a) Mekkora gyorsulással kéne mozgatni az edényt, hogy egyik falára 0,01 Pa túlnyomás hasson?

b) Mekkora elektromos feszültség keletkezik ezalatt az edény szembenfekvő két oldallapja között, ha az edény fémből készült? Az elektron tömege $m_e = 9{,}1\ 10^{-19}$ kg.

XI. osztály

F.G. 105. Képzeljük el, hogy Kolozsvárt az Északi-sarkkal egy egyenes csatorna köti össze, amiben légpárnás vonat közlekedik. A légellenállástól és a magas hőmérséklettől eltekintünk. A Föld sugara $R \approx 6400$ km, a csatorna hossza jó közelítéssel 45°-os ívet köt össze, és ismert, hogy a vonatra mindenkor csak a tőle befele található tömegek vonzása hat, a kifele található gömbhéj összhatása nulla. Számítsuk ki:

a) Ha a szerelvény — teljesen súrlódásmentesen — magára hagyva (a motor bekapcsolása nélkül) Kolozsvárról elindul, mekkora maximális sebességet érne el?

b) Mennyi idő alatt jutna az Északi-sarkra?

c) Írjuk fel a mozgás egyenleteit!

F.G. 106. Egy áramköri szakasz az $R = 10$ kΩ és $L = 50/\pi$ mH, valamint az $R = 10$ kΩ és a $C = 500/\pi$ μF csoportokat, amelyeknek elemei egymással párhuzamosan vannak összekapcsolva, egymás után sorban tartalmazza, sarkaira pedig zéró és végtelen között változtatható frekvenciájú, $U = 220$ V feszültségű áramforrás van rákapcsolva.

a) Számítsuk ki a ν_0 rezonanciafrekvenciát.

b) Ábrázoljuk az áramerősséget, valamint a tekercs és a kondenzátor sarkain a feszültséget a frekvencia függvényében!

c) Rezonancia esetén mekkora a két feszültség közötti fáziseltérés?

Informatika

Nemes Tihamér Számítástechnikai Verseny, 2. forduló, 1995.

IX–X. osztály

I. 59. A Kísérleti Fanemesítő Intézet újfajta fenyőfákat nemesített ki. A fenyőfa törzséből pontosan 2 ág ágazik el, vagy egyetlenegy sem. Az egyes ágak ugyanolyan hosszúak és vastagok, mint a törzs, s a végükből legfeljebb újabb 2-2 ág ágazik el, vagy egy sem. Ezek megint ugyanolyan hosszúak, mint a törzs. Egy fát zárójelekkel és F betükkel írunk le a számítógép számára: *(baloldali ág) F (jobboldali ág)* formában. A fának törzse biztosan van.

Példa:

```
ágnélküli fa:
F

kétágú fa:
(F) F (F)

bonyolultabb fa:
( (F) F (F) ) F (F)
```

Írj programot, amely meghatározza

A. a fa magasságát (a leghosszabb út hosszát a gyökértől valamelyik ág végéig) - a fenti három példában ez 1, 2 illetve 3,

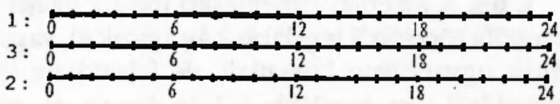
B. a fa tömegét (feltételezve, hogy a törzs, illetve a vele azonos tömegű ágdarabok egységnyi tömegűek) - a fenti példában ez rendre 0, 1, illetve 2.

I. 60. Egy városban több televízióadó műsorát lehet fogni. Egy szöveges állományban tároljuk, hogy melyik mikor ad (feltesszük, hogy az adásidők a hét minden napján ugyanakkor vannak, a következő napra nem nyúlnak át, s egész órától egész óráig tartanak), egyes adók naponta többször is sugározhatnak műsort.

Az állomány minden sorában három szám található, egymástól egy - egy szóközzel elválasztva; az első az adó sorszáma, a második az adás kezdete, a harmadik pedig a vége (balról zárt, jobbról nyílt intervallumként). Az órák száma 0 és 24 közötti egész. Az állomány üres is lehet.

Példa:

```
1  18  22
1  6   10
3  16  20
2  12  20
```



Írj programot, amely

A. megadja a leghosszabb olyan időszakot egy napon belül , amikor az állományban tárolt adatok szerint egyetlen TV-adás sem fogható a városban (a fenti példában: 0–6)!

B. meghatározza, hogy a nap melyi kegyórás időszakában lehet a legtöbb műsor közül választani, s megadja ezek számát (a fenti példában: 18–19 vagy 19–20 a jó időszak, s ekkor 3 adást lehet nézni)!

I. 61. Egy nagyvárosban 3 (földalatti) metróvonal található, s mindegyiken sok-sok állomás. A három vonalnak vagy egyetlen közös állomása, vagy pedig az 1.-2-nak és a 2.-3 -nak külön átszállási helye van. Egy külföldi turista áll az egyik metróállomáson, s egy másik metróállomásra akar eljutni. Készíts programot, amely beolvassa e két állomás nevét, majd megmondja, hogy a turistának az induló állomásról, milyen irányba (melyik végállomás felé) hány megállót kell utaznia, s ha át kell szállnia, akkor ezt az átszállás előtti, illetve utáni metróvonalra is megadja. A létező metróállomások nevét megtalálhatod a METRO.DAT állományban. (Az állományban soronként 1 adat szerepel, először az 1. vonal állomásainak száma, majd egyesével az állomások neve, utána a 2. vonal állomásainak száma...) Az átszállóhely(ek) a közös név alapján ismerhető(k) fel.

I. 62. Környezetünkben biológiai felmérést végeztünk, ún. táplálkozási párokat azonosítottunk.(mi eszik mit?). A növények nem esznek semmilyen élőlényt, az állatok pedig vagy növényeket, vagy más állatokat esznek. A BIO.INP állományban soronként egy-egy táplálkozási párt nevezünk meg, ahol a pár jelentése: az elsőnek megadott eszi a másodiknak megadottat, pl."róka eszi fogoly", "csiga eszi fű". A két nevet egyetlen szóköz választja el. A BIO.INP állomány üres is lehet.

Készíts programot, amely kiválasztja a (csak) növényevő állatokat! Figyelem: ami nem eszik semmit, az növény.

Példa:

Bemenet:	Eredmény:
róka fogoly	csiga
róka feketerigó	földigiliszta
fogoly földigiliszta	
csiga fű	
feketerigó csiga	
földigiliszta avar	
feketerigó gabonamag	

XI–XII. osztály

I. 63. A Kísérleti Fanemesítő Intézet újfajta fenyőfákat nemesített ki. A fenyőfa törzséből legalább 2 ág ágazik el, vagy egyetlenegy sem. Az egyes ágak ugyanolyan hosszúak, de feleakkora tömegűek, mint a törzs, s a végükből újra legalább 2-2 ág ágazik el, vagy egy sem. Ezek megint ugyanolyan hosszúak, mint amiből kinőttek, de feleakkora tömegűek.

Egy fát zárójelekkel és F betűkkel írunk le a számítógép számára: *F (első ág) (második ág)...(n.ág) formában.* A fának törzse biztosan van.

Példa:

```
ágnélküli fa:
F
```


```
kétágú fa:
F (F) (F) (F) (F)
```

```
sokágú fa:
F (F) (F) (F) (F)
```

```
bonyolultabb fa:
F (F (F) (F) (F)) (F)
```

Írj programot, amely meghatározza
A. a fa magasságát (a leghosszabb út hosszát a gyökértől valamelyik ág végéig) - a fenti négy példában ez 1,2,2 illetve 3,
B. a fa tömegét (feltételezve, hogy a törzs egységnyi tömegű) a fenti példában ez rendre 1,2,3, illetve 2.75,
C. a közös elágazásból induló ágak számának maximumát - a fenti példában ez rendre 0, 2, 4, illetve 3.

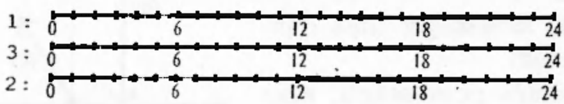
I. 64. Egy városban N (>1) napon át több televizióadó műsorát lehet fogni. Egy szöveges állományban tároljuk, hogy melyiken mikor van adás. Egyes adók bármikor (akár többször is) sugározhatnak műsort, az adásidő egyik napról a másikra is átnyúlhat, sőt akár N napon át, megállás nélkül is tarthat.

Az állomány minden sorában hét szám található egymástól egy-egy szóközzel elválasztva, az első az adó sorszáma, a következő három az adás kezdete (napsorszám, óra, perc), az utolsó három pedig a vége (ugyanilyen jelentéssel) - az adásidő balról zárt, jobbról nyílt intervallumot jelent.

Az órák száma 0 és 24, a percek száma 0 és 59 közötti egész. N értéke az állományban található napsorszámok alapján határozható meg. Az állomány üres is lehet.

Példa: (egy napon belüli, egész órakor kezdődő és végződő adásokkal)

```
1  1  18  0  1  22  0
1  1   6  0  1  10  0
3  1  16  0  1  20  0
2  1  12  0  1  20  0
```



Írj programot, amely
A. megadja a leghosszabb olyan időszakot, amikor az állományban tárolt adatok szerint egyetlen TV-adás sem fogható a városban (a fenti példában:

```
1. nap, 0.00-6.00),
```

B. meghatározza, hogy ay N-edik nap melyik percében lehet a legtöbb műsor közül választani, s akkor hány közül lehet (a fenti példában: 1. nap 18.00 és 19.59 között bármelyik perc jó, ekkor 3 adás fogható)

I. 65. Készíts programot, amely a billentyűzetről tetszőleges sorrendben beolvassa egy konvex sokszög csúcsainak egész koordinátáit, majd kiírja őket olyan sorrendben, ahogyan a sokszög oldalai mentén bejárhatjuk őket az óramutató járásával ellentétes irányban! Kiindulópontnak a sokszög legkisebb x-koordinátájú csúcsát vedd (ha több ilyen van, akkor közülük a legkisebb y-koordinátájút). A koordináták biztosan helyesek, nem kell ellenőrizni őket.

A koordinátarendszer a szokásos, a csúcsok koordinátáját az (x,y) egész számpár adja meg, ahol x az abszcissza és y az ordináta. Az origó a (0,0) koordinátájú pont, x jobbra, y fölfelé nő.

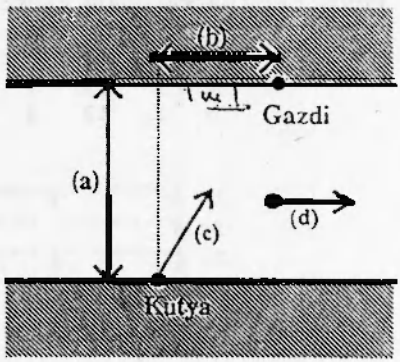
Példa:

```
bemenô számsorozat: 2 -2 -2 4 -2 -3 1 2
értelmezése: (2,-2), (-2,4), (-2,-3), (1,2)
az eredmény: (-2, -3), (2,-2), (2,-2), (1,2), (-2,4)
```

I. 66. Egy kutya úgy úszik át a folyón a túlparton álló gazdájához, hogy minden pillanatban a gazdi irányába igyekszik. Ezt a mozgást kell közelítő módszerrel modelleznéd. A program számítsa ki, hogy a gazdájától milyen távolságra ér partot a kutya, és ez mennyi ideig tart! Ehhez a következő, valós értékű paramétereket kell beolvasnia a programnak a billetyüzetről:

a. A folyó szélességét méterben

b. A gazda távolságát méterben a kutya kezdőpontjának vetületétől a túlparton (pozitív, ha a folyásiránnyal azonos irányban van, negatív az ellenkező esetben).



c. A kutya sebességét (m/s, végig ugyanaz).

d. A folyó sebességét (m/s mindenütt ugyanaz).

e. A közelítés pontosságát, azaz annak az időintervallumnak a hosszát másodpercekben, amelyen belül a program egyenes vonalú mozgással számolhat.

Grafikus ábrázolás nem szükséges, az értékelésnél nem vesszük figyelembe, a programod kipróbálását azonban segítheti.

A folyó két partját párhuzamos egyeneseknek tekintjük. A modellezés akkor álljon le, amikor a kutya már egy méternél közelebb kerül a túlsó parthoz.

A kutya mozgását haladási iránya, saját sebessége, valamint a folyó sebessége határozza meg.Mint tudjuk, mindkét sebesség állandó.A haladási irány, illetve a kutya sebességének a haladási iránytól függő x és y irányú összetevője azonban csak egy-egy időintervallumon belül tekinthető állandónak.

A kutya haladási irányát az alábbi képlettel számíthatjuk ki:

$$\text{Irány} = \text{ArcTan}\left(\frac{\text{GazdiYKoordináta} - \text{KutyaYKoordináta}}{\text{GazdiXKoordináta} - \text{KutyaXKoordináta}}\right)$$

(Arc Tan: arkusz tangens függvény, megadja, hogy adott tangens érték mekkora szöghöz tartozik, $-\frac{\Pi}{2} \leq ArcTan(x) \leq \frac{\Pi}{2}$)

Egy időintervallum alatt a kutya x irányban

(Kutyasebesség * Cos (Irány) + Vízsebesség) * időintervallum hossza, y irányban pedig

KutyaSebesség * Sin(Irány) * Időintervallum hossza

utat tesz meg, hiszen a koordinátarendszert úgy célszerű megválasztani, hogy a víz az x-tengely mentén folyon.

Példa:

```
paraméterek: (a) : 200, (b): 100, (c): 5, (d): 6, (e):
eredmények: az eltelt idô: 157, a partot érés távolsága a gazditól kb.206
```

(a távolság a közelítés miatt valós szám lesz, itt egy közelítő egész számot adunk meg).

I. 66. Van egy gépünk, amely egy 40 jel hosszúságú *szalagból* és egy iró-olvasó *fejből* áll. Jel egy maximum 40 elemű halmaz, az ábécé egy-egy eleme lehet. A gép maximum 30 különböző állapotban lehet.

A gép egy -egy jelet olvas a szalagról (onnan, ahol a fej van). A géphez tartozó szabálytáblázat irja elő,hogy a beolvasott jeltől és a gép pillanatnyi állapotától függően mit kell csinálni előbb a szalaggal, majd a fejjel, illetve mi lesz a gép következő állapota. Az elvégzendő művelet az ábécé egy elemének a szalagra írása, a fej jobbra, illetve balra mozgatása vagy helyben hagyása lehet.

Van a gépnek egy speciális állapota, a *végállapot.* Ha a gép ebbe az állapotba kerül, akkor az olvasott jeltől függetlenül leáll. Ha a szabálytáblázatban nincs a gép aktuális állapotára és az olvasott jelre vonatkozó utasítás, akkor a gép automatikusan a végállapotba kerül. A gép indulásakor az, író-olvasó fej a szalag 20. pozícióján áll (a sorszámozást az 1. pozíciótól kezdjük) és az 1-es sorszámú állapotban van.

Írj programot a fent leírt gép szimulálására!

A GEP.INP állomány első sora a szalagon levő jeleket tartalmazza (tehát pontosan 40 karakter hosszú). Az állomány további soraiban a szabálytáblázat elemei vannak minden sorban egy szabály. A szabályok megadásának sorrendje tetszőleges. Egy (*állapot, jel*) párhoz csak egyetlen egy szabály tartozhat. (Ennek ellenőrzése nem szükséges)

A szabályok formátuma: (a1, j1) :: (a2, j2, *),

ahol *a1* az aktuális, *a2* pedig az új állapot,

j1 az aktuális állapotban olvasott, míg *j2* a szalagon a helyébe írandó jel, a * pedig *B*,*J*, vagy — lehet. A *B* azt jelenti, hogy balra, a *J* azt, hogy jobbra kell mozgatni a fejet, míg — esetén nincs mozgatás.

Az állapotokat sorszámukkal jelöljük (a sorszámozás 1-tőkl kezdődik), a végállapot sorszáma 0.

A. Ha a gép működése nem fejeződik be 1000 lépésen belül, akkor a GEP.OUT első sorába a VÉGTELEN szót írjuk.

B. Ha a gép működése közben a fej elhagyja a szalagot, az első sorba értelemszerűen a HIBA, BALRA KILÉPETT vagy a HIBA, JOBBRA KILÉPETT üzenetet írjuk.

C. Ha a gép aktuális állapotára és az olvasott jelre a szabálytáblázatban nincs utasítás, akkor az első sorba a NEMDEFINIALT szót, a második sorba pedig az aktuális állapot sorszámát, egy szóközt és az olvasott jelet írjuk.

D. Ha a gép működése hibátlanul befejeződik, akkor a file első sorába a VEGES szót, a következő sorba pedig a szalag tartalmát írjuk.

Megoldott feladat

Informatika

I.36. feladat, 1993-94/5-6. szám

A számegyenesen N számpárral N szakaszt határozunk meg, amely lefedi a számegyenes megfelelő részeit. Irjunk algoritmust, amely megadja, mekkora részt takar a lefedés a számegyenesen.

Példa: (-2,5), (7,9), (8,10). Eredmény: 10

Megoldás:

A feladatot általánosabban oldjuk meg: az adatokat egy bemeneti szövegállomány tartalmazza, minden sorában egy számpárt (a két szám között legalább egy szóközzel). Ezeket a számpárokat egymás után olvassuk, s egy láncolt listában megőrizzük az addig kapott intervallumokat, amelyek együttesen megadják a kívánt lefedést. Az adatokat helyeseknek tekintjük, tehát nem ellenőrizzük.

```
program nn;       { FIRKA 1993-94/5-6.szám, I.36. fel.}

type parok = ^elem;            { intervallumok láncolt listája }
     elem  = record
               a,b : real;
               kov : parok;
             end;

var f    : text;   {bemeneti állomány, soronként egy számpárral}
```