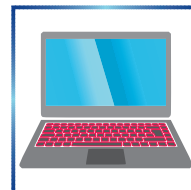


## Informatikából kitűzött feladatok



**I. 577.** Egyenletesnek nevezzük azt az  $N$  jegyű számot, amelyben az egyes számjegyek értéke az előző helyiértéken álló számjegytől legfeljebb 1-gyel tér el.

Írjunk programot, amely egy beolvasott  $N$ -jegyű nem egyenletes számhoz meghatározza a nála kisebb legnagyobb és a nála nagyobb legkisebb egyenletes számot.

A program az  $N$ -jegyű ( $2 \leq N \leq 1\,000\,000$ ) egész számot a standard bemenetről olvassa be.

A standard kimenet első sorában a beolvasott számnál kisebb legnagyobb egyenletes számot, a második sorában pedig a nála nagyobb legkisebb egyenletes számot írjuk ki.

Példa bemenet	Kimenet
8778349	8777899 8778765

*Értékelés:* a megoldás lényegét leíró dokumentáció 1 pontot ér. 5 pont kapható arra a programra, amely a  $2 \leq N \leq 1\,000$  bemenetekre helyes kimenetet ad. További 4 pont kapható a  $2 \leq N \leq 1\,000\,000$  bemenetekre 1 másodperc futásidő alatt helyes kimenetet adó programokra.

Beküldendő egy tömörített `i577.zip` állományban a program forráskódja, valamint a program rövid dokumentációja, amely tartalmazza a megoldás rövid leírását, és megadja, hogy a forrásállomány melyik fejlesztői környezetben fordítható.

**I. 578 (É).** A gazdaság helyzetéről rengeteget elárul a közlekedési infrastruktúra. Ebben a feladatban hazánk 2007 és 2021 közötti megyénkénti adatait fogjuk górcső alá venni.

- Hozzunk létre a táblázatkezelőben egy új munkafüzetet. Ennek két munkalapja legyen **Adatok** és **Valasz**. Ezekre az A1-es cellától kezdve töltjük be a munkalapok nevével megegyező, UTF-8 kódolású, tabulátorral tagolt txt-fájlokat.
- A két munkalapon végezzük el az alábbi mintákon látható formázásokat, és oldjuk meg a leírás szerinti feladatokat.
- Az **Adatok** munkalapon rendezzük át a megyék adatait a megye neve szerint névsorba.
- Az évenkénti legnagyobb fejlesztés adatait határozzuk meg a **Valasz** munkalap B3:G3 tartományában, tehát azt, hogy az előző évihez képest mikor és hol volt a legnagyobb növekedés, és mekkora volt a hossznövekedés értéke, mind köz-, mind vasúton.

- Határozzuk meg a **Valasz** munkalap B7:G7 tartományában azt, hogy a teljes időszakra az egész országot tekintve hogyan és mennyivel változott az úthossz, mind köz-, mind vasúton. A B7 és E7 cellákban e három szöveg valamelyike szerepelhet: „nőtt”, „nem változott” vagy „csökkent”.
- A B10 és E10 cellákba azon megyék neve kerüljön, amelyeknél 2012-ben a legnagyobb volt a négyetkilométerenkénti úthossz, mind köz-, mind vasúton.
- A B12 cellába számoljuk ki hazánk területét az adatok alapján.
- AZ A16, A17, ... cellákba kerüljön azon megyék neve, amelyekben az időszak végén, 2021-ben nem volt nagyobb a közúti és vasúti útvonalhossz összege, mint 2007-ben.
- Feltételes formázással oldjuk meg, hogy az A16:A34 tartomány cellái közül csak azok kapjanak a minta szerinti keretet és tónust, amelyek nem üresek.
- Egy új munkalagra készítsünk a minta szerint diagramot Békés, Csongrád–Csanád és Fejér megye közúti hosszának alakulásáról.

Az adatok forrása: a KSH

[https://www.ksh.hu/stadat\\_files/sza/hu/sza0041.html](https://www.ksh.hu/stadat_files/sza/hu/sza0041.html);

[https://www.ksh.hu/stadat\\_files/sza/hu/sza0039.html](https://www.ksh.hu/stadat_files/sza/hu/sza0039.html)

és [https://www.ksh.hu/stadat\\_files/fo1/hu/fo10006.html](https://www.ksh.hu/stadat_files/fo1/hu/fo10006.html) oldalai.

*Minták:*

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	<b>Területi egység neve</b>	<b>Terület [km<sup>2</sup>]</b>	<b>2007</b>	<b>2008</b>	<b>2009</b>	<b>2010</b>	<b>2011</b>	<b>2012</b>	<b>2013</b>	<b>2014</b>	<b>2015</b>	<b>2016</b>	<b>2017</b>	<b>2018</b>
2			<b>Normál nyomtávú vasútak hossza [kilométer]</b>											
3	<i>Pest (Budapesttel együtt)</i>	6 917,1	813	768	768	771	771	771	771	771	771	693	679	679
4	<i>Fejér</i>	4 358,6	401	492	492	492	492	492	492	492	492	454	455	455
5	<i>Komárom-Esztergom</i>	2 264,3	117	225	225	225	225	225	225	225	225	225	226	226
6	<i>Veszprém</i>	4 463,8	562	434	434	433	433	433	433	433	433	435	436	436
7	<i>Győr-Moson-Sopron</i>	4 207,7	306	383	383	414	402	402	391	391	424	367	367	367
8	<i>Vas</i>	3 336,1	303	322	322	323	324	324	307	307	305	282	282	282
9	<i>Zala</i>	3 783,8	267	267	267	268	274	274	274	274	271	261	262	262
10	<i>Baranya</i>	4 429,7	346	384	384	381	381	381	381	381	381	344	345	345
11	<i>Somogy</i>	6 065,1	532	475	475	475	475	475	475	475	475	493	494	494
12	<i>Tolna</i>	3 703,2	235	237	237	206	206	206	206	206	206	245	246	246
13	<i>Borsod-Abaúj-Zemplén</i>	7 247,1	528	510	510	510	510	510	510	510	510	463	464	464
14	<i>Heves</i>	3 637,2	292	283	283	283	283	283	283	283	283	301	301	301
15	<i>Nógrád</i>	2 544,5	128	133	133	133	133	133	133	133	133	200	201	201
16	<i>Hajdú-Bihar</i>	6 210,8	438	496	496	469	469	469	469	469	469	470	471	471
17	<i>Jász-Nagykun-Szolnok</i>	5 581,6	477	477	477	503	503	503	503	503	503	452	454	454
18	<i>Szabolcs-Szatmár-Bereg</i>	5 935,9	663	391	391	391	391	391	391	391	391	495	496	496
19	<i>Bács-Kiskun</i>	8 444,9	687	543	543	543	543	543	543	543	543	546	548	548
20	<i>Békés</i>	5 629,7	433	445	445	445	445	445	445	445	445	401	402	402
21	<i>Csongrád-Csanád</i>	4 264,6	340	310	310	310	310	310	310	310	310	311	312	312

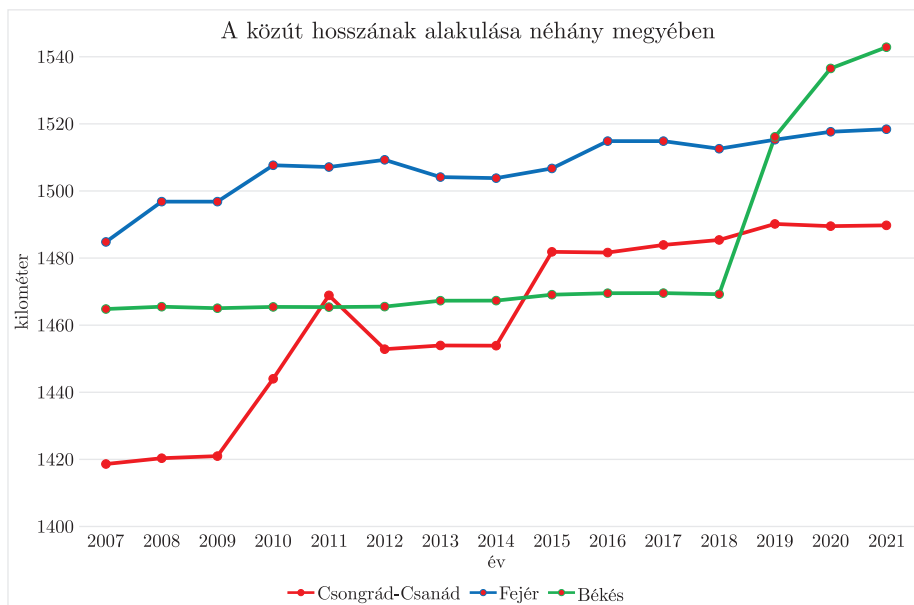
**Adatok** munkalap

	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF
1	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021
2	<b>A közutak hossza [kilométer]</b>														
3	Pes	2 685	2 760	2 757	2 771	2 782	2 793	2 792	2 788	2 785	2 786	2 801	2 833	2 830	
4	Fe	1 485	1 497	1 497	1 508	1 507	1 509	1 504	1 507	1 515	1 515	1 513	1 515	1 518	1 518
5	Ka	890	890	891	891	895	893	893	895	896	896	895	907	909	
6	Ves	1 629	1 644	1 649	1 650	1 650	1 647	1 635	1 639	1 640	1 639	1 645	1 649	1 645	1 648
7	Gye	1 741	1 751	1 752	1 752	1 762	1 762	1 766	1 806	1 833	1 844	1 847	1 847	1 924	1 929
8	Vas	1 525	1 525	1 536	1 540	1 542	1 541	1 541	1 550	1 566	1 581	1 581	1 573	1 574	1 620
9	Zala	1 686	1 722	1 723	1 726	1 725	1 725	1 733	1 737	1 741	1 752	1 752	1 750	1 750	1 747
10	Bal	1 643	1 643	1 642	1 717	1 717	1 720	1 718	1 723	1 725	1 724	1 724	1 723	1 723	1 723
11	Son	1 736	1 762	1 762	1 761	1 766	1 777	1 790	1 795	1 799	1 796	1 795	1 795	1 826	1 821
12	Tol	1 087	1 087	1 087	1 216	1 216	1 208	1 209	1 209	1 209	1 209	1 209	1 209	1 209	1 209
13	Bor	2 582	2 591	2 587	2 588	2 587	2 585	2 583	2 586	2 586	2 588	2 588	2 584	2 585	2 648
14	He	1 275	1 274	1 274	1 274	1 274	1 271	1 270	1 270	1 272	1 271	1 269	1 277	1 297	1 295
15	Nó	945	946	946	944	944	948	947	948	948	948	951	954	954	957
16	Haj	1 669	1 669	1 669	1 669	1 669	1 667	1 667	1 662	1 662	1 663	1 676	1 717	1 718	1 754
17	Jás	1 316	1 316	1 319	1 320	1 330	1 330	1 329	1 329	1 331	1 331	1 331	1 331	1 359	1 366
18	Szo	2 150	2 152	2 152	2 152	2 157	2 156	2 208	2 227	2 235	2 236	2 239	2 242	2 247	2 236
19	Bék	2 256	2 248	2 250	2 239	2 240	2 252	2 253	2 253	2 267	2 266	2 265	2 270	2 269	2 278
20	Bék	1 465	1 465	1 465	1 465	1 465	1 465	1 468	1 468	1 470	1 470	1 469	1 516	1 537	1 543
21	Cs	1 419	1 420	1 421	1 444	1 469	1 453	1 454	1 454	1 482	1 484	1 485	1 490	1 490	1 490

**Adatok munkalap**

	A	B	C	D	E	F	G
1	<b>legnagyobb fejlesztés az előző évhez képest</b>	vasút			közút		
2		év	területi egység	hossz [km]	év	területi egység	hossz [km]
3							
4							
5	<b>teljes változás 2007-től 2021-ig</b>	vasút			közút		
6		változás iránya		hossz [km]	változás iránya		hossz [km]
7							
8							
9	<b>legfejlettebb megyénk 2012-ben</b>	vasúti viszonylatban			közúti viszonylatban		
10							
11							
12	<b>Magyarország területe [km<sup>2</sup>]</b>						
13							
14	<b>megyék, amelyek nem fejlődtek 2007. és 2021. között</b>						
15							
16							
17							
18							

**Valasz munkalap**



diagram

Segédszámításokat az **Adatok** munkalap 22. sorától lefelé vagy az AG oszloptól jobbra található cellákban végezhetünk. A megoldásban saját függvény vagy makró nem használható.

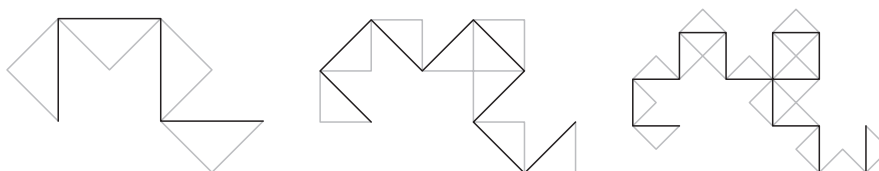
Beküldendő egy tömörített **i578.zip** állományban a megoldást tartalmazó munkafüzet és a megoldás rövid leírását bemutató dokumentáció.

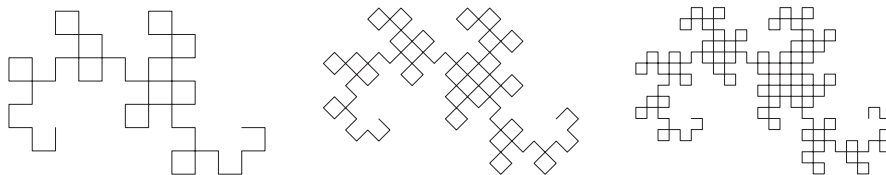
**I. 579.** A Technógrafika eredetileg a Logo programozási nyelvhez készült, de ma már több oktatási célú fejlesztő környezetnek és nyelvnek része, például elérhető Scratchben vagy Pythonban. Egy magyar fejlesztés keretében a Technógrafikát alkalmazhatjuk a JavaScript nyelv segítségével. Az Agent JS programozási felület, valamint a dokumentáció és néhány példa található a

<https://vintaai.github.io/agent/>

címen.

Különösen izgalmas területe a Technógrafikának a rekurzív ábrák és fraktálok rajzolása. Készítsük el a Koch-görbe, a sárkánygörbe, valamint a Sierpiński-háromszög ábráját az Agent JS segítségével úgy, hogy a rekurzió szintje a felhasznál-





náló által beállítható legyen. A három fraktálról a KöMaL 2021. májusi számában is olvashattunk, innen származik a sárkánygörbe készítését szemléltető *ábra*.

Beküldendő egy tömörített `i579.zip` állományban a három ábrát létrehozó oldal forráskódja.

**I/S. 67.** Adott egy  $N \times M$ -es téglalap, melyet le szeretnénk helyezni a síkra úgy, hogy az egyik csúcsa az origóba essen. Adjuk meg, hogy hányféleképpen tehetjük ezt meg úgy, hogy a másik három csúcsa is rácspontra, azaz egész koordinátákkal rendelkező pontra essen.

A bemenet egyetlen sorában az  $N$  és  $M$  számok, a téglalap oldalhosszai találhatóak szóközzel elválasztva.

A kimenet egyetlen sorában egy szám szerepeljen: hányféleképpen lehet helyezni a téglalapot a síkra úgy, hogy az egyik csúcsa az origóba, a többi csúcsa rácspontra essen.

*Példa:*

Bemenet	Kimenet
2 2	4
5 10	24

*Korlátok:*  $1 \leq N, M \leq 10^6$ . Időlimit: 0,4 mp.

*Értékelés:* a pontok 50%-a kapható, ha a program helyes kimenetet ad az  $1 \leq N \leq 100$  esetekre.

Beküldendő egy `is67.zip` tömörített állományban a megfelelően dokumentált és kommentezett forrásprogram, amely tartalmazza a megoldás lépéseit, valamint megadja, hogy a program melyik fejlesztői környezetben futtatható. A dokumentáció tartalmazza a megoldás elméleti háttérét, az esetleg felhasznált forrásokat. Ne tartalmazzon kódrészleteket, azok magyarázata kódkommentek formájában a forrásprogramban szerepeljen.

**S. 166.** Marika néni palacsintázójában egy ember süti és tölti a palacsintákat. Eddig, ha érkezett egy rendelés, akkor megsütötte a megfelelő számú palacsintát, majd beletöltötte a kért tölteléket, és odaadta a pénztárosnak, aki a fizetést elhanyagolható időn belül elintézte. Az utóbbi időben azonban nagyon népszerű lett a hely, ezért néha nagyon sokat kell várni. Hamar kiderült, hogy a sütés és palacsintatöltés közötti váltás sok időt vesz el. Azt szeretnénk kideríteni, hogy ez valóban probléma-e.

Írjunk programot, amely kiszámolja, mennyi a várakozási idők maximuma, ha minden rendelést külön-külön szolgálunk ki, illetve mennyi a várakozási idők maximumának lehető legkisebb értéke, ha pont jókor váltunk a sütés és töltés között.

Ezen a helyen nem szeretik a pazarlást, így csak akkor állnak neki egy palacsinta megsütésének, ha már van rá leadott rendelés. Tehát olyan nem fordulhat elő, hogy több megsütött palacsintánk van, mint a kiszolgáltatlan rendelések összege.

A bemenet első sorában egy palacsinta megsütésének ideje  $S$ , egy palacsinta töltésének ideje  $T$ , és a kettő közötti váltás ideje  $V$  szerepel. A második sorban a rendelések  $R$  száma van. A következő  $R$  sor mindegyikében két egész szám szerepel: az adott rendelés leadásának időpontja és a kért palacsinták száma.

A kimenet első sorába írjuk ki, mennyi a várakozási idők maximuma, ha a rendeléseket egyesével szolgáljuk ki, azaz mindig csak annyi palacsintát sütünk, amennyi a következő rendeléshez kell. A kimenet második sorába pedig azt, hogy mennyi a várakozási idők maximumának lehető legkisebb értéke.

*Minta:*

Bemenet (a / jel sortörést helyettesít)	Kimenet (a / jel sortörést helyettesít)
2 1 1 / 2 / 0 5 / 2 1	20 / 19

*Magyarázat* (zárójelben a befejezési idő szerepel): az első esetben megsütünk 5-öt (10), majd megtöltjük (16), majd sütünk még egyet (18), majd azt is megtöltjük (20). A várakozási idő csökkenthető, ha előbb megsütünk 5-öt (10), majd mivel van rá rendelés, még egyet (12), majd megtöltjük az első ötöt (18), majd még egyet (19).

*Korlátok:*  $1 \leq S, T, V \leq 10^4$ ,  $R \leq 10^5$ . A rendelés ideje és a rendelések számának összege is legfeljebb  $10^5$ . Időlimit: 1 mp.

*Értékelés:* a pontok 20%-a kapható, ha a program által adott kimenet első sora helyes. További 40% kapható, ha a program helyes kimenetet ad az  $R \leq 10$  és a palacsinták számának összege legfeljebb 20 esetekben.

Beküldendő egy `s166.zip` tömörített állományban a megfelelően dokumentált és kommentezett forrásprogram, amely tartalmazza a megoldás lépéseit, valamint megadja, hogy a program melyik fejlesztői környezetben futtatható. A dokumentáció tartalmazza a megoldás elméleti háttérét, az esetleg felhasznált forrásokat. Ne tartalmazzon kódrészleteket, azok magyarázata kódkommentek formájában a forrásprogramban szerepeljen.

✱

**A feladatok megoldásai regisztráció után a következő címen tölthetők fel:**

<https://www.komal.hu/munkafuzet>

**Beküldési határidő: 2023. január 15.**

✱