų gimnazijos klasių komandos, kurios varžėsi tarpusavyje palaikomos mokytojų ir komandų sirgalių. Intelektualaus žaidimo metu moksleiviai turėjo logiškai išmąstyti atsakymus į įvairiausių klausimus, kuriuos jiems paruošė įvairių sričių specialistai. Žaidimo metu buvo užduodami klausimai iš technologinių ir fizinių mokslų, humanitarinių ir socialinių mokslų bei menų srities. Žaidime buvo keturi turai. Tuomet buvo susumuoti per visus turus surinkti taškai ir paskelbti žaidimo nugalėtojai.

Žinomas komandų skaičius  $n$  ( $n < 30$ ), kiekvienos komandos numeris ir turuose kiekvienos komandos surinkti taškai. Parašykite programą, kuri rastų kiekvieno turo komandas, kurios surinko daugiausia taškų, komandas, kurios surinko mažiausiai taškų ir žaidimo nugalėtojus.

Duomenų failo **protmuisis\_data.txt** pirmoje eilutėje įrašytas žaidime dalyvavusių komandų skaičius  $n$ . Tolesnėse  $n$  eilučių įrašyti duomenys apie kiekvieną komandą – komandos numeris, triženklis sveikasis skaičius ir kiekviename ture surinktų taškų suma.

Rezultatų faile **protmuisis\_res.txt** įrašykite kiekvieno turo komandas, surinkusias daugiausiai ir mažiausiai taškų, ir komandą, kuri laimėjo žaidimą. Jei yra kelios laimėtojos, turi būti išvardintos visos tokia tvarka, kokia išdėstytos pradiniam sąrašė.

<b>protmuisis_data.txt</b>	<b>protmuisis_res.txt</b>
16	Daugiausiai:
101 10 8 5 24	I ture 203
102 10 12 10 16	II ture 102
103 8 6 6 12	III ture 102 303 403 404
104 4 6 5 3 0	IV ture 403
201 4 4 2 0 202 5 5 3 20	Maziausiai:
203 12 2 5 10	I ture 204
204 2 5 8 10	II ture 203
301 8 7 9 20	III ture 201
302 7 5 9 10	IV ture 201
303 9 10 10 22	Laimejo: 403
304 8 9 7 10	
401 10 8 9 15	
402 10 8 8 25	
403 10 10 10 26	
404 7 9 10 22	

### Reikalavimai

- Naudokite sveikųjų skaičių masyvus.
- Sukurkite pradinių duomenų skaitymo funkciją.
- Sukurkite funkciją vienos komandos surinktų taškų sumai rasti.
- Sukurkite funkcijas daugiausiai ir mažiausiai taškų surinkusioms komandoms rasti.
- Sukurkite rezultatų rašymo į failą funkciją.

## 6. Skaitomiausios knygos

Leidyklos „Alma Littera“  $n$  ( $n < 21$ ) knygų vertina  $m$  ( $m < 11$ ) skaitytojų. Kiekvienas skaitytojas kiekvieną knygą vertina balais nuo 0 iki 10. Vertinimai yra sveikieji skaičiai.

Pradinių duomenų failo **knygos\_data.txt** pirmoje eilutėje įrašyti skaičiai  $n$  ir  $m$ . Tolesnėse  $n$  eilučių įrašytas knygos kodas  $k$  (sveikasis skaičius iš intervalo  $[1; 20]$ ) ir po  $m$  sveikųjų skaičių, vienas nuo kito atskirtų tarpais – kiekvieno vertintojo knygai skirti balai. Skaičiuojant galutinį knygos įvertinimą, sumuojami didžiausias ir mažiausias įvertinimai (jei didžiausių ir mažiausių įvertinimų yra daugiau, negu po vieną, tuomet sumuojama tik po vieną didžiausią ir mažiausią įvertinimą). Apskaičiuota suma yra galutinis knygos įvertinimas.

Parašykite programą, surandančią ir rezultatų faile **knygos\_res.txt** spausdinančią:

- Koks yra galutinis kiekvienos knygos įvertinimas. Rezultatus surašykite faile vienai knygai skirdami vieną eilutę: knygos kodas ir įvertinimas, atskirti tarpais. Sąrašas išdėstytas knygų kodų didėjimo tvarka
- Kuri knyga įvertinta blogiausiai. Jei kelios knygos įvertintos blogiausiai, jų visų kodai turi būti spausdinami vienoje eilutėje didėjimo tvarka, atskiriant juos vieną nuo kito tarpais.
- Kuri knyga įvertinta geriausiai. Jei kelios knygos įvertintos geriausiai, jų visų numeriai turi būti spausdinami vienoje eilutėje mažėjimo tvarka.

<b>knygos_data.txt</b>	<b>knygos_res.txt</b>
5 6	Knygu ivertinimai:
2 2 5 6 6 5 2	1 11
1 3 8 8 7 7 7	2 8
4 4 5 5 7 7 6	3 8
3 5 5 5 5 3 3	4 11
5 7 8 8 7 2 6	5 10
	Blogiausiai ivertintos knygos: 2 3
	Geriausiai ivertintos knygos: 4 1

### Reikalavimai programai

- Naudokite sveikųjų skaičių masyvus.
- Sukurkite pradinių duomenų skaitymo funkciją `void`.
- Sukurkite mažiausios ir didžiausios reikšmių paieškos funkcijas, grąžinančias apskaičiuotą reikšmę per funkcijos vardą.
- Sukurkite `void` funkciją, faile spausdinančią blogiausiai įvertintas knygas.
- Sukurkite `void` funkciją, faile spausdinančią geriausiai įvertintas knygas.
- Sukurkite `void` funkciją, faile spausdinančią knygų kodus ir įvertinimą.

### 3 lygis

#### 7. Akcijos (XV olimpiada, 2004 m.)

Kiekvienos dienos ryte nustatoma kiek tą dieną kainuos viena banko akcija. Kaina nekinta iki kitos dienos ryto.

Jei žinotume, kokios bus tikslios akcijų kainos tam tikrą laiko periodą, optimaliai pirkdami ir parduodami akcijas galėtume gerai užsidirbti.

Duota turima gryną pinigų suma, laiko periodas (t. y. dienų skaičius) ir vienos akcijos kaina kiekvieną periodo dieną. Parašykite programą, kuri apskaičiuotų, kokią didžiausią gryną pinigų sumą galime turėti pasibaigus periodui, jei optimaliai pirsime ir parduosime akcijas.

Akcijos perkamos ar parduodamos už NUSTATYTAŲ DIENOS KAINĄ. Laikykite, kad akcijų, kurias galima pirkti, skaičius neribotas, t. y. akcijas galite pirkti už visus turimus pinigus (suprantama, galima pirkti ir parduoti tik sveiką akcijų skaičių).

Pradiniai duomenys pateikti faile **akcijos\_data.txt**. Pirmoje failo eilutėje įrašyti du sveikieji skaičiai: P ( $100 \leq P \leq 1000$ ) ir N ( $2 \leq N \leq 4000$ ). P – tai turima pinigų suma eurais, N – dienų skaičius.

Toliau faile yra N eilučių, kuriose surašytos akcijų kainos kiekvieną periodo dieną, po vieną sveikąjį skaičių eilutėje. Antroje eilutėje įrašyta vienos akcijos kaina pirmąją periodo dieną, trečioje eilutėje – vienos akcijos kaina antrąją dieną ir t. t. Vienos akcijos kaina gali svyruoti nuo 1 iki 100 eurų.

Rezultatą – vieną sveikąjį skaičių – išspausdinkite faile **akcijos\_res.txt**. Tai didžiausia gryną pinigų suma, kurią galite turėti pasibaigus periodui.

<b>akcijos_data.txt</b>	<b>akcijos_res.txt</b>
210 6 70 90 80 50 60 70	370
210 3 90 80 50	210

#### 8. Sandėliai (XVIII olimpiada, 2006-2007 m.m.)

Bendrovė „Olimpiados sprendimai“ turi sandėlio registrą, kuriame registruojama atvežtų prekių siuntų kilmės šalis, atvežimo data ir kiekis. Į sandėlį, kuris pradiniu momentu buvo tuščias, prekės buvo atvežamos siuntomis. Visų prekių siuntų atvežimo data skirtinga, be to, visos atvežamos prekės yra vienos rūšies, nors gali būti iš skirtingų šalių.

Pirmiau visuomet realizuojama anksčiau atvežta siunta.

Atvežus visas prekes ir dalį jų realizavus paaiškėjo, kad sandėliuke tebeguli N prekių. Pagal ES reikalavimus, įmonė turi teikti ataskaitas, kiek kokios šalies prekių buvo realizuota. Kadangi žinome, kad anksčiau atvežta siunta anksčiau ir realizuojama, tokią informaciją galima nesunkiai nustatyti.

Žinomi sandėlio registro duomenys, likusių prekių skaičius ir dominančios šalies kodas. Parašykite programą, kuri suskaičiuotų, kiek prekių iš dominančios šalies jau realizuota.

Pradiniai duomenys pateikti faile **sandeliai\_data.txt**. Pirmoje eilutėje yra du tarpais atskirti sveikieji skaičiai N ( $0 \leq N \leq 10\,000\,000$ ) ir K ( $1 \leq K \leq 1000$ ). Skaičius N nusako likusių prekių skaičių, skaičius K – registro eilučių (t.y. siuntų, nes vienai siuntai skiriama viena eilutė) skaičių.

Antroje eilutėje nurodytas ieškomos šalies kodas, kurį sudaro dvi didžiosios lotyniškos raidės. Likusiose K eilučių įrašyti duomenys apie kiekvieną siuntą.

Kiekvienos eilutės pradžioje nurodytas šalies kodas (dvi didžiosios lotyniškos raidės), toliau siuntos atvežimo data (dešimt simbolių) bei prekių kiekis toje siuntoje  $P_i$  (sveikasis skaičius;  $0 < P_i \leq 10\,000$ ). Šalies kodas, siuntos atvežimo data ir prekių kiekis toje siuntoje vienas nuo kito atskirti lygiai vienu tarpo simboliu.

Rezultatą – realizuotų nurodytos šalies prekių skaičių – reikia įrašyti į failą **sandeliai\_res.txt**. Jeigu iš nurodytos šalies nebuvo realizuota nei viena prekė, reikia įrašyti 0.

<b>sandeliai_data.txt</b>	<b>sandeliai_res.txt</b>
8 6 LV LV 2005-02-01 5 LT 1995-05-04 14 LT 2005-02-03 10 LT 2006-03-11 5 LV 2005-11-23 2 ES 2006-09-01 2	6
10000 3 LV LT 2006-01-01 1 LT 2006-01-02 1 LV 2006-01-03 10000	0