

**2008 m. INFORMACINIŲ TECHNOLOGIJŲ VALSTYBINIO BRANDOS
EGZAMINO UŽDUOTIES**

VERTINIMO INSTRUKCIJA

Pagrindinė sesija

Testas

Klausimo numeris	Maks. taškų skaičius	Atsakymai ir komentarai																																																			
1	1	1. Pažymėti visą surinktą tekstą ir Formatavimo komandų grupėje pasirinkti komandą Didžiosios/Mažosios raidės (<i>Change Case...</i>) pasirinkti Visos didžiosios (<i>Uppercase</i>). 2. Pažymėti surinktą tekstą ir Formatavimo komandų grupės šriftų kortelėje pasirinkti Visos didžiosios (<i>All caps</i>). 3. Galima sukurti naują stilių su reikiamu formatavimu arba pasirinkti iš esamų stilių sąrašo ir suteikti jį surinktam tekstui. Už vieną panašų atsakymą 1 taškas.																																																			
2	1	C																																																			
3	2	Atstumai tarp raidžių pasikeis 2 fragmente, nes raidės išretintos tarp jų įterpian tarpą. 1 taškas skiriamas už teisingai nurodytą teksto fragmentą. 1 taškas skiriamas už teisingą priežasties paaiškinimą.																																																			
4	3	2 – maketu, maketavimu. 3 – stiliumi. 4 – šablonu, dokumento ruošiniu, trafaretu, forma, ruošiniu. Po 1 tašką už kiekvieną teisingą atsakymą.																																																			
5	1	B																																																			
6	1	Skaičiuoklė tokį rezultatą parodė dėl to, kad langelio F2 skaičių (<i>Number</i>) formato skaitmenų kiekis trupmeninėje rezultato dalyje (<i>Decimal places</i>) yra nulis arba langeliui F2 suteiktas sveikojo skaičiaus formatas. Už panašų atsakymą 1 taškas .																																																			
7	5	<table><tr><th>Langelis</th><th>D6</th><th>E6</th><th>F6</th><th>G6</th><th>H6</th></tr><tr><td>Rezultatas</td><td>20</td><td>98</td><td>TRUE</td><td>FALSE</td><td>NE</td></tr></table> Po 1 tašką už kiekvieną teisingą įrašą.	Langelis	D6	E6	F6	G6	H6	Rezultatas	20	98	TRUE	FALSE	NE																																							
Langelis	D6	E6	F6	G6	H6																																																
Rezultatas	20	98	TRUE	FALSE	NE																																																
8	4	Galimi atsakymų variantai: <table><tr><td></td><td>A</td><td>B</td><td></td><td>A</td><td>B</td></tr><tr><td>1</td><td>Upė</td><td>Plotas</td><td>1</td><td>(2) Upė</td><td>(1) Plotas</td></tr><tr><td>2</td><td>Volga</td><td></td><td>2</td><td>(3) Volga</td><td></td></tr><tr><td>3</td><td></td><td></td><td>3</td><td></td><td></td></tr><tr><td>4</td><td></td><td>231000</td><td>4</td><td></td><td>(4) 23100</td></tr></table> <table><tr><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr><tr><td>1</td><td>Upė</td><td>Volga</td><td></td><td></td><td></td><td></td></tr><tr><td>2</td><td>Plotas</td><td></td><td></td><td>231000</td><td></td><td></td></tr></table> Po 1 tašką už kiekvieną teisingą įrašą.		A	B		A	B	1	Upė	Plotas	1	(2) Upė	(1) Plotas	2	Volga		2	(3) Volga		3			3			4		231000	4		(4) 23100		A	B	C	D	E	F	1	Upė	Volga					2	Plotas			231000		
	A	B		A	B																																																
1	Upė	Plotas	1	(2) Upė	(1) Plotas																																																
2	Volga		2	(3) Volga																																																	
3			3																																																		
4		231000	4		(4) 23100																																																
	A	B	C	D	E	F																																															
1	Upė	Volga																																																			
2	Plotas			231000																																																	
9	3	1 – naršyklėmis. 2 – saitą, nuorodą, hipernuorodą, hipersaitą. 3 – svetaine, portalas. Po 1 tašką už kiekvieną teisingą atsakymą.																																																			

Klausimo numeris	Maks. taškų skaičius	Atsakymai ir komentarai																																																																								
10	2	1. Atviroji programa – laisvai platinama programa, kurios <u>pirminis tekstas, dažniausiai užrašytas programavimo kalba, laisvai platinamas</u> . Ją galima modifikuoti laikantis prie jos pridėtų arba joje nurodytų licencijos sąlygų. 2. Nemokama programa – programa, kuria galima nemokamai naudotis laikantis jos autoriaus nustatytų taisyklių (licencijos sąlygų). Po 1 tašką už kiekvieną panašų teisingą atsakymą. Jei teigiama, kad atviroji programa neturi autorinių teisių ar pan. – vertinama 0 taškų .																																																																								
11	1	B																																																																								
12	1	C																																																																								
13	2																																																																									
<table><tr><td></td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td></tr><tr><td>1 eilutė</td><td></td><td></td><td></td><td></td><td></td><td></td><td>A</td><td>R</td><td>B</td><td>A</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>2 eilutė</td><td>V</td><td>I</td><td>E</td><td>N</td><td>A</td><td>S</td><td></td><td>O</td><td>R</td><td></td><td>V</td><td>I</td><td>E</td><td>N</td><td>A</td><td>S</td><td>=</td><td>V</td><td>I</td><td>E</td><td>N</td><td>A</td><td>S</td></tr></table>				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	1 eilutė							A	R	B	A														2 eilutė	V	I	E	N	A	S		O	R		V	I	E	N	A	S	=	V	I	E	N	A	S
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23																																																			
1 eilutė							A	R	B	A																																																																
2 eilutė	V	I	E	N	A	S		O	R		V	I	E	N	A	S	=	V	I	E	N	A	S																																																			
		Po 1 tašką už kiekvieną teisingą eilutę.																																																																								
14	1	D																																																																								
15	2	<table><tr><td>11</td></tr><tr><td>12</td></tr><tr><td>13</td></tr><tr><td>14</td></tr><tr><td>21</td></tr></table> <p>1 taškas už teisingai atspausdintas bent 3 iš eilės einančias reikšmes. 1 taškas už teisingą n reikšmę paskutiniame langelyje.</p>	11	12	13	14	21																																																																			
11																																																																										
12																																																																										
13																																																																										
14																																																																										
21																																																																										
16	3	Vienas iš galimų sprendimo būdų: <pre>Write(' ': (4-i + 1)); for j := 1 to i do Write ('**'); WriteLn;</pre> <p>Galimi ir kiti sprendimo būdai. 1 taškas už tinkamą <u>tuščių pozicijų</u> vietą ekrane. 1 taškas už tinkamą * vietą, 1 taškas už tinkamą * skaičių. 0 taškų už *, jei * yra daugiau nei prašoma užduotyje.</p>																																																																								
17	2	17.1. Atlikus šią programą ekrane bus spausdinama TRUE. 17.2. Funkcija Tikrinimas realizuoja C algoritimą. Po 1 tašką už kiekvieną teisingą atsakymą (net tuo atveju, kai 17.1 ir 17.2 sukeisti vietomis).																																																																								
18	4	Galimi įvairūs teisingi sprendimo būdai. Pavyzdžiui toks: <pre>procedure Pirma (a, b, c, d : integer; var pl, pr : integer); begin</pre>																																																																								

Klausimo numeris	Maks. taškų skaičius	Atsakymai ir komentarai																																																																																																																								
		<pre>pl := (c - a) * (b - d); pr := 2 * ((c - a) + (b - d)); end;</pre> <p>Kreipinys į procedūrą: Pirma (x1, y1, x2, y2, s, p);</p> <p>1 taškas už teisingą procedūros antraštę, 1 taškas už teisingą ploto skaičiavimą, 1 taškas už teisingą perimetro skaičiavimą, 1 taškas už tinkamą kreipinį į procedūrą. Jei yra sintaksės klaidų (pvz.: vietoj „*“ - taškas, vietoj „:=“ – „=“) vertinimas mažinamas 1 tašku už kiekvieną skirtingą klaidą.</p>																																																																																																																								
19	3	<table><tr><td></td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td></tr><tr><td>1 eilutė</td><td>P</td><td>S</td><td>K</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>2 eilutė</td><td>L</td><td>A</td><td>B</td><td>A</td><td>I</td><td></td><td>G</td><td>R</td><td>A</td><td>Ž</td><td>I</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>3 eilutė</td><td>P</td><td>A</td><td>S</td><td>A</td><td>K</td><td>A</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> <p>Po 1 tašką už kiekvieną teisingą eilutę.</p>		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	1 eilutė	P	S	K																			2 eilutė	L	A	B	A	I		G	R	A	Ž	I											3 eilutė	P	A	S	A	K	A																																															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21																																																																																																					
1 eilutė	P	S	K																																																																																																																							
2 eilutė	L	A	B	A	I		G	R	A	Ž	I																																																																																																															
3 eilutė	P	A	S	A	K	A																																																																																																																				
20	3	<p>21.1. Procedūra Darbas šalina iš masyvo elementą.</p> <p>21.2. Programa T21 pašalina iš masyvo pasikartojančius elementus.</p> <p>21.3. Tikslių skaičių pozicijų nereikalaujama.</p> <table><tr><td></td><td>9</td><td></td><td>7</td><td></td><td>1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> <p>Po 1 tašką už kiekvieną teisingą atsakymą.</p>		9		7		1																																																																																																																		
	9		7		1																																																																																																																					
21	5	<table><tr><td></td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>1</td></tr><tr><td>1 eilutė</td><td></td><td></td><td>1</td><td></td><td></td><td>1</td><td>*</td><td>*</td><td>*</td><td>N</td><td>E</td><td>T</td><td>I</td><td>N</td><td>K</td><td>A</td><td>*</td><td></td><td></td></tr><tr><td>2 eilutė</td><td></td><td></td><td>2</td><td></td><td></td><td>2</td><td>*</td><td>*</td><td>*</td><td>T</td><td>I</td><td>N</td><td>K</td><td>A</td><td>*</td><td></td><td></td><td></td><td></td></tr><tr><td>3 eilutė</td><td></td><td></td><td>3</td><td></td><td></td><td>1</td><td>*</td><td>*</td><td>*</td><td>N</td><td>E</td><td>T</td><td>I</td><td>N</td><td>K</td><td>A</td><td>*</td><td></td><td></td></tr><tr><td>4 eilutė</td><td></td><td></td><td>8</td><td></td><td></td><td>1</td><td>*</td><td>*</td><td>*</td><td>T</td><td>I</td><td>N</td><td>K</td><td>A</td><td>*</td><td></td><td></td><td></td><td></td></tr><tr><td>5 eilutė</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> <p>Po 1 tašką už teisingas reikšmes kiekviename stulpelyje, 1 taškas – už teisingą rašymą pozicijose, 1 taškas už tinkamą žvaigždučių kiekį.</p>		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	1	1 eilutė			1			1	*	*	*	N	E	T	I	N	K	A	*			2 eilutė			2			2	*	*	*	T	I	N	K	A	*					3 eilutė			3			1	*	*	*	N	E	T	I	N	K	A	*			4 eilutė			8			1	*	*	*	T	I	N	K	A	*					5 eilutė																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	1																																																																																																							
1 eilutė			1			1	*	*	*	N	E	T	I	N	K	A	*																																																																																																									
2 eilutė			2			2	*	*	*	T	I	N	K	A	*																																																																																																											
3 eilutė			3			1	*	*	*	N	E	T	I	N	K	A	*																																																																																																									
4 eilutė			8			1	*	*	*	T	I	N	K	A	*																																																																																																											
5 eilutė																																																																																																																										

Praktinės užduotys

1. Tyrimai

Maksimalus vertinimas – 27 taškai

Ukmergėje yra m ($1 \leq m \leq 100$) autobusų maršrutų. Norėdami išsiaiškinti, kiek autobusų reikia kiekvienam maršrutui, mokiniai skaičiavo, kiek pervežama keleivių. Stebėtoju vienoje stotelėje dirbo tik vienas mokinys. Mokiniai stebėjo (dirbo) visose miesto stotelėse, išskyrus paskutinę kiekvieno maršruto stotelę, kurioje išlipa visi dar važiuojantys keleiviai. Buvo registruojami visų maršrutų keleiviai. Savo stebėjimo rezultatus kiekvienas mokinys rašė duomenų lape: maršruto numerį ir kiek įlipo keleivių (teigiamas skaičius) arba/ir maršruto numerį ir kiek išlipo keleivių (neigiamas skaičius). Atvejai, kai stotelėje nebuvo įlipusių ir/arba išlipusių keleivių, duomenų lape nebuvo registruojami. (Žiūrėkite Aidos duomenų lapo pavyzdį).

Stebėtoja Aida

Maršrutas	Įlipo/išlipo
6	-5
6	12
1	4
6	1
5	2

Duomenys. Visi stebėjimų duomenys surašyti tekstiniame faile `U1.txt`. Pirmoje eilutėje yra visų mokinių surinktų duomenų skaičius n ($n \geq 1$). Toliau eilučių yra po du skaičius: maršruto numeris ir keleivių skaičius (teigiamas, jeigu įlipo, neigiamas – jeigu išlipo).

Parašykite programą, kuri spausdintų faile `U1rez.txt` mokinių stebėjimų rezultatus.

1. Pirmoje eilutėje spausdintų maršrutų, kuriais važiuo bent vienas keleivis, numerius didėjimo tvarka.
2. Antroje eilutėje spausdintų, kiek keleivių vežta kiekvienu maršrutu maršrutų numerių didėjimo tvarka.
3. Trečioje eilutėje spausdintų kiek kiekvieno maršruto autobusų keleivių išlipo visose tarpinėse stotelėse arba nulį, jeigu nė vienas keleivis iš to maršruto autobusų neišlipo. Spausdintų maršrutų numerių didėjimo tvarka.
4. Ketvirtoje eilutėje spausdintų maršruto, kurio visais autobusais vežta daugiausia keleivių, numerį. Jeigu tokie maršrutai yra keli, tai spausdintų mažiausią numerį.

Kiekvienam skaičiui spausdinti skirtos 6 pozicijos.

Pastaba. Vežtų maršrutu keleivių skaičius yra lygus įlipusių į visus to maršruto autobusus keleivių skaičių sumai.

Nurodymai:

- Rašydami programą naudokite tik vienmačius sveikųjų skaičių masyvus.
- Programoje neturi būti sakinių, skirtų darbui su ekranu.
- Rezultatų faile turi būti keturios eilutės. Jeigu ne visi skaičiavimai atlikti, tuomet atitinkamoje eilutėje spausdinkite žodį NE.
- Parašykite procedūrą duomenims iš failo skaityti.
- Parašykite procedūrą tik maršrutų numeriams spausdinti didėjimo tvarka.
- Parašykite funkciją maršruto, kuriuo vežta daugiausia keleivių, numeriui rasti.

Pavyzdžiai

U1.txt	Paiškinimai	U1rez.txt					
12	Stebėjimų skaičius	3	4	6	8	9	12
6 -1	Maršrutas 6, išlipo 1	27	9	5	45	9	16
6 5	Maršrutas 6, įlipo 5	-10	-5	-3	0	0	-16
3 15	Maršrutas 3, įlipo 15	8					
4 9	Maršrutas 4, įlipo 9						
6 -2	Maršrutas 6, išlipo 2						
12 16	Maršrutas 12, įlipo 16						
8 45	Maršrutas 8, įlipo 45						
4 -5	Maršrutas 4, išlipo 5						
3 12	Maršrutas 3, įlipo 12						
3 -10	Maršrutas 3, išlipo 10						
12 -16	Maršrutas 12, išlipo 16						
9 9	Maršrutas 9, įlipo 9						
	Rezultatų pavyzdys, kai ne visi skaičiavimai atlikti	3	4	6	8	9	12
		27	9	5	45	9	16
		-10	-5	-3	0	0	-16
		NE					

Programos vertinimas

Vertinimo kriterijai	Taškai	Pastabos
Testai.	17	Taškai skiriami, jeigu programa pateikia teisingus visų testų rezultatus.
Teisinga duomenų skaitymo iš failo procedūra.	3	Vertinama tada, kai neskiriama taškų už testus.
Teisinga procedūra maršrutų numeriams spausdinti didėjimo tvarka.	4	
Teisinga funkcija maršruto, kuriuo vežta daugiausia keleivių, numeriui rasti.	3	
Teisingos kitos procedūros ir funkcijos, jeigu jų yra, ir pagrindinė programa.	6	
Rezultatų faile yra visos keturios eilutės (su skaičiais arba/ir žodžiu NE).	1	
Vienmačio masyvo duomenų tipas (tipai) ir kintamųjų aprašymas.	1	Visada vertinama.
Yra nurodytos procedūros ir funkcijos, jose nėra jokių kitų veiksmų, tik nurodytieji.	6	
Prasmingi kintamųjų vardai.	1	
Komentarai, rašybos taisyklės.	1	
Programos rašymo stilius, nėra sakinių darbui su ekranu.	1	
Iš viso	27	

Programos vertinimo detalizavimas

Vertinimo kriterijai	Taškai	Pastabos
Testai.	17	Taškai skiriami, jeigu programa pateikia teisingus visų testų rezultatus.
Teisinga duomenų skaitymo iš failo procedūra: <ul style="list-style-type: none"> failo atidarymas ir uždarymas, pirmos eilutės skaitymas; ciklo organizavimas; duomenų skaitymo sakiniai ciklo viduje. 	③ 1 1 1	Vertinama tada, kai neskiriama taškų už testus.
Teisinga procedūra maršrutų numeriams spausdinti didėjimo tvarka: <ul style="list-style-type: none"> ciklo organizavimas; spausdinimo sakinyje cikle; maršrutų numerių pateikimas didėjimo tvarka (tai gali būti jau skaitymo procedūroje pagal indeksą sudėti numeriai, arba organizuojama paieška numerių didėjimo tvarka, arba rikiavimas); Jeigu ne visos stotelės, tai taškas neskiriamas failo pateikimas (failo kintamasis pateikiamas per parametrus, failas atidaromas papildymui, failas nurodomas globaliu kintamuoju). 	④ 1 1 1 1	
Teisinga funkcija maršruto, kuriuo vežta daugiausia keleivių, numeriui rasti: <ul style="list-style-type: none"> rezultato kintamajam suteikiama pradinė reikšmė, rezultatas priskiriamas funkcijos vardui; paieškos ciklo organizavimas; sąlygos sakinyje cikle. 	③ 1 1 1	
Teisingos kitos procedūros ir funkcijos, jeigu jų yra, ir pagrindinė programa: <ul style="list-style-type: none"> pagalbiniai kintamieji lokalūs; teisingi kreipiniai į nurodytas funkcijas ir procedūras; kiti pagrindinės programos sakiniai, kitos procedūros ir funkcijos, jeigu yra. 	⑥ 1 3 2	
Rezultatų faile yra visos keturios eilutės (su skaičiais arba/ir žodžiu NE).	1	
Vienmačio masyvo duomenų tipas (tipai) ir kintamųjų aprašymas.	1	Visada vertinama.
Yra nurodytos procedūros ir funkcijos, jose nėra jokių kitų veiksmų, tik nurodytieji. Po vieną tašką už antraštę ir po vieną tašką už tai, kad procedūra bei funkcija tiksliai atitinka užduoties nurodymus.	6	
Prasmingi kintamųjų vardai.	1	
Komentarai, rašybos taisyklės.	1	
Programos rašymo stilius, nėra sakinių darbui su ekranu.	1	
Iš viso	27	

// Programos pavyzdys testams kurti, vertinimo sistemai patikrinti.

// **Bandomieji testai.**

1	3									
3 15	15									
	0									
	3									
10	1	2	3	4	10					
3 15	39	16	27	9	45					
4 9	-2	0	-10	-5	-45					
1 -2	10									
2 16										
10 45										
4 -5										
1 39										
10 -45										
3 12										
3 -10										
5	1	2	3							
3 15	2	16	15							
3 -15	-2	0	-15							
1 -2	2									
2 16										
1 2										
12	3	4	6	8	9	12				
3 15	27	9	5	45	9	16				
4 9	-10	-5	-3	0	0	-16				
6 -2	8									
12 16										
8 45										
4 -5										
6 -1										
6 5										
3 12										
3 -10										
12 -16										
9 9										
51	3	4	6	8	9	12	15	26	33	34
33 15	93	36	20	180	36	64	15	0	30	9
4 9	-40	-15	-9	0	0	-64	0	-3	0	-5
6 -2	8									
12 16										
8 45										
4 -5										
26 -1										
6 5										
3 12										
3 -10										
12 -16										
9 9										
3 15										
4 9										
6 -2										
12 16										
8 45										
34 -5										
6 -1										
6 5										
3 12										
3 -10										
12 -16										
9 9										
33 15										
34 9										
26 -2										
12 16										
8 45										
4 -5										
6 -1										
6 5										
3 12										
3 -10										
12 -16										
9 9										
3 15										
4 9										
6 -2										
12 16										
8 45										
4 -5										

6 -1	
6 5	
3 12	
3 -10	
12 -16	
9 9	
3 15	
4 9	
15 15	

// Programos pavyzdys testams kurti, vertinimo sistemai patikrinti.

```
program P1;
const Cfd = 'U111.txt';
      Cfr = 'U111rez.txt';
      Cn = 100;
type Mas = array[1..Cn] of integer;
var  A,           // Maršrutai - įlipo
     B : Mas;     // Maršrutai - išlipo
//-----
procedure Duomenys;
var i, n, m, sk : integer;
    fd : text;
begin
  for i := 1 to Cn do begin A[i] := 0; B[i] := 0; end;
  Assign(fd, Cfd); Reset(fd);
  ReadLn(fd, n);
  for i := 1 to n do begin
    ReadLn(fd, m, sk);
    if sk > 0 then A[m] := A[m] + sk
      else B[m] := B[m] + sk;
  end;
  Close(fd);
end;
//-----
procedure Numeriai(var fr : text);
var i : integer;
begin
  for i := 1 to Cn do
    if (A[i] <> 0) or (B[i] <> 0) then
      Write(fr, i:5, ' ');
  WriteLn(fr);
end;
//-----
procedure Spausdinti(var fr : text; var A, B : mas);
var i : integer;
begin
  for i := 1 to Cn do
    if (A[i] <> 0) or (B[i] <> 0) then Write(fr, A[i]:5, ' ');
  WriteLn(fr);
end;
//-----
function Kuris : integer;
var i, d : integer;
begin
  d := 1;
  for i := 1 to Cn do
    if A[i] > A[d] then d := i;
  Kuris := d;
end;
//-----
var fr : text;
begin
  Duomenys;
  Assign(fr, Cfr); Rewrite(fr);
  Numeriai(fr);
  Spausdinti(fr, A, B);
  Spausdinti(fr, B, A);
  WriteLn(fr, Kuris:5);
  Close(fr);
  WriteLn('Labas ');
  ReadLn;
end.
```


2. Transportas

Maksimalus vertinimas – 23 taškai

Vilniuje yra n stotelių, kuriose sustoja miesto transporto troleibusai. Troleibusų maršrutai numeruojami sveikaisiais skaičiais, tačiau nebūtinai iš eilės. Kiekvienoje stotelėje yra informacinė lentelė, kurioje surašyti maršrutų, kurių troleibusai sustoja toje stotelėje, numeriai nebūtinai didėjimo tvarka. Maršruto ilgis nusakomas stotelių skaičiumi.

Duomenys yra tekstiname faile `U2.txt`. Pirmoje eilutėje yra stotelių skaičius mieste n ($1 \leq n \leq 100$). Toliau kiekvienoje eilutėje yra duomenys apie konkrečią stotelę: pavadinimas (20 simbolių), informacinėje lentelėje esančių maršrutų numerių skaičius ir maršrutų numeriai.

Parašykite programą, kuri į tekstinio failo `U2rez.txt` pirmąją eilutę spausdintų ilgiausio maršruto numerį, jeigu yra keli tokie maršrutai, tuomet iš jų mažiausią numerį. Toliau spausdintų surasto ilgiausio maršruto stotelių pavadinimus po vieną eilutėje tokia tvarka, kokia stotelės buvo išvardytos duomenų faile.

Nurodymai:

- Rašydami programą naudokite masyvus su įrašo tipo elementais; papildomai gali būti naudojami ir kitokio tipo masyvai.
- Programoje neturi būti sakinių, skirtų darbui su ekranu.
- Parašykite procedūrą, skirtą tik duomenims skaityti iš failo.
- Parašykite funkciją ilgiausio maršruto numeriui rasti.
- Parašykite procedūrą, kuri spausdintų į tekstinį failą nurodyto maršruto stotelių pavadinimus.

Pavyzdys

<code>U2.txt</code>	Paiškinimai	<code>U2rez.txt</code>
5 Rytas 5 1 2 5 4 6 Vakarai 3 12 5 4 Baltasis lokys 6 12 1 6 8 7 3 Panerys 1 12 Rudasis tiltas 3 8 14 4	Stotelių skaičius Stotelės pavadinimas, maršrutų skaičius, maršrutų numeriai	4 Rytas Vakarai Rudasis tiltas

Programos vertinimas

Vertinimo kriterijai	Taškai	Pastabos
Testai.	18	Taškai skiriami, jeigu programa pateikia teisingus visų testų rezultatus.
Teisinga duomenų skaitymo iš failo procedūra.	3	Vertinama tada, kai neskiriama taškų už testus.
Teisinga funkcija ilgiausio maršruto numeriui rasti.	3	
Teisinga procedūra spausdinti į tekstinį failą nurodyto maršruto stotelių pavadinimams.	4	
Teisingos kitos procedūros ir funkcijos, jeigu jų yra, ir pagrindinė programa.	8	
Įrašo duomenų tipas, masyvas su įrašo tipo elementais, kintamieji.	1	Visada vertinama.
Yra nurodytos procedūros ir funkcija, jose nėra jokių kitų veiksmų, tik nurodytieji.	3	
Komentariai, rašybos taisyklės.	1	
Programos rašymo stilius, nėra sakinių darbui su ekranu.	1	
Iš viso	23	

Programos vertinimo detalizavimas

Vertinimo kriterijai	Taškai	Pastabos
Testai.	18	Taškai skiriami, jeigu programa pateikia teisingus visų testų rezultatus.
Teisinga duomenų skaitymo iš failo procedūra: <ul style="list-style-type: none"> failo atidarymas ir uždarymas, pirmos eilutės skaitymas; ciklų organizavimas; duomenų skaitymo sakiniai ciklo viduje. 	③ 1 1 1	Vertinama tada, kai neskiriama taškų už testus.
Teisinga funkcija ilgiausio maršruto numeriui rasti: <ul style="list-style-type: none"> rezultato kintamajam suteikiama pradinė reikšmė, rezultatas priskiriamas funkcijos vardui; paieškos ciklo organizavimas; sąlygos sakinyje cikle. 	③ 1 1 1	
Teisinga procedūra spausdinti į tekstinį failą nurodyto maršruto stotelių pavadinimams: <ul style="list-style-type: none"> ciklo organizavimas; ciklo kintamieji lokalūs; spausdinimo sakinyje cikle; failo pateikimas (failo kintamasis perduodamas per parametrus, failas atidaromas papildyti, failas nurodomas globaliu kintamuoju). 	④ 1 1 1 1	
Teisingos procedūros ir funkcijos, jeigu jų yra, ir pagrindinė programa: <ul style="list-style-type: none"> pagalbiniai kintamieji lokalūs; teisingi kreipiniai į nurodytas funkcijas ir procedūras; jeigu nespausdinamas ilgiausio maršruto numeris, tuomet įvertinimas mažinamas vienu tašku; kiti pagrindinės programos sakiniai, kitos procedūros ir funkcijos, jeigu yra (1 tšk. skaitymo procedūros vidinio ciklo organizavimas; po vieną tašką už teisingus skaičiavimų veiksmus procedūrose ir funkcijose). 	⑧ 1 3 4	
Įrašo duomenų tipas, masyvas su įrašo tipo elementais, kintamieji. Vertinama, kai sukurtas įrašo duomenų tipas panaudotas.	1	Visada vertinama.
Yra nurodytos procedūros ir funkcijos, jose nėra jokių kitų veiksmų, tik nurodytieji.	3	
Komentariai, rašybos taisyklės. Programos rašymo stilius, nėra sakinių darbui su ekranu.	1	
Iš viso	23	

// Programos pavyzdys testams kurti, vertinimo sistemai patikrinti.

// Bandomieji testai.

1	1
Rytas 5 1 2 5 4 6	Rytas
7	12
Rytas 5 1 2 5 4 6	Vakarai
Vakarai 3 12 5 4	Ramusis parkas
Ramusis parkas 8 12 5 4 1 2 6 3 10	Baltasis lokys
Baltasis lokys 6 12 1 6 8 7 3	Panerys
Panerys 1 12	Eglynas
Eglynas 2 12 1	
Rudasis tiltas 3 8 14 4	
2	2
Rytas 1 2	Rytas
Vakarai 1 5	
7	5
Rytas 1 5	Rytas
Vakarai 1 5	Vakarai
Ramusis parkas 1 5	Ramusis parkas
Baltasis lokys 1 5	Baltasis lokys
Panerys 1 5	Panerys
Eglynas 1 5	Eglynas
Rudasis tiltas 1 5	Rudasis tiltas
3	5
Rytas 3 5 7 9	Rytas
Vakarai 3 5 7 9	Vakarai
Ramusis parkas 3 5 7 9	Ramusis parkas

Toliau pateikiami trys visiškai skirtingi programų tekstai.

Pirmasis – įrašo viduje nėra masyvo. Formuojamas duomenų masyvas, kuriame surašomi duomenys „stotelė – maršrutas“. Atskirai formuojamas maršrutų stotelių skaičiaus masyvas.

Antrasis – įrašas, skirtas duomenims saugoti. Maršrutams kuriamas atskiras masyvėlis, kuriame surašomi maršrutų ilgiai. Maršruto numeris – indeksas.

Trečiasis – įrašas skirtas maršrutams aprašyti. Kiekvienam maršrutui (indeksas masyve) surašomos stotelės. Tai daroma duomenų skaitymo metu, t.y. duomenys saugomi kitu patogesniu pavidalu tolesniems veiksams. Nereikia specialiai skaičiuoti kiekvieno maršruto ilgio.

Lyg ir yra problemos sprendimo pasirinkimo erdvė.

Pirmoji versija

```
program P2a;
const Cn = 100;    // Stotelių skaičius
      Ck = 60;    // Maršrutų skaičius
type  Mas = array[1..Ck] of integer;
      Stotele = record
                    pav : string[20];
                    A : Mas;           // Maršrutų numeriai
                    n : integer;       // Kiek maršrutų stoja stotelėje
                end;
      VisosSt = array[1..Cn] of Stotele;
var fd : text;
    nM : integer; // Stotelių skaičius mieste
    M : VisosSt;  // Miesto stotelių sąrašas
    B : Mas;      // Maršrutų sąrašas
    nB : integer;
//-----
procedure Duomenys;
var fd : text;
    i, j : integer;
begin
    Assign(fd, 'P2duom.txt'); Reset(fd);
    ReadLn(fd, nM);           // Stotelių skaičius
    for i := 1 to nM do begin
        Read(fd, M[i].pav, M[i].n); // Maršrutų skaičius stotelėje
        for j := 1 to M[i].n do
            Read(fd, M[i].A[j]);    // Maršrutų numeriai
        ReadLn(fd);
    end;
    Close(fd);
end;
//-----
// Masyvo indeksas yra maršruto numeris; Ilgis - stotelių skaičius
procedure MarsrutuIlgiai;
var i, j : integer;
begin
    for i := 1 to Ck do B[i] := 0;
    for i := 1 to nM do
        for j := 1 to M[i].n do
            B[M[i].A[j]] := B[M[i].A[j]] + 1;
    end;
//-----
// Ilgiausio maršruto numeris; iš kelių vienetų - mažiausias numeris
function IlgiausiasM : integer;
var L, i : integer;
begin
    L := 1;
    for i := 1 to Ck do
        if B[i] > B[L] then L := i;
    IlgiausiasM := L;
end;
//-----
// Ilgiausio maršruto stotelių pavadinimai
procedure Spausdinti(var fr : text; nr : integer);
var i, j : integer;
```

```

begin
  for i := 1 to nM do
    for j := 1 to M[i].n do
      if M[i].A[j] = nr then WriteLn(fr, M[i].pav);
    end;
  //-----
var fr : text; ilg : integer;
begin
  Duomenys;
  MarsrutuIlgiai;
  ilg := IlgiausiasM;
  Assign(fr, 'P2rez.txt'); Rewrite(fr);
  WriteLn(fr, ilg);
  Spausdinti(fr, ilg);
  Close(fr);
end.
//-----

```

Antroji versija

```

program P2b;
const Cn = 100;    // Stotelių skaičius
      Ck = 60;    // Maršrutų skaičius
type  zodis = string[20];
      Mas = array[1..Cn] of zodis;
      Marsrutas = record
        A : Mas;           // Stotelių pavadinimai
        n : integer;       // Stotelių skaičius - maršruto ilgis
      end;
      VisiM = array[1..Ck] of Marsrutas;
var fd : text;
    nM : integer; // Maršrutų skaičius mieste
    M : VisiM;    // Miesto maršrutų sąrašas
//-----
procedure Duomenys;
var fd : text;
    i, j, p, nr : integer;  pav : zodis;
begin
  for i := 1 to Ck do M[i].n := 0;
  Assign(fd, 'P2duom.txt'); Reset(fd);
  ReadLn(fd, nM);           // Stotelių skaičius
  for i := 1 to nM do begin
    Read(fd, pav, p);       // Stotelės pavadinimas ir maršrutų kiekis
  stotelėje
    for j := 1 to p do begin
      Read(fd, nr);         // Maršruto numeris
      M[nr].n := M[nr].n + 1;
      M[nr].A[M[nr].n] := pav;
    end;
    ReadLn(fd);
  end;
  Close(fd);
end;
//-----
// Ilgiausio maršruto numeris; iš kelių vienodų - mažiausias numeris
function IlgiausiasM : integer;
var L, i : integer;
begin
  L := 1;
  for i := 1 to Ck do
    if M[i].n > M[L].n then L := i;
  IlgiausiasM := L;
end;

```

```
//-----
// Ilgiausio maršruto stotelių pavadinimai
procedure Spausdinti(var fr : text; nr : integer);
var i : integer;
begin
    for i := 1 to M[nr].n do
        WriteLn(fr, M[nr].A[i]);
    end;
//-----
var fr : text; ilg : integer;
begin
    Duomenys;
    ilg := IlgiausiasM;
    Assign(fr, 'P2brez.txt'); Rewrite(fr);
    WriteLn(fr, ilg);
    Spausdinti(fr, ilg);
    Close(fr);
end.
```

Trečioji versija

```
program P2c;
const Cn = 100;    // Stotelių skaičius
      Ck = 60;    // Maršrutų skaičius
type zodis = string[20];
type Stotele = record
    pav : zodis;    // Stotelė
    mn  : integer;  // Maršruto numeris
end;
Visi = array[1..Cn * Ck] of Stotele;
Mas = array[1..Ck] of integer;
var fd : text;
    nM : integer; // Masyvo M duomenų skaičius
    M  : Visi;    // Stotelė - maršrutas sąrašas
    B  : Mas;     // Maršrutų sąrašas
    nB : integer; // B masyvo ilgis
//-----
procedure Duomenys;
var fd : text;
    i, j, n, k : integer;
    pav : zodis;
begin
    Assign(fd, 'P2duom.txt'); Reset(fd);
    ReadLn(fd, k);           // Duomenų eilučių skaičius
    nM := 0;
    for i := 1 to k do begin
        Read(fd, pav, n);
        for j := 1 to n do begin
            nM := nM + 1;
            M[nM].pav := pav;
            Read(fd, M[nM].mn);
        end;
        ReadLn(fd);
    end;
    Close(fd);
end;
//-----
// Masyvo indeksas yra maršruto numeris; Ilgis - stotelių skaičius
procedure MarsrutuIlgiai;
var i, j : integer;
begin
    for i := 1 to Ck do B[i] := 0;
    for i := 1 to nM do
        B[M[i].mn] := B[M[i].mn] + 1;
    end;
```

```
//-----  
// Ilgiausio maršruto numeris; iš kelių vienojų - mažiausias numeris  
function IlgiausiasM : integer;  
var L, i : integer;  
begin  
    L := 1;  
    for i := 1 to Ck do  
        if B[i] > B[L] then L := i;  
    IlgiausiasM := L;  
end;  
//-----  
// Ilgiausio maršruto stotelių pavadinimai  
procedure Spausdinti(var fr : text; nr : integer);  
var i : integer;  
begin  
    for i := 1 to nM do  
        if M[i].mn = nr then WriteLn(fr, M[i].pav);  
    end;  
//-----  
var fr : text; ilg : integer;  
begin  
    Duomenys;  
    MarsrutuIlgiai;  
    ilg := IlgiausiasM;  
    Assign(fr, 'P2crez.txt'); Rewrite(fr);  
    WriteLn(fr, ilg);  
    Spausdinti(fr, ilg);  
    Close(fr);  
    Readln;  
end.
```