

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

INFORMATIKA

EMELT SZINTŰ GYAKORLATI VIZSGA

minden vizsgázó számára

2023. május 22. 8:00

Időtartam: 240 perc

Beadott dokumentumok	
Piszkozati pótlapok száma	
Beadott fájlok száma	

A beadott fájlok neve

OKTATÁSI HIVATAL

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Forrás:

A feladatlap bázisszövegei az eredeti forrásszövegek módosításával (rövidítésével, nyelvtani egyszerűsítésével), adatainak felhasználásával, de az eredeti szöveg, adatok integritásának megtartása mellett jöttek létre. Az eredeti szövegek, adatok, képek forrása:

1. Grace Hopper

https://hu.wikipedia.org/wiki/Grace_Hopper Utolsó letöltés 2019. december 11.

[https://hu.wikipedia.org/wiki/Grace_Hopper#/media/Fájl:Commodore_Grace_M._Hopper,_USN_\(covered\)_head_and_shoulders_crop.jpg](https://hu.wikipedia.org/wiki/Grace_Hopper#/media/Fájl:Commodore_Grace_M._Hopper,_USN_(covered)_head_and_shoulders_crop.jpg) Utolsó letöltés 2019. december 11.

https://ichef.bbci.co.uk/news/624/cpsprodpb/5546/production/_93903812_2211e3a7-dedf-423e-b107-5704518cdc29.jpg Utolsó letöltés 2019. december 11.

https://en.wikipedia.org/wiki/Naval_Reserve_Medal Utolsó letöltés 2019. december 11.

https://en.wikipedia.org/wiki/Presidential_Medal_of_Freedom Utolsó letöltés 2019. december 11.

https://en.wikipedia.org/wiki/Armed_Forces_Reserve_Medal Utolsó letöltés 2019. december 11.

https://en.wikipedia.org/wiki/National_Defense_Service_Medal Utolsó letöltés 2019. december 11.

[https://en.wikipedia.org/wiki/World_War_II_Victory_Medal_\(United_States\)](https://en.wikipedia.org/wiki/World_War_II_Victory_Medal_(United_States)) Utolsó letöltés 2019. december 11.

<https://pcforum.hu/szotar/compiler> Utolsó letöltés 2019. december 11.

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Fontos tudnivalók

A vizsgán **használható eszközök**: a vizsgázó számára kijelölt számítógép, papír, toll, ceruza, vonalzó, lepecsételt jegyzetlap.

A feladatlap belső oldalain és a jegyzetlapon készíthet **jegyzeteket**, ezeket a vizsga végén be kell adni, de tartalmukat nem fogják értékelni.

A feladatokat **tetszőleges sorrendben megoldhatja**.

Felhívjuk a figyelmet a **gyakori** (10 percenkénti) **mentésre**, és feltétlenül javasoljuk a mentést minden esetben, mielőtt egy másik feladatba kezd.

Vizsgadolgozatát a feladatlapon található **azonosítóval megegyező** nevű **vizsgakönyvtárba** kell mentenie! Ellenőrizze, hogy a feladatlapon található kóddal megegyező nevű könyvtár elérhető-e, ha nem, még a vizsga elején jelezze a felügyelő tanárnak!

Munkáit a **vizsgakönyvtárba** **mentse**, és a vizsga végén **ellenőrizze**, hogy minden megoldás a megadott könyvtárban van-e, mert csak ezek értékelésére van lehetőség! Ellenőrizze, hogy a beadandó állományok olvashatók-e, mert a nem megnyitható állományok értékelése nem lehetséges!

Amennyiben az adatbázis-kezelés feladatát LibreOffice Base alkalmazásban oldja meg, a táblamódosító lekérdezéseket leíró SQL-parancsokat vagy a LibreOffice Base adatbázis-állomány részeként vagy pedig egy külön szövegállományban kell beadnia. Szövegfájl beadása esetén a szövegfájl neve egyértelműen utaljon a tartalmára (például *SQL-parancsok.txt*), valamint az állományban a parancs mellett szerepeltesse az előírt lekérdezésnevet!

MySQL adatbázis-motor használata esetén az adatbázis adatait is le kell menteni egy úgynevezett „**dump**” fájlba.

A beadott program csak abban az esetben értékelhető, ha a vizsgázó létrehozta a választott programozási környezetnek megfelelő forrásállomány(oka)t a vizsgakönyvtárban, és az tartalmazza a részfeladatok megoldásához tartozó forráskódot.

A **forrásfájlokat** a vizsgakönyvtárban találja.

Javasoljuk, hogy a feladatokat először **olvassa végig**, utána egyenként oldja meg az egyes részfeladatokat!

Amennyiben számítógépével **műszaki probléma** van, jelezze a felügyelő tanárnak! A jelzés ténye és a megállapított hiba jegyzőkönyvezésre kerül. A kiesett idővel a vizsga ideje hosszabb lesz. Amennyiben a hiba mégsem számítógépes eredetű, a javító tanár értékeléskor köteles figyelembe venni a jegyzőkönyv esetleírását. (A rendszergazda nem segítheti a vizsgázót a dolgozat elkészítésében.)

A vizsga végén a feladatlap első oldalán Önnek fel kell tüntetnie a **vizsgakönyvtárban és állományokban található, Ön által előállított és beadott fájlok számát, illetve azok nevét**. A vizsga végeztével addig ne távozzon, amíg ezt meg nem tette, és a felügyelő tanárnak ezt be nem mutatta!

Kérjük, jelölje be, hogy mely operációs rendszeren dolgozik, és melyik programozási környezetet használja!

Operációs rendszer: ☐ Windows ☐ Linux

Programozási környezet:

- | | | |
|----------------------------------|------------------------------|-------------------------------------|
| <input type="radio"/> FreePascal | <input type="radio"/> GCC | <input type="radio"/> Visual Studio |
| <input type="radio"/> Lazarus | <input type="radio"/> Perl 5 | <input type="radio"/> _____ |
| <input type="radio"/> JAVA SE | <input type="radio"/> Python | <input type="radio"/> _____ |

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

1. Grace Hopper

Az informatika fejlődéstörténetének úttörői között számos nő szerepelt. Ebben a feladatban a „Nők az informatikában” sorozat kezdő cikkét kell Grace Hopperről elkészítenie az alábbi leírás és minta alapján. Az elkészítéséhez használja fel az *eletrajz.txt* UTF-8 kódolású szövegállományt és a *civil.png*, *katonai.jpg*, *kituntetese.png* és *Mark1.jpg* képeket!

1. Hozza létre szövegszerkesztő program segítségével az *amazing_grace* nevű dokumentumot a program alapértelmezett formátumában a források felhasználásával! A dokumentumban ne legyenek felesleges szóközök és üres bekezdések! A dokumentumban alkalmazzon automatikus elválasztást!
2. Legyen a dokumentum álló tájolású és A4-es lapméretű! Minden margót állítson 2,2 cm-re!
3. A teljes szövegben alkalmazzon Times New Roman (Nimbus Roman) betűtípust!
4. A szövegtörzs formázásánál a következő beállításokat végezze el:
 - a. a betűméret 11 pontos legyen;
 - b. a bekezdések sorkizárt igazításúak legyenek, egyszeres sorközzel, ha a feladat mást nem kér;
 - c. a bekezdések előtt 0 és után 6 pontos térköz legyen!
5. A főcím 18 pontos az alcímek 16 pontos betűméretűek és mindkettő esetén kiskapitális, félkövér betűstílusúak legyenek! A címek előtt és után 12 pontos térköz legyen!
6. A cím utáni második bekezdésben lévő idézetet alakítsa dőlt betűstílusúvá!
7. Az első alcím fölötti hat bekezdést alakítsa felsorolássá, a minta szerinti felsorolásjellel! A felsoroláson belül ne legyen térköz! Amennyiben a szövegszerkesztőjében a minta szerinti felsorolási jel nem elérhető, akkor rombusz alakú (◆) felsorolás jelet használjon!
8. Készítsen lábjegyzetet a „*compiler*” szóhoz! A lábjegyzet szövegét a forrásban, a „*compiler*” szó után {} jelek között találja. A szövegtörzsből törölje a lábjegyzet szövegét és a {} jeleket!
9. Szúrja be a *civil.png* képet a minta szerinti helyre és igazítsa a jobb margóhoz! A kép méretét az arányok megtartásával módosítsa! A szöveg kövesse a kép körvonalát a minta szerint!
10. A „*Fiatalkora*” alcímtől kezdődően a minta szerinti törzsszöveg bekezdéseinek első sorát 1 cm-rel húzza beljebb!
11. A „*Karrierje*” alcím utáni bekezdésbe illessze be a bal margóhoz igazítva a *Mark1.jpg* képet és méretezze az oldalarányok megtartásával 4,7 cm magasságúra! A szöveg távolsága a képtől a jobb oldalon 0,3 cm legyen!
12. Készítse el a képaláírás szövegét! A képhez tartozó szöveget a bekezdés végén {} jelek között találja. A szövegtörzsből törölje a képaláírás szövegét és a {} jeleket! A szöveget formázza 10 pontos betűmérettel és dőlt stílussal és igazítsa középre a képhez képest!
13. Az „*1949-ben*” kezdetű bekezdésben az idézet szövegét alakítsa dőlt betűstílusúvá!

- 30 pont**

Minta:

GRACE MURRAY HOPPER

1906. december 9. – 1992. január 1.

„Az emberek allergiásak a változásokra. Szeretik azt mondani: »Mindig így csináltuk.« Megpróbálok harcolni ezzel. Ezért van egy óra a falamon, ami az óramutató járásával ellentétes irányban halad.”

- ❖ Ellentengernagy, az Amerikai Egyesült Államok Haditengerészietének tisztje.
- ❖ Matematikus, a számítógép-tudomány egyik úttörője.
- ❖ Egyike volt a Harvard Mark I számítógép első programozóinak.
- ❖ Ő írta az első fordítóprogramot (compiler).
- ❖ Felvázolta a számítógéptől független programnyelv ötletét, ami a COBOL, az egyik első modern programozási nyelvi megalkotásához vezetett.
- ❖ Neki tulajdonítják a „debugging” kifejezés elterjesztését.

FIATALKORA

Grace Hopper NY-i Yorkban született 1906-ban Grace Brewster Murray néven. A csipek közül ő volt a legidősebb. Kiváncsi gyerek volt és ezt a tulajdonságát egész életében nagyon elhatározta, hogy kiderít az ébrenlétéről működését. Hét éves korától szétvette anyja rádióját volna mi a terve. Ezek után már csak egy óráig lehetett. A középiskolát a Yale (Hartridge School) végezte Plainfieldben (New Jersey államban). 16 éves korában a nem meglehetősen híres Yale-i Egyetemre került. A Yale-i Egyetemre került. A Yale-i Egyetemre került. 1928-ban BSc (Bachelor) fokozatot szerzett Matematika és fizika szakon. Eredményeinek nyert a Yale Beta Kappa egyesületi tagjai közé is. 1930-ban megkezdte MSC (Master) fokozat a Phi Epsilon. 1934-ben PhD fokozatot szerzett matematikából. Már 1931-ben elkezdte tanítani a volt iskolájában, a Vassar College-ban, ahol 1941-ben docensnek nevezték ki. 1945-ben a Vassar College-ban tanított. 1946-ban Foster Hopperhez, 1948-ban elvált. 1948-ban férjhez, de volt férj nem tartotta.

KARRIERJE

Hopper 1943-ban távolléti engedélyt kapott a Vassar Főiskolától és belépett az Államok Haditengerészetének tartalékosai közé. Sok nőhöz hasonlóan önkéntesként szolgált a "Women Accepted for Volunteer Emergency Service" - hivatalos nevén: U.S. Naval Reserve.



Grace Hopper Howard Aikennel és a MARK-I számítógép csapatával (1944).

¹ A szoftverek az ember számára könnyen kezelhető forráskódjából a célplatformon közvetlenül futtatható készítő program.

Harvard számítógép laborban maradt. 1949-ben professzorként visszatért a Vassar Főiskolára, de a Haditengerészet megbízott munkatársaként tovább folytatta a munkát a Harvard Egyetemen.

1949-ben Eckert-Mauchly Computer Corporation vállalat alkalmazója lett mint matematikus és csatlakozott az UNIVAC I számítógép fejlesztő csoportjához. Az 1950-es évek elején a cég beolvadt a Remington Rand vállalatba. 1952-ben sikerült elkészítenie az első fordítóprogramot is, ez volt az A-0 fordítóprogram. „Senki nem hitt benne. Volt egy működő fordítóprogramom és senkit sem érdekelt. Azt mondták, hogy a számítógép csak aritmetikai műveletek elvégzésére képes.” - mesélte Hopper.

1954-ben Hoppert a vállalat programozási igazgatójává nevezték ki. Részele fejlesztette az első fordítóprogramra épülő programozási nyelveket például a MATH-MATIC és a FLOW-MATIC programnyelvet is. Ez lett később az alapja a COBOL programozási nyelvnek. Ezért is hívják Hoppert a COBOL anyjának.

1967 és 1977 között Hopper volt az igazgatója a Haditengerészet Informatikai Tervező Hivatalához (Navy's Office of Information Systems Planning) tartozó Programozási Nyelv (Navy Programming Languages Group) csoportjának. 1973-ban kapitánnyá léptették elő.

Hopper 1966-ban, a nyugalmazási szabályoknak megfelelően, 60 éves korában vonult nyugdíjállományba a Haditengerészet Flottatartálékoltí fregattkapitány rangban. 1967-ben visszahívták a szolgálatba egy hat hónapos időszakra, amit határozatlan időre meghosszabbítottak. 1971-ben ismét nyugdíjba vonult, de 1972-ben ismét felkérték a szolgálatra. 1973-ban Elmo R. Zumwalt, Jr. admirális kapitányává nevezték ki.



KITÜNTETÉSEI



Defense Distinguished Service Medal
 Legion of Merit
 Meritorious Service Medal
 American Campaign Medal
 World War II Victory Medal
 National Defense Service Medal
 Armed Forces Reserve Medal with two Hourglass Devices
 Naval Reserve Medal
 Presidential Medal of Freedom (posthumous)

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

2. Levelezőverseny

Egy osztály tanulóinak egy része háromfordulós egyéni levelezős versenyen vett részt. A verseny eredményeit értékeljük ki táblázatkezelő-programmal.

A három forduló és az összesítés adatait az *1fordulo.txt*, *2fordulo.txt*, *3fordulo.txt* és *osszesites.txt* táblázattal tagolt, UTF-8 kódolású állományok tartalmazzák. Mind a négy fájlban azonos nevek szerepelnek, névsorba rendezve. Ahol a fordulók adataiban nincs a versenyzőnek pontszáma, az azt jelenti, hogy abban a fordulóban nem adott be megoldást.

A megoldás során vegye figyelembe a következőket!

- Amennyiben lehetséges, a megoldás során képletet, függvényt, hivatkozást használjon.
- Segédszámításokat a J oszloptól jobbra végezhet.
- A részfeladatok között van olyan, amely egy korábbi kérdés eredményét használja fel. Ha a korábbi részfeladatot nem sikerült teljesen megoldania, használja a megoldását úgy, ahogy van, vagy írjon be egy valószínűnek tűnő eredményt, és azzal dolgozzon tovább! Így ugyanis pontokat kaphat erre a részfeladatra is.

1. Az *1fordulo.txt*, *2fordulo.txt*, *3fordulo.txt* és *osszesites.txt* forrásfájlok tartalmát helyezze el a táblázatkezelő fájlnevekkel azonos nevű munkalapján (**1fordulo**, **2fordulo**, **3fordulo** és **osszesites**) az A1-es cellától kezdődően! Munkáját *levelezoverseny* néven mentse el a táblázatkezelő alapértelmezett formátumában!
2. „**Wincs Eszter**” tanuló neve az 1. forduló adataiban jól szerepel, de a többi munkalapon téves. A nevek egységes kezelésére gondolva, oldja meg, hogy a 2., a 3. forduló és az összesítés munkalapja az 1. forduló munkalapjáról hivatkozással vegye át a neveket!
3. Mindhárom forduló munkalapján az F1-es cellában határozza meg az elérhető maximális pontszámot, majd az F oszlop megfelelő celláiban a versenyzők pontszámainak összegét az adott fordulóban.

A további feladatokat az „**osszesites**” munkalapon oldja meg!

4. A B1:D1 tartomány celláiban a fordulók maximálisan elérhető pontszámait és a B3:D18 tartomány celláiban a versenyzők egyes fordulóban elért összesített pontszámait jelenítse meg a megfelelő munkalapok celláira hivatkozva! Az értékeket úgy jelenítse meg, hogy ha az eredeti munkalapon változtatás történik, akkor az „**osszesites**” munkalapon automatikusan helyes érték jelenjen meg!
5. A B19:D19 tartomány celláiban számítsa ki a fordulóban elért eredmények átlagát úgy, hogy csak a 0-nál nagyobb eredményeket vegye figyelembe az adott fordulóban! (Feltételezheti, hogy fordulónként legalább egy 0-nál nagyobb érték van.) Az átlagok két tizedesjeggyel jelenjenek meg!
6. Az E1-es cellában számítsa ki, a fordulók maximális pontszámainak összegét! Az E3:E18 tartomány celláiban adja meg a versenyzők összesített pontszámát!
7. A verseny értékelésében nem vesznek részt azok a versenyzők, akik valamelyik fordulóban 0 pontot értek el. A nem értékelt versenyzők teljes adatsora a minta szerint kapjon automatikusan piros háttérszínt! Az adatok változása esetén is helyes legyen a piros háttérszín!

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

8. Az *A23:B26* tartomány celláiban ponttartományok határait látja. A *C23:C26* tartomány celláiban adja meg, hogy az egyes tartományokban hány fő található az összesített pontszámok alapján! Olyan képletet készítsen, ami a határok változtatása után is helyes eredményt ad!
9. A *B29:B31* tartomány celláiban adja meg az első három helyezett pontszámát, majd a *C29:C31* tartomány celláiban a pontszámokhoz tartozó neveket!
10. Az *A1:E19* tartományban a szegélyezést állítsa be a minta szerint! A *C23:C26* tartomány celláiban állítsa be a minta szerinti egyéni formátumot! Az *A28:C31* tartomány celláiban 14 pontos félkövér karakterek legyenek és a cellák igazítása rendre: jobbra, középre, balra. Az oszlopokat állítsa olyan szélesre, hogy az *A1:E19* tartomány celláinak tartalma olvasható legyen!

15 pont

Minta:

	A	B	C	D	E
1	max. pontszám	40	40	40	120
2	Név	1. forduló	2. forduló	3. forduló	összesítés
3	Arany Áron	38	21,5	31	90,5
4	Asztalos Amália	31	23,5	29	83,5
5	Balog Barna	36,5	17,5	0	54
6	Esztergom Andrea	40	39	38	117
7	Esztergom Eszter	38	38,5	39	115,5
8	Farkas Ferenc	29	31,5	32	92,5
9	Havasi Hedvig	37	0	28	65
10	Kala Pál	37	39,5	38	114,5
11	Major Anna	39,5	34	37,5	111
12	Mezei Virág	37,5	19	25	81,5
13	Mustár Márta	38,5	35,5	27	101
14	Pécsi Péter	0	0	0	0
15	Reszet Elek	29	40	38	107
16	Ultra Viola	35	39	39	113
17	Wincs Eszter	13	18,5	0	31,5
18	Zöld Piroska	40	28	31	99
19	átlag	34,60	30,36	33,27	
20					
21					
22					
23	100,5	120	7 fő		
24	80,5	100	5 fő		
25	60,5	80	1 fő		
26	0	60	3 fő		
27					
28	helyezés	pont	név		
29	1	117	Esztergom Andrea		
30	2	115,5	Esztergom Eszter		
31	3	114,5	Kala Pál		

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

3. Album

Az arcfelismerő alkalmazások elterjedésével a fotóalbumok is jól rendszerezhetők. Egy családi digitális képgyűjtemény és a család tagjainak néhány adata áll rendelkezésre a *fenykep.txt*, a *szemely.txt* és a *kapcsolok.txt* állományban.

- Készítsen új adatbázist *album* néven! A mellékelt állományokat importálja az adatbázisba a fájlnevvvel azonos táblanéven (***fenykep***, ***kapcsolok***, ***szemely***)! Az állományok tabulátorral tagolt, UTF-8 kódolású szövegfájlok, az első soruk a mezőneveket tartalmazza. A létrehozás során állítsa be a megfelelő típusokat és az elsődleges kulcsokat!

Táblák:

fenykep (*id*, *evszam*, *szines*, *meret_x*, *meret_y*)

<i>id</i>	A digitális kép azonosítója (szám), ez a kulcs
<i>evszam</i>	A fotó készítési éve (szám)
<i>szines</i>	A kép színes-e (logikai). Fekete-fehér képnél hamis és színesnél igaz.
<i>meret_x</i>	A kép szélessége képpontban (szám)
<i>meret_y</i>	A kép magassága képpontban (szám)

kapcsolok (*fenyid*, *szemid*)

<i>fenyid</i>	A fotó azonosítója (szám), az összetett kulcs része
<i>szemid</i>	A képen felismert személy azonosítója (szám), az összetett kulcs része

szemely (*id*, *nev*, *szulev*)

<i>id</i>	A személy azonosítója (szám), ez a kulcs
<i>nev</i>	A személy neve (szöveg), családtag esetén egyedi keresztnév, különben nem kitöltött
<i>szulev</i>	A személy születési éve (szám), családtagon kívüli személy esetén nem kitöltött



A következő feladatok megoldásánál a lekérdezéseket és a jelentést a zárójelben olvasható néven mentse! Ügyeljen arra, hogy a lekérdezésekben pontosan a kívánt mezők szerepeljenek! Felesleges mezőt és rekordot ne jelenítsen meg!

- A színes képek a legelterjedtebbek a fotózásban, de művészi képeket ma is fekete-fehérben készítenek sokszor. Készítsen lekérdezést, amely évszám szerinti csökkenő sorrendben megjeleníti a fekete-fehér képek évszámát, szélességét és magasságát! (***2feketefehér***)
- Készítsen lekérdezést, amely a „***Vince***” nevű családtag fotóit sorolja fel! A listában a fénykép azonosítója, Vince életkora a fotón, és a kép mérete, azaz a kép szélességének és magasságának szorzata jelenjen meg! (***3vince***)
- Lekérdezés segítségével sorolja fel azoknak a fényképeknek az azonosítóját és készítés évét, amelyen szerepel olyan családtag, aki a fénykép készítésének évében született! A listában többször szerepelhet az a kép, amelyen több ilyen családtag van! (***4baba***)

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

- Határozza meg lekérdezés segítségével annak a fotónak az azonosítóját és készítési évét, amelyen a legtöbb ember látható! Ha több ugyanannyit tartalmazó van, akkor mindet megjelenítheti. **(5sokan)**
- A családtagok egyéni fotói mindig érdekesek. Szeretnénk listázni azokat a képeket, amelyeken valamelyik családtag egyedül szerepel. Egészítse ki a jelzett helyen az alábbi lekérdezést úgy, hogy az ilyen képek azonosítója és a képen szereplő családtag neve jelenjen meg! A kiegészített lekérdezést mentse! **(6egyeni)**

```
SELECT nev, fenyid
FROM személy, kapcsoló
WHERE id = szemiid AND
```

...

A fenti lekérdezés szövege a források között a *6alap.sql* fájlban megtalálható.

- Matyi Anna unokája. Készítsen lekérdezést, amely felsorolja azokat a fotókat, amelyen mind a ketten szerepelnek! A listában a fénykép azonosítója, és a kép készítési ideje jelenjen meg! **(7kozos)**
- Készítsen jelentést, amely évente kilistázza, hogy melyik családtag hány fotón szerepel! A jelentés létrehozását lekérdezéssel készítse elő! A jelentés elkészítésekor a mintából a mezők sorrendjét, a címet és a mezőnevek megjelenítését vegye figyelembe! A jelentés formázásában a mintától eltérhet. **(8szereples)**

A családtagok képeinek száma évente		
Év	Név	Darab
1999	Anna	4
	Botond	5
	Rebeka	5
	Vilmos	3
	Zsombor	1
2000	Anna	4
	Bence	2

30 pont

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

4. Szállítószalag

Egy speciális, kör alakú üzemcsarnokban különböző anyagok szállításához körpályán mozgó szállítószalagot állítottak üzembe. Erre a szalagra bárhol feltehetnek bármilyen anyagot egy rekeszbe zárva, annak beállításával, hogy hova szánják. A célhelyen a szalagról automatikusan lekerül a rekesz. A szalag az óramutató járásával egyező irányban mozog, akkor indul el, ha rekesz kerül rá és addig mozog, amíg van rajta szállítandó rekesz. Az ilyen szállítási feladatokat kell feldolgoznia.

Rendelkezésre áll a *szallit.txt* nevű adatfájl, amelynek első sorában két egész szám található, egymástól egy szóközzel elválasztva. Az első a szállítószalag hossza, a második a szállítószalag egy egységnyi elmozdulásához szükséges idő. A fájl további legfeljebb 1000 sorában soronként négy egész szám található, egymástól szóközzel elválasztva. Az első szám megadja, hogy mikor tették a szalagra a rekeszt. A következő kettő megmutatja, hogy honnan hova tart a szállítás (a megtett út a szalag hosszánál mindig kevesebb). Az utolsó pedig a szállítandó tömeget jelenti. A hosszúságot és a pozíciókat távolságegységben, az időt időegységben, a tömeget pedig tömegegységben adták meg. Az adatfájl egyetlen adata sem nagyobb 500-nál.

Például:

```
200 3
1 134 64 34
14 22 129 83
14 135 54 21
23 31 40 61
...
```

A példában a fájl első 5 sora látható. Az első sor mutatja, hogy 200 egység hosszú a szalag, és egy egységnyi távolságot 3 időegység alatt tesz meg. A 2. sor tartalmazza az első rekesz adatait. A 3. sor szerint a 14. időegységben a 22-es hosszúságegységnél felhelyeznek egy rekeszt, amely a 129-es hosszúságegységnél kerül le. A rekesz 83 egység tömegű. A rekesz tehát $129 - 22 = 107$ egység utat tesz meg, és a $14 + 107 \cdot 3 = 335$ -ös időegységben ér célba. A következő sorban szereplő rekesz szintén a 14. időegységben kerül a szalagra, a 135-ös hosszúságegységtől az 54-es hosszúságegységig halad. A megtett távolság ezért a 0 hosszúságegységig 200-135 egység, azt követően pedig 54 egység, azaz összesen 119 egységnyi távolságot tesz meg a 21 egység tömegű rekesz.

Készítsen programot, amely az állomány adatait felhasználva az alábbi kérdésekre válaszol! A program forráskódját mentse *szalag* néven! A program megírásakor a felhasználó által megadott adatok helyességét, érvényességét nem kell ellenőriznie, és feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek.

A képernyőre írást igénylő részfeladatok esetén – a mintához tartalmában hasonlóan – írja ki a képernyőre a feladat sorszámát (például: 2. feladat), és utaljon a kiírt tartalomra is! Ha a felhasználótól kér be adatot, jelenítse meg a képernyőn, hogy milyen értéket vár! Mindkét esetben az ékezetmentes kiírás is elfogadott.

1. Olvassa be a *szallit.txt* állomány adatait, és annak felhasználásával oldja meg a következő feladatokat!
2. Kérje be egy szállítás sorszámát, majd írassa ki annak indulási és célhelyét! (A szállításokat 1-től sorszámozzuk.)

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

3. Készítsen függvényt `tav` néven, amely megadja a szállítás távolságát a szalag hosszának, valamint az indulási és a célhelynek ismeretében! A függvényt használja fel a későbbi feladatok megoldása során. A függvényfejet az alábbiaknak megfelelően készítse el, megoldásában az ott szereplő változóneveket használja!

Függvény `tav(szalaghossz, indulashelye, erkezeshelye : egész szám) : egész szám`

4. Határozza meg, hogy a rendelkezésre álló szállítások során mekkora volt a legnagyobb szállítási távolság! Írja a képernyőre a maximális távolságot és az összes ilyen hosszúságú szállítás sorszámát!
5. Adja meg, mekkora tömeg haladt el összesen a 0 pozíciójú hely előtt! Az onnan induló vagy oda érkező rekeszeket ne vegye figyelembe!
6. Kérjen be egy időpontot, és határozza meg az adott időpontban szállított rekeszek sorszámát! Az éppen akkor induló rekeszeket vegye figyelembe, de a célba érőket ne! Ha nem volt szállított rekesz, akkor a rekeszek sorszáma helyett az „üres” szót írja ki!
7. Hozza létre a `tomeg.txt` fájlt, amely megadja, hogy az egyes helyekről összességében mennyi tömeget szállítottak el! Azok a helyek ne jelenjenek meg a fájlban, ahonnan nem történt szállítás! (A fájlba írt adatok sorrendje tetszőleges.)

Példa a szöveges kimenetek kialakításához:

```
2. feladat
Adja meg, melyik adatsorra kíváncsi! 3
Honnan: 135 Hova: 54

4. feladat
A legnagyobb távolság: 195
A maximális távolságú szállítások sorszáma: 31 33

5. feladat
A kezdőpont előtt elhaladó rekeszek össztömege: 957

6. feladat
Adja meg a kívánt időpontot! 300
A szállított rekeszek halmaza: 1 2 3 6 7 10 11
```

Példa az `tomeg.txt` szöveges állomány tartalmára:

```
5 37
12 35
16 26
17 33
18 41
22 83
...
```

45 pont

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

	pontszám	
	maximális	elért
Szövegszerkesztés, prezentáció, grafika, weblapkészítés 1. Grace Hopper	30	
Táblázatkezelés 2. Levelezőverseny	15	
Adatbázis-kezelés 3. Album	30	
Algoritmizálás, adatmodellezés 4. Szállítószalag	45	
A gyakorlati vizsgarész pontszáma	120	

dátum

javító tanár

	pontszáma egész számra kerekítve	
	elért	programba beírt
Szövegszerkesztés, prezentáció, grafika, weblapkészítés		
Táblázatkezelés		
Adatbázis-kezelés		
Algoritmizálás, adatmodellezés		

dátum

dátum

javító tanár

jegyző