

1. Tétel

Az **atestat.in** állomány két sort tartalmaz. Az első sorában egy nem nulla természetes szám van, n -el jelöljük ($5 < n < 15$). Az állomány második sora n darab egymástól szóközzel elválasztott, legtöbb 4 jegyű egész számot tartalmaz, n elemű egész számsort alkotva.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyő első sorába kiírja azokat a számokat az állományból, amelyek tükörszámok (palindromok). Egy szám akkor és csakis akkor tükörszám, ha egyenlő a tükörképével (fordítottjával). Például **1221** tükörszám.
- a képernyő második sorába egymástól szóközzel elválasztva kiírja a valódi osztók darabszámát az állomány mindegyik értékére;
- az **atestat.out** állomány első sorába beírja a számsorból az összes páratlan számot, szóközzel elválasztva.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

atestat.in		Kimeneti adatok:
6		a) 121 7 1551 5
121 14 7 1551 27 5		b) 1 2 0 6 2 0
		atestat.out tartalma:
	c)	121 7 1551 27 5

2. Tétel

Az **atestat.in** állomány két sort tartalmaz. Az első sorában egy nem nulla természetes szám van, n -el jelöljük ($5 < n < 30$). Az állomány második sora n darab egymástól szóközzel elválasztott, legtöbb 4 jegyű pozitív egész számot tartalmaz. A számsorban lesz legalább egy páros és egy páratlan szám.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyő első sorába kiírja azokat a számokat az állományból, amelyek prímszámok. Egy szám akkor és csakis akkor prímszám, ha pontosan két osztója van, 1 és önmaga. Az 1 nem prímszám.
- a képernyő második sorába egymástól szóközzel elválasztva kiírja a legnagyobb páros és a legkisebb páratlan számot, amelyik megtalálható az állomány értékei között;
- az **atestat.out** állományba beírja azt a két páratlan prímszámot, amelyek összege az előbbi alpontonál meghatározott legnagyobb páros szám.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

atestat.in		Kimeneti adatok:
6		a) 13 7 101 11 5
124 13 7 101 11 5		b) 124 5
		atestat.out tartalma:
	c)	101 23 vagy 113 11

3. Tétel

Az `atestat.in` állomány két sort tartalmaz. Az első sorában egy nem nulla természetes szám van, n -el jelöljük ($5 < n < 30$). Az állomány második sora n darab egymástól szóközzel elválasztott, legtöbb 9 jegyű természetes számot tartalmaz, n elemű természetes számsort alkotva.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyő első sorába szóközzel elválasztva kiírja az állomány értékei közül a páros számokat;
- a képernyőre különböző sorokba kiírja az adott számsorból a legkisebb **a** számot és a legnagyobb **b** számot;
- az `atestat.out` állományba beírja a b) alpontnál meghatározott **a** és **b** számok legnagyobb közös osztóját.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>		Kimeneti adatok:
6	a)	372 242 44
123 55 372 3465 242 44	b)	44 3465
		<code>atestat.out</code> tartalma:
	c)	11

4. Tétel

Az `atestat.in` állomány két sort tartalmaz. Az első sorában egy nem nulla természetes szám van, n -el jelöljük ($5 < n < 30$). Az állomány második sora n darab egymástól szóközzel elválasztott, legtöbb 9 jegyű természetes számot tartalmaz, n elemű természetes számsort alkotva.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyő első sorába szóközzel elválasztva kiírja az állomány értékei közül a páratlan számokat;
- a képernyő második sorába egymástól szóközzel elválasztva kiírja az összes olyan számot az állományból, amelyek csak páros számjegyeket tartalmaz. Ha nincs ilyen érték, akkor a „**Nem létezik**” üzenetet kell kiírja.
- az `atestat.out` állományba beírja azt a legnagyobb és azt a legkisebb természetes számot, amelyek az `atestat.out` első értékének különböző, nem nulla számjegyeiből építhető fel.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>		Kimeneti adatok:
6	a)	1233 1785 15657 457
1233 22 1785 56 15657 457	b)	22
		<code>atestat.out</code> tartalma:
	c)	321 123

5. Tétel

Az `atestat.in` állomány két sort tartalmaz. Az első sorában egy nem nulla természetes szám van, n -el jelöljük ($5 < n < 30$). Az állomány második sora n darab egymástól szóközzel elválasztott, legtöbb 4 jegyű természetes számot tartalmaz, n elemű természetes számsort alkotva.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyő első sorába szóközzel elválasztva kiírja az állomány értékei közül a páros számokat;
- a képernyő második sorába egymástól szóközzel elválasztva kiírja az összes olyan számot az állományból, amelyek csak páratlan számjegyeket tartalmaz. Ha nincs ilyen érték, akkor a „**Nem létezik**” üzenetet kell kiírja.
- az `atestat.out` állományba beírja a számsorból az összes olyan számot, amelyekre a prím osztóinak a száma k ; a k természetes számot a billentyűzetről olvassuk be.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>						Kimeneti adatok:				
6						a)	30	8	42	700
30	8	42	15	700	37	b)	15	37		
a beolvasott érték $k=3$						c)	30	42	700	
						atestat.out tartalma:				

6. Tétel

Az `atestat.in` állomány két sort tartalmaz. Az első sorában egy nem nulla természetes szám van, n -el jelöljük ($5 < n < 30$). Az állomány második sora n darab egymástól szóközzel elválasztott, valós számot tartalmaz, n elemű valós számsort alkotva.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyő első sorába szóközzel elválasztva kiírja az összes számot az állományból;
- a képernyő következő sorába kiírja 2 tizedes pontossággal a megadott számsor negatív értékeinek a számtani középátlóját (ha nincs negatív szám a megadott számsorban, akkor 0 ír ki);
- a billentyűzetről beolvas két nem nulla természetes számot p_1 és p_2 ($1 < p_1 < p_2 < n$), növekvő sorrendbe rendezi azokat a számokat az állományból, amelyek sorszáma (indexe) p_1 és p_2 között van (p_1 -t és p_2 -t is beleértve), majd az `atestat.out` állomány első sorába szóközzel elválasztva beírja az új számsort.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa: a billentyűzetről beolvasott értékek: $p_1=2$ és $p_2=4$

<code>atestat.in</code>						Kimeneti adatok:						
6						a)	-56.765	2.3	4.56	-1.2	-1.8	3
-56.765	2.3	4.56	-1.2	-1.8	3	b)	-19.92					
a beolvasott értékek $p_1=2$ $p_2=4$						c)	-56.765	-1.2	2.3	4.56	-1.8	3
						atestat.out tartalma:						

7. Tétel

Az `atestat.in` állomány két sort tartalmaz. Az első sorában egy nem nulla természetes szám van, n -el jelöljük ($5 < n < 30$). Az állomány második sora n darab egymástól szóközzel elválasztott, legtöbb 4 jegyű természetes számot tartalmaz, n elemű természetes számsort alkotva. A számsor tartalmaz legalább két páratlan számot.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyő első sorába szóközzel elválasztva kiírja az olvasás fordított sorrendjében az összes számot az állományból;
- a képernyő második sorába egymástól szóközzel elválasztva kiírja, hogy az eredeti számsorból hány számjegye van mindegyik értéknek;
- az `atestat.out` állomány első sorába beírja a megadott számsor összes páratlan értékének az összegét.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>		Kimeneti adatok:
6	a)	1001 242 2 71 555 13
13 555 71 2 242 1001	b)	2 3 2 1 3 4
		<code>atestat.out</code> tartalma:
	c)	1640

8. Tétel

Az `atestat.in` állomány két sort tartalmaz. Az első sorában egy nem nulla természetes szám van, n -el jelöljük ($5 < n < 30$). Az állomány második sora n darab egymástól szóközzel elválasztott, legtöbb 4 jegyű egész számot tartalmaz, n elemű egész számsort alkotva. A számsor értékei *csökkenő sorrendbe* vannak megadva.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyő első sorába szóközzel elválasztva kiírja az összes számot az állományból az olvasás fordított sorrendjében;
- a képernyő második sorába kiírja az **”IGEN”** vagy **„NEM”** üzenetet, amely megfelel a billentyűzetről beolvasott x szám a megadott számsorban való keresése eredményének;
- az `atestat.out` állomány első sorába beírja a megadott számsor összes pozitív értékének az összegét.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>		Kimeneti adatok:
6	a)	2 13 71 242 555 1001
1001 555 242 71 13 2	b)	IGEN
A beolvasott érték $x=555$		<code>atestat.out</code> tartalma:
	c)	1884

9. Tétel

Az `atestat.in` állomány első sorában egy nem nulla természetes szám van, n -el jelöljük ($2 < n < 20$) és egy négyzetes A mátrix sorainak és oszlopainak a számát jelenti. A következő n sor mindegyike n darab egymástól szóközzel elválasztott, pontosan 2 jegyű természetes számot tartalmaz, amelyek az A négyzetes mátrix elemeit jelentik.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyőre kiírja az A mátrix elemeit, sorról sorra, egy soron belül egy szóközzel vannak elválasztva a számok;
- megváltoztatja az A mátrixot úgy, hogy a mellékátló összes elemének az értéke a billentyűzetről megadott k szám legyen és a képernyőre kiírja a megváltoztatott mátrixot;
- az `atestat.out` állomány első sorába egymástól szóközzel elválasztva beírja a megváltoztatott mátrixból az összes prímszámot. Ha a mátrixban egy prímszám sincs, akkor az állomány első sorába a „**NEM**” üzenetet írja.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa: ahol $k=36$

<code>atestat.in</code>		<u>Kimeneti adatok:</u>		
3	a)	12	31	24
12 31 24		35	67	10
35 67 10		41	18	97
41 18 97				
	b)	12	31	36
		35	36	10
		36	18	97
a beolvasott érték				
k=36				
	c)	<code>atestat.out</code> tartalma:		
		31	97	

10. Tétel

Az `atestat.in` állomány első sorában két nem nulla természetes szám van, n -el és m -el jelöljük ($2 < n, m < 20$) és egy A mátrix sorainak illetve oszlopainak a számát jelenti. A következő n sor mindegyike m darab, egymástól szóközzel elválasztott, természetes számot tartalmaz, amelyek az A mátrix elemeit jelentik. Az állomány utolsó sorában egy k -val jelölt érték van.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyőre kiírja az A mátrix elemeit, sorról sorra, egy soron belül egy szóközzel vannak elválasztva a számok;
- a képernyőre kiírja az A mátrix azon sorait, amelyekben a nulla előfordulásának a száma k . Ha nem létezik ilyen sor akkor kiírja a „**Nem létezik**” üzenetet;
- az `atestat.out` állomány első sorába egymástól szóközzel elválasztva beírja az összes páratlan számot a mátrix c oszlopából, ahol a c -t a billentyűzetről olvassa be. Ha nincs ilyen érték, akkor az állomány első sorába a „**NEM**” üzenetet írja.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>		Kimeneti adatok:
4 3	a)	0 0 24
0 0 24		0 67 10
0 67 10		41 0 0
41 0 0		212 95 101
212 95 101		
2	b)	1 3
		<code>atestat.out</code> tartalma:
A beolvasott szám $c=2$	c)	67 95

11. Tétel

Az **atestat.in** állomány első sorában egy nem nulla természetes szám van, **n**-el jelöljük ($2 < n < 20$) és egy négyzetes **A** mátrix sorainak és oszlopainak a számát jelenti. A következő **n** sor mindegyike **n** darab, egymástól szóközzel elválasztott, természetes számot tartalmaz, amelyek az **A** négyzetes mátrix elemeit jelentik.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyőre kiírja az **A** mátrix elemeit, sorról sorra, egy soron belül egy szóközzel vannak elválasztva a számok;
- a képernyő következő sorába kiírja az **A** mátrix főátló alatti elemei szorzatának az utolsó számjegyét;
- az **atestat.out** állomány első sorába egymástól szóközzel elválasztva beírja az összes tükörszámot (palindrom) a mátrixból. Egy szám palindrom, ha a számot balról jobbra olvasva kapott érték megegyezik a jobbról balra olvasott számmal, például: **11**, **22**, **33**. Ha nincs ilyen érték, akkor az állomány első sorába a „**Nem létezik**” üzenetet írja.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

atestat.in		Kimeneti adatok:		
3	a)	15	20	37
15 20 37		10	21	44
10 21 44		12	40	15
12 40 15	b)	0		
		atestat.out tartalma:		
	c)	44		

12. Tétel

Az **atestat.in** állomány első sorában egy nem nulla természetes szám van, **n**-el jelöljük ($2 < n < 20$) és egy négyzetes **A** mátrix sorainak és oszlopainak a számát jelenti. A következő **n** sor mindegyike **n** darab egymástól szóközzel elválasztott, legtöbb 4 jegyű természetes számot tartalmaz, amelyek az **A** négyzetes mátrix elemeit jelentik.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyőre kiírja az **A** mátrix elemeit, sorról sorra, egy soron belül egy szóközzel vannak elválasztva a számok;
- a képernyő következő sorába kiírja az **A** mátrix mellékátló elemeinek az összegét;
- az **atestat.out** állomány első sorába egymástól szóközzel elválasztva beírja a mátrix mindegyik sorának a legkisebb értékét.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

atestat.in		Kimeneti adatok:
3	a)	12 31 24
12 31 24		35 67 10
35 67 10		28 18 97
28 18 97	b)	119
	c)	atestat.out tartalma:
		12 10 18

13. Tétel

Az `atestat.in` állomány első sorában egy nem nulla természetes szám van, n -el jelöljük ($2 < n < 20$) és egy négyzetes A mátrix sorainak és oszlopainak a számát jelenti. A következő n sor mindegyike n darab egymástól szóközzel elválasztott, legtöbb 9 jegyű egész számot tartalmaz, amelyek az A négyzetes mátrix elemeit jelentik.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyőre kiírja az A mátrix elemeit, sorról sorra, egy soron belül egy szóközzel vannak elválasztva a számok;
- a billentyűzetről beolvass egy k -val jelölt legtöbb 9 jegyű természetes számot és ellenőrzi, hogy a megadott szám megtalálható-e az A mátrixban, majd a képernyőre kiír egy, a keresés eredményének megfelelő üzenetet;
- az A mátrixot megváltoztatja úgy, hogy a mátrix utolsó sorában minden érték 0 lesz, majd az `atestat.out` állományba írja a megváltoztatott mátrixot, sorról sorra, egy soron belül egy szóközzel vannak elválasztva a számok.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>		Kimeneti adatok:		
3	a)	-1	311	234
-1 311 234		13	-9	67
13 -9 67		5	-92	4
5 -92 4	b)	IGEN		
Billentyűzetről beolvassa $k = 13$	c)	atestat.out tartalma:		
		-1	311	234
		13	-9	67
		0	0	0

14. Tétel

Az **atestat.in** állomány első sorában egy nem nulla természetes szám van, **n**-el jelöljük ($2 < n < 20$) és egy négyzetes **A** mátrix sorainak és oszlopainak a számát mutatja. A következő **n** sor mindegyike **n** darab egymástól szóközzel elválasztott valós számot tartalmaz, amelyek a mátrix elemeit jelentik.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyőre kiírja az **A** mátrix elemeit, sorról sorra, egy soron belül egy szóközzel vannak elválasztva a számok;
- a képernyő következő sorába kiírja az **A** mátrix főátló elemeinek a számtani középátlóját;
- az **atestat.out** állomány első sorába beírja a mátrix **k** sorában lévő elemek összegét, ahol **k** ($0 < k \leq n$) egy billentyűzetről beolvasott természetes szám.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

atestat.in		Kimeneti adatok:
3	a)	1.29 12.4 9.46
1.29 12.4 9.46		3.1 23.2 -5.6
3.1 23.2 -5.6		-9.9 8.5 -8.2
-9.9 8.5 -8.2	b)	5.43
Billentyűzetről beolvassa k=2	c)	atestat.out tartalma: 20.7

15. Tétel

Az **atestat.in** állomány egyetlen, legtöbb **250** karakterből álló sort tartalmaz, az első karakter betű, a szavak a szövegben egy szóközzel vannak elválasztva egymástól. A szavak csak az angol ábécé kis- és nagybetűiből állnak.

Írj egy *Pascal/C/C++* programot, amely:

- a képernyőre írja a szöveget mind nagy betűkkel;
- a képernyő következő sorába írja a szöveg mássalhangzóinak a darabszámát;
- az **atestat.out** állomány első sorába beírja, hogy hány szó van a szövegben.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

atestat.in	A mikroprocesszor a számítógép Agya
Kimeneti adatok:	
- képernyőn :	a) A MIKROPROCESSZOR A SZAMITOGEP AGYA
	b) 18
- atestat.out	c) 5

16. Tétel

Az `atestat.in` állomány egyetlen, legtöbb **100** karakterből álló sort tartalmaz, az első karakter betű, a szavak a szövegben egy szóközzel vannak elválasztva egymástól. A szavak csak az angol ábécé kis- és nagybetűiből állnak.

Írj egy *Pascal/C/C++* programot, amelyik:

- a képernyőre egy sorba kiírja a szöveget csupa nagybetűvel;
- a képernyő következő sorába kiírja a magánhangzók számát;
- az `atestat.out` állomány első sorába beírja az eredeti szövegből az 'a' betűvel kezdődő szavak számát.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>	Aranyalma aghegyen bari beg a zold gyepen
Kimeneti adatok:	a) ARANYALMA AGHEGYEN BARI BEG A ZOLD GYEPEN b) 14
	Az <code>atestat.out</code> tartalma: c) 2

17. Tétel

Az `atestat.in` állomány egyetlen, legtöbb **70** karakterből álló sort tartalmaz, az első karakter betű, a szavak a szövegben egy szóközzel vannak elválasztva egymástól. A szavak csak az angol ábécé nagybetűiből állnak.

Írj egy *Pascal/C/C++* programot, amelyik:

- a képernyőre egy sorba kiírja a szöveget csupa kisbetűvel;
- a képernyő következő sorába kiírja a szövegben szereplő szavak számát;
- olvass be a billentyűzetről egy betűt, minden 'K' betűvel kezdődő szó első betűjét cseréld le a billentyűzetről beolvasott betűre és írd be az `atestat.out` állomány első sorába a módosított szöveget.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>	Kimeneti adatok:
KET PEK KET SZEP KEPET KER	a) ket pek ket szep kepet ker b) 6
A billentyűzetről beolvasott betű 'H'	Az <code>atestat.out</code> tartalma: c) HET PEK HET SZEP HEPET HER

18. Tétel

Az `atestat.in` állomány egyetlen, legtöbb **100** karakterből álló sort tartalmaz, az első karakter betű, a szavak a szövegben egy szóközzel vannak elválasztva egymástól. A szavak csak az angol ábécé kis- és nagybetűiből vagy számjegyekből állnak.

Írj egy *Pascal/C/C++* programot, amelyik:

- a képernyőre egy sorba kiírja a szöveget csupa kisbetűvel;
- kiírja a képernyőre, külön sorokba, az eredeti szövegben szereplő szavakat;
- az `atestat.out` állomány első sorába beírja a szövegben szereplő számjegyek számát.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>	Az OSZTALYBAN 30 tanulo VAN ebbol 10 Lany
Kimeneti adatok:	
a)	az osztalyban 30 tanulo van ebbol 10 lany
b)	Az OSZTALYBAN 30 tanulo VAN ebbol 10 Lany
<code>atestat.out</code>	c) 4

19. Tétel

Az `atestat.in` állomány egyetlen, legtöbb **50** karakterből álló sort tartalmaz, az első és utolsó karakter betű, a szavak a szövegben egy szóközzel vannak elválasztva egymástól. A szavak csak az angol ábécé kis- és nagybetűiből állnak.

Írj egy *Pascal/C/C++* programot, amelyik:

- a képernyőre egy sorba kiírja a szöveget csupa nagybetűvel;
- a képernyő következő sorába kiírja az eredeti szöveget, de szóközők nélkül;
- beírja az `atestat.out` állományba háromszögformában az összes középrészsort, amelyiket a szöveg első szavából kiindulva alkothatunk egészen a szó közepén található egy vagy két karakterig.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>	Kimeneti adatok:
Vizilovunk NAGYON szereti a Vizez	a) VIZILOVUNK NAGYON SZERETI A VIZET b) VizilovunkNAGYONSzeretiaVizez
	Az <code>atestat.out</code> tartalma:
	c) Vizilovunk izilovun zilovu ilov lo

20. Tétel

Az **atestat.in** állomány egyetlen, legtöbb **100** karakterből álló sort tartalmaz, az első és utolsó karakter betű, a szavak a szövegben egy szóközzel vannak elválasztva egymástól. A szavak csak az angol ábécé kis- és nagybetűiből állnak.

Írj egy *Pascal/C/C++* programot, amelyik:

- a képernyőre egy sorba kiírja a szöveget csupa nagybetűvel;
- a képernyő következő sorába kiírja az eredeti szöveget, de szóközők nélkül;
- az atestat.out állományba írd be az anagramma párokat. Két szó anagramma akkor, ha egyforma a hosszuk és ugyanazokból a betűkből állnak, esetleg más sorrendben. A két szó összehasonlításánál nem teszünk különbséget kis és nagybetűk között.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

atestat.in	Kimeneti adatok:
Komor romok Let maradtam Hadartam Tel	a) KOMOR ROMOK LET MARADTAM HADARTAM TEL b) KomorromokLetmaradtamHadartamTel
	Az atestat.out tartalma: c) Komor romok Let Tel

21. Tétel

Az **atestat.in** állomány egyetlen, legtöbb **100** karakterből álló sort tartalmaz, az első és utolsó karakter betű, a szavak a szövegben egy szóközzel vannak elválasztva egymástól. A szavak csak az angol ábécé kis- és nagybetűiből állnak.

Írj egy *Pascal/C/C++* programot, amelyik:

- a képernyőre egy sorba kiírja a szöveget csupa kisbetűvel;
- a képernyő következő sorába kiírja az eredeti szöveget, de szóközők nélkül;
- az atestat.out állományba írd be a tükörszavakat (palindrom szó). Egy szó palindrom, ha bármilyen két, a közepétől egyforma távolságra található betűje azonos (oda-vissza ugyanaz). Nem teszünk különbséget kis és nagybetűk között.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

atestat.in	Kimeneti adatok:
Anna hazament inni es Enni	a) anna hazament inni es enni b) AnnahazamentinniesEnni
	Az atestat.out tartalma: c) Anna inni

22. Tétel

Az **atestat.in** állomány első sorában az **n** ($2 < n < 10$) nullától különböző természetes szám található. A következő **n** sor mindegyikében két nullától különböző **a** és **b** szám pár található, jelentésük egy-egy algebrai tört számlálója (**a**) és nevezője (**b**). A két számot egymástól egy szóközzel választja el.

Írj egy *Pascal/C/C++* programot, amelyik:

- a képernyőre egy sorba kiírja az **n** törtet **a/b** (számláló/nevező) alakban egy-egy szóközzel elválasztva;
- a képernyő következő sorába kiírja az első olyan törtet, amelyiknek legnagyobb a számlálója;
- az **atestat.out** állomány első sorába be kell írni az **atestat.in** állományból beolvasott összes irreducibilis alakú törtet **a/b** alakban, egymástól szóközzel elválasztva.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Összetett adattípust (rekordot, struktúrát) kell használni.

Példa:

atestat.in		Kimeneti adatok:
6	a)	2/3 9/4 7/4 9/10 1/5 2/8
2 3	b)	9/4
9 4		Az atestat.out tartalma:
7 4	c)	2/3 9/4 7/4 9/10 1/5
9 10		
1 5		
2 8		

23. Tétel

Az `atestat.in` állomány első sorában az n ($2 < n < 10$) nullától különböző természetes szám található. A következő n sor mindegyikében x és y természetes szám pár található, jelentésük az xOy síkban egy pont koordinátái. A két számot egymástól egy szóköz választja el.

Írj egy *Pascal/C/C++* programot, amelyik:

- a képernyőre kiírja az xOy síkbeli pontokat (x,y) alakban, minden pontot külön sorba;
- a képernyő következő sorába kiírja az Ox tengelyen található pontok számát;
- az `atestat.out` állományba külön sorokba beírja (x,y) alakban azokat a pontokat, amelyeknek mindkét koordinátája páros szám.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Összetett adattípust (rekordot, struktúrát) kell használni.

Példa:

<code>atestat.in</code>		Kimeneti adatok:
8		a) (0 , 0)
0	0	(1 , 2)
1	2	(3 , 0)
3	0	(2 , 2)
2	2	(2 , 3)
2	3	(0 , 4)
0	4	(6 , 8)
6	8	(5 , 0)
5	0	b) Az Ox tengelyen található pontok száma 3
		Az <code>atestat.out</code> tartalma:
		c) (0 , 0)
		(2 , 2)
		(0 , 4)
		(6 , 8)

24. Tétel

Az `atestat.in` állomány első sorában egy n ($1 < n < 10$) természetes szám található, a következő n sorban pedig egy-egy szó, mindegyik az angol ábécé nagybetűiből áll.

Írj egy *Pascal/C/C++* programot, amelyik:

- a képernyő első sorába kiírja a szavakat csupa kisbetűvel;
- a képernyő következő sorába kiírja az eredetiek közül a leghosszabb szót;
- az `atestat.out` állomány első sorába írj be $n+1$ karaktert, amelyeket a következő szabály szerint kapsz: az első karakter az első szó első betűje, a második karakter a második szó második betűje, ..., az utolsó karakter „.” lesz. Ha a k -ik szó nem tartalmaz legalább k darab betűt, akkor az eredménybe a k -adik pozícióba egy szóköz kerül.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>	Kimeneti adatok:
7	a) rum ler letelem vetemenyes zsirsertesek alma lapzarta
RUM	b) ZSIRSERTESEK
LER	Az atestat.out tartalma:
LETELEM	c) RETES T.
VETEMENYES	
ZSIRSERTESEK	
ALMA	
LAPZARTA	

25. Tétel

Az `atestat.in` állomány első sorában az n ($2 < n < 10$) nem nulla természetes szám található, egy *nem irányított gráf* csomópontjainak számát jelenti. A következő sorokban az állomány végéig szám párok vannak $x\ y$ formában, jelentésük pedig az x csomópont egy éllel össze van kötve az y csomóponttal (az x és y csomópontok szomszédosak).

Írj egy *Pascal/C/C++* programot, amelyik:

- kiírja a képernyőre a beolvasott gráf szomszédossági mátrixát;
- beolvas a billentyűzetről egy v természetes számot – a gráf egyik csomópontját jelenti – és kiírja a beolvasott csomópont fokszámát;
- módosítja az A szomszédossági mátrixot úgy, hogy a kapott gráf teljes legyen, majd kiírja az így kapott mátrixot a `atestat.out` állományba, soronként, a soron belül az elemeket egymástól szóköz válassza el.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>		Kimeneti adatok:
4	a)	0 0 1 1
1 3		0 0 1 0
1 4		1 1 0 0
2 3		1 0 0 0
	b)	1
A billentyűzetről beolvasott érték $v = 2$	c)	Az <code>atestat.out</code> tartalma:
		0 1 1 1
		1 0 1 1
		1 1 0 1
		1 1 1 0

26. Tétel

Az **atestat.in** állomány első sorában az **n** ($2 < n < 10$) nem nulla természetes szám található, egy *irányított gráf* csomópontjainak számát jelenti. A következő sorokban az állomány végéig szám párok vannak **x y** formában, jelentésük pedig az **(x,y)** irányított él a gráfban.

Írj egy *Pascal/C/C++* programot, amelyik:

- kiírja a képernyőre a beolvasott gráf szomszédossági mátrixát;
- beolvas a billentyűzetről egy **v** természetes számot – a gráf egyik csomópontját jelenti – és kiírja a beolvasott csomópont ki- és befokát (kimenő és bejövő élek számát);
- az **atestat.out** állományba írd ki az összes olyan **2** hosszúságú irányított utat, amelyik a fenti **v** billentyűzetről beolvasott csomópontból indul.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

atestat.in		Kimeneti adatok:
5	a)	0 1 0 0 1
1 2		0 0 1 1 0
1 5		0 0 0 0 0
2 3		0 0 0 0 0
2 4		1 0 1 0 0
5 3		
5 1		
	b)	1 2
	c)	Az atestat.out tartalma:
		1 2 3
		1 2 4
		1 5 3

A billentyűzetről beolvasott érték **v = 1**

27. Tétel

Az `atestat.in` állomány első sorában az n ($2 < n < 10$) nem nulla természetes szám található, egy *nem irányított gráf* csomópontjainak számát jelenti. A következő sorokban az állomány végéig szám párok vannak $x\ y$ formában, jelentésük pedig az x csomópont egy éllel össze van kötve az y csomóponttal (az x és y csomópontok szomszédosak).

Írj egy *Pascal/C/C++* programot, amelyik:

- kiírja a képernyőre a beolvasott gráf szomszédossági mátrixát;
- kiírja a képernyőre a gráf izolált (elszigetelt) pontjait (ha léteznek ilyenek);
- módosítja az A szomszédossági mátrixot úgy, hogy billentyűzetről beolvasott v csomópont legyen szomszédos az összes többi csomóponttal, és az így kapott mátrixot írd ki az `atestat.out` állományba.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>		Kimeneti adatok:
5	a)	0 0 1 1 0
1 3		0 0 1 0 0
1 4		1 1 0 0 0
2 3		1 0 0 0 0
		0 0 0 0 0
	b)	5
A billentyűzetről beolvasott érték $v = 2$	c)	Az <code>atestat.out</code> tartalma:
		0 1 1 1 0
		1 0 1 1 1
		1 1 0 0 0
		1 1 0 0 0
		0 1 0 0 0

28. Tétel

Az `atestat.in` állomány első sorában az n ($2 < n < 10$) nem nulla természetes szám található, egy *nem irányított gráf* csomópontjainak számát jelenti. A következő sorokban az állomány végéig szám párok vannak $x\ y$ formában, jelentésük pedig az x csomópont egy éllel össze van kötve az y csomóponttal (az x és y csomópontok szomszédosak).

Írj egy *Pascal/C/C++* programot, amelyik:

- kiírja a képernyőre a beolvasott gráf szomszédossági mátrixát;
- kiírja a képernyőre a maximális fokszámú csomópontot (csomópontokat);
- módosítja az A szomszédossági mátrixot úgy, hogy a billentyűzetről beolvasott v csomópont legyen izolált (elszigetelt) pont, majd kiírja az így kapott mátrixot az `atestat.out` állományba.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>		Kimeneti adatok:
5	a)	0 1 1 1 0
1 2		1 0 1 0 0
1 3		1 1 0 1 0
1 4		1 0 1 0 1
2 3		0 0 0 1 0
3 4		
4 5		
	b)	1 3 4
A billentyűzetről beolvasott érték $v = 2$	c)	Az <code>atestat.out</code> tartalma:
		0 0 1 1 0
		0 0 0 0 0
		1 0 0 1 0
		1 0 1 0 1
		0 0 0 1 0

29. Tétel

Az `atestat.in` állomány első sorában az n ($2 < n < 10$) nem nulla természetes szám található, egy *nem irányított gráf* csomópontjainak számát jelenti. A következő sorokban az állomány végéig szám párok vannak $x\ y$ formában, jelentésük pedig az x csomópont egy éllel össze van kötve az y csomóponttal (az x és y csomópontok szomszédosak).

Írj egy *Pascal/C/C++* programot, amelyik:

- kiírja a képernyőre a beolvasott gráf szomszédossági mátrixát;
- beolvas a billentyűzetről egy számsort és eldönti, hogy a beolvasott csomópontok sorozata lehet-e séta a gráfban, ekkor a „IGEN” üzenetet írja ki, ellenkező esetben a „NEM” üzenetet.
- beolvassa a billentyűzetről a v csomópontot és az `atestat.out` állományba kiírja a beolvasott csomópont szomszédossági listáját.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

<code>atestat.in</code>		Kimeneti adatok:
5	a)	0 1 1 1 0
1 2		1 0 1 0 0
1 3		1 1 0 1 0
1 4		1 0 1 0 1
2 3		0 0 0 1 0
3 4		
4 5		
A billentyűzetről beolvasott számsor 1 4 3 2		b) IGEN
v=3		c) Az <code>atestat.out</code> tartalma: 1 2 4

30. Tétel

Az **atestat.in** állomány első sorában az **n** ($2 < n < 10$) nem nulla természetes szám található, egy *irányított gráf* csomópontjainak számát jelenti. A következő sorokban az állomány végéig szám párok vannak **x y** formában, jelentésük pedig az **(x,y)** irányított él a gráfban.

Írj egy *Pascal/C/C++* programot, amelyik:

- kiírja a képernyőre a beolvasott gráf szomszédossági mátrixát;
- kiírja a képernyőre az összes olyan csomópontot, amelyekre a kimenő élek száma egyenlő a bejövő élek számával (a kifok és a befok egyforma);
- beolvas a billentyűzetről egy számsort és eldönti, hogy a beolvasott csomópontok sorozata lehet-e irányított út a gráfban, ekkor a „IGEN” üzenetet írja ki az atestat.out állományba, ellenkező esetben a „NEM” üzenetet.

Megjegyzés: A programnak tartalmaznia kell legalább egy felhasználói alprogramot.

Példa:

atestat.in		Kimeneti adatok:
5	a)	0 1 0 0 1
1 2		0 0 1 1 0
1 5		0 0 0 0 0
2 3		0 0 0 0 0
2 4		1 1 1 0 0
5 3		
5 1		
5 2		
A billentyűzetről beolvasott számsor 1 2 3		b) 2
		c) Az atestat.out tartalma: IGEN