



Un nou mileniu... cu PROBLEME

Ionuț Stan

Am trăit cu toții, mari și mici, profesori și elevi, bucuria intrării într-un nou mileniu. Visăm la o lume în care să nu avem probleme... Evident nu vorbesc despre "problemele de informatică", de care, cu siguranță, nu vor să se despartă cei pasionați. În cele ce urmează, vă vom prezenta rezolvările a două probleme propuse de către dl. prof. Sorin Groza, apoi propunem spre rezolvare câteva probleme, la care așteptăm rezolvări cât mai reușite.

Serviciul de gardă

În vederea stabilirii serviciului de gardă în noaptea dintre milenii la o unitate militară, comandantul nominalizează n soldați (S) și n gradați (G). Comandantul, știind că în post intră odată două persoane și că ofițerul responsabil îi pune în pereche pe cei prevăzuți în listă, primul cu ultimul, al doilea cu penultimul etc., stabilește ordinea în listă astfel încât să nu se formeze pereche dintr-un soldat și un gradat. (De ce? Băieții vor înțelege - când vor face armata.) De altfel, comandantul mai știe că n este număr par și va folosi această proprietate în momentul în care va întocmi lista.

Scrieți un program care, plecând de la numele și gradele (soldat sau gradat, adică S sau G) celor care vor intra în gardă, afișează pe ecran lista comandantului din care ofițerul va forma perechi, conform cerinței. De asemenea, programul trebuie să afișeze și perechile ce intră în serviciu.

Exemplu

Intrare

$n=4$

Chirila, Macarie, Coza, Domsa, Lador,
Orlea, Hila, Pacurar
S, S, G, G, S, G, S, G

Ieșire (o soluție posibilă):

lista: Chirila, Coza, Lador, Pacurar,
Domsa, Macarie, Orlea, Hila
serviciul:
Chirila - Hila
Coza - Orlea
Lador - Macarie
Pacurar - Domsa

Rezolvare

Numele ostașilor se vor păstra într-un șir de articole numit *ostasi*, fiecare articol având două câmpuri: *nume* și *grad*. Acest șir se va ordona astfel încât în prima parte să apară gradații, apoi să apară soldații. Pentru a crea lista conform cerinței problemei, aceasta se afișează astfel încât să ajungă în pereche fie doi soldați, fie doi gradați. Problema a fost descompusă în subprobleme conform cărora în program s-au implementat subprogramele *Citire_date*, *Ordonare*, *Listare_unul_cate_unul* și *Listare_pe_perechi* ale căror funcție rezultă clar din denumirile lor.

Listing Garda.PAS

```
Program Garda;  
uses Crt;  
type ostas=record  
    nume,grad:string[30]  
end;  
    tipostasi=array[1..100] of ostas;  
var n,m:Byte;  
    ostasi:tipostasi;  
  
procedure Citire_date(var ostasi:  
    tipostasi;var n:Byte);  
var i:Byte;  
begin  
    repeat  
        Write('Cati soldati/gradati intra ',  
            'in serviciu? ');  
  
        Readln(n)  
    until n mod 2=0;
```



```

m:=n*2;      { numarul total al celor }
              { care vor efectua garda }
Writeln('Introduceti datele');
for i:=1 to m do
with ostasi[i] do
begin
    Write('Dati numele ostasului ',i,
          ':');

    Readln(num);
    Write('Dati gradul ostasului ',i,
          ' (G sau S): ');

    Readln(grad);
    Writeln('-----')
end
end;

procedure Ordonare(var ost:tipostasi;
                   m:Byte);

var gata:Boolean;
    i:Byte;
    aux:ostas;
begin
    repeat { ordonare cu metoda bulelor }
        gata:=true;
        for i:=1 to m-1 do
            { ordonam sirul de ostasi }
            if ost[i].grad<ost[i+1].grad then
            begin { interschimbam doi ostasi }
                aux:=ost[i];    { atribuire la }
                                { nivel de articol }
                ost[i]:=ost[i+1]; ost[i+1]:=aux;
                gata:=false
            end
        until gata
    end;

procedure Listare_unul_cate_unul
                   (ostasi:tipostasi);
    { afisare lista cu ostasi }

var i:Byte;
begin
    Write('lista: ');
    for i:=1 to n div 2 do
        Write(ostasi[i].nume,' ',
              ostasi[n+i].nume,' ');

    for i:=n div 2 + 1 to n do
        Write(ostasi[n+i].nume,' ',
              ostasi[i].nume,' ');

    Writeln
end;

procedure Listare_pe_perechi
                   (ostasi:tipostasi);
    { afisare lista cu perechi }

var i:Byte;

```

```

begin
    for i:=1 to n div 2 do
    begin
        Writeln(ostasi[i].nume,'-',
                ostasi[n-i+1].nume);
        Writeln(ostasi[i+n].nume,'-',
                ostasi[m-i+1].nume)

    end
end;

Begin
    Clrscr;
    Citire_date(ostasi,n);
    Ordonare(ostasi,m);
    Listare_unul_cate_unul(ostasi);
    Listare_pe_perechi(ostasi);
    Readkey
End.

```

Plasarea oglinzilor

Se consideră un șir de n elemente care au valorile 0 și 1. Definim operația plasare a două oglinzi, una la stânga elementului de pe poziția i , iar alta la dreapta elementului de pe poziția j . Această operație are ca efect interschimbarea elementelor egal depărtate de extreme a elementelor cuprinse între pozițiile i și j , inclusiv.

Scrieți un program care va citi două șiruri de lungime n și va stabili pozițiile în care vor trebui plasate cele două oglinzi în cadrul primului șir, astfel încât în urma plasării oglinzilor - conform definiției - primul șir să devină identic cu al doilea șir.

Exemplu

Intrare

```

n = 8
0 0 1 1 0 1 1 0
0 1 0 1 1 0 1 0

```

Ieșire

Oglinzile trebuie puse în pozițiile 2 și 6

Rezolvare

Pentru a evita toate încercările de a plasa cele două oglinzi, se determină porțiunea maximă de elemente consecutive pe care cele două șiruri nu coincid (nucleul). Astfel se caută cea mai din stânga pereche de elemente ce nu coincid și apoi cea mai din dreapta pereche. Pozițiile corespunzătoare acestor elemente (diferite două câte două) se păstrează în variabilele a și b . În continuare se verifică dacă oglinzile au efectul așteptat. În acest scop se interschimbă primul element cu ultimul, al doilea cu penultimul etc. din nucleu și se copiază peste vechiul nucleu în oricare dintre cele două șiruri. În urma acestei operații avem un șir1 nou care se compară element cu element cu șir2 (neschimbat). Dacă se constată că acum (deci în urma oglin-

dirii) cele două șiruri coincid, se vor afișa pozițiile celor două oglinzi, în caz contrar se va afișa un mesaj corespunzător.

Listing OGLINZI.PAS

```
Program oglinzi
uses Crt;
const mesaj1='Sirurile sunt identice!';
      mesaj2='Nu exista solutie!';
type tip_sir=array[1..100] of Integer;
var nr,      { numar elemente in siruri }
    a, { prima pozitie din primul sir }
    { unde cele doua siruri nu au }
    { elemente identice }
    b, { ultima pozitie din al doilea }
    { sir unde cele doua siruri nu }
    { au elemente identice }
    k,      { indice in nucleu }
    nr_identice, { numarul elemente- }
    { lor identice din cele }
    { doua siruri }
    aux,i:Integer;
    nucleu,      { portiunea unuia din }
    { siruri care trebuie oglindita }
    sir1,sir2:tip_sir;
```

Begin

```
Clrscr;
{ citire date }
Write('Numarul elementelor = ');
Readln(nr);
WriteLn('Introduceti primul sir');
for i:=1 to nr do Read(sir1[i]);
WriteLn('Introduceti al doilea sir');
for i:=1 to nr do Read(sir2[i]);
{ verificarea identitatii sirurilor }
nr_identice:=0;
for i:=1 to nr do
    if sir1[i]=sir2[i] then
        Inc(nr_identice);
if nr_identice=nr then
    Write(mesaj1);
    { determinare pozitii oglinzi }
if nr_identice<nr then
begin
    for i:=nr downto 1 do
        if sir1[i]<>sir2[i] then a:=i;
    for i:=1 to nr do
        if sir1[i]<>sir2[i] then b:=i;
        { extragere nucleu }
    for i:=a to b do
begin
    Inc(k);
    nucleu[k]:=sir1[i]
end;
```

```
{ efectuare oglindire in nucleu }
for i:=1 to k div 2 do
begin
    aux:=nucleu[i];
    nucleu[i]:=nucleu[k-i+1];
    nucleu[k-i+1]:=aux
end;
{ copierea nucleului oglindit }
for i:=a to b do
    sir1[i]:=nucleu[i-a+1];
    { verificare efect oglindire }
nr_identice:=0;
for i:=1 to nr do
    if sir1[i]=sir2[i] then
        Inc(nr_identice);
if nr_identice=nr then
    Write('Oglinzile trebuie puse ',
        'in pozitiile ',a,' si ',b)
    else Write(mesaj2)
end;
Readkey
End.
```

În continuare vă propunem spre rezolvare câteva probleme simple:

Învățarea alfabetului

Copiii din clasele mici învață din greu alfabetul pe când cei mai mari trag chiulul. În clasa I un copil a "păcăuit" pentru că a încercat să vadă cum este dacă nu își învață "lecția" pentru ora viitoare. Învățătoarea l-a pedepsit punându-l să scrie n litere mari, printre care acesta trebuie să scrie cel puțin odată toate literele mari pe care le-a învățat până atunci.

Citind numărul de litere n care trebuie să le scrie copilul și literele pe care acestea le-a scris, să se scrie un program care verifică dacă în șirul de litere apar sau nu anumite litere de mai multe ori. Se vor afișa literele care apar de mai multe ori (număr apariții > 1) și numărul aparițiilor acestora.

Exemplu

Intrare

5

A B C A C

Ieșire

A 2

C 2

Numere prime... cu divizori

Elevii clasei a V-a au învățat la matematică despre numerele prime că au exact doi divizori: 1 și numărul însuși. La informatică au învățat că numerele se pot reprezenta și în alte baze, nu doar în baza 10. Cei doi profesori s-au gândit să le dea o temă copiilor de clasa a V-a, care să combine ce-



begin



le două informații învățate. Ei vor trebui să găsească toate numerele mai mici decât un număr dat n , care au proprietatea că scrise în baza 10 nu sunt prime, dar scrise în altă bază și citite în baza 10, sunt prime.

Scrieți un program care găsește aceste numere, precum și bazele în care au această proprietate.

Exemplu

Intrare

$n = 5$

Ieșire

numar in baza 10: 4 nu este prim

numar 31 in baza 3 (in baza 10 este prim)

Ghiociei și viorele

Copiii din clasa a VI-a au mers la pădure după ghiociei. Ei au cules ghiociei și viorele, în total un număr de n flori de primăvară. Au observat că ghiociei și viorele erau găsite după o anumită regulă: între două viorele erau ghiociei câte unul, apoi câte doi și tot așa mai departe.

Citind un număr natural n care să reprezinte numărul florilor și inițialele lor (G pentru ghiociei, V pentru viorele) să se spună dacă florile se găsesc sau nu într-o configurație care îndeplinește cerința de mai sus. Afișați un mesaj (DA sau NU), precum și pozițiile viorelelor.

Exemple

Intrare

$n = 6$

V G V G G V

Ieșire

DA

1 3 6

Intrare

$n = 5$

G V G G V

Ieșire

NU

Serviciul pe clasă

La începutul anului școlar dirigintele a întocmit planificarea serviciului în clasă, câte doi elevi fiind planificați pentru fiecare zi. Deoarece atunci când au fost întrebați, elevii au declarat că doresc să fie de serviciu câte o dată și un

băiat, dirigintele a stabilit următoarea regulă: prima fată din catalog va sta cu primul băiat, a doua cu al doilea și așa mai departe. Dirigintele știe că are n elevi în clasă, dar nu a numărat câte fete și câți băieți sunt, așa că se poate întâmpla să constate la sfârșit că unii copii au rămas fără pereche.

Citindu-se numărul de elevi, numele și genul lor (F pentru fete, B pentru băieți), să se spună dacă au rămas elevi fără pereche și care sunt aceștia.

Exemplu

Intrare

$n=5$

Ana f

Ioana f

Andrei b

Anca f

Paul b

Ieșire

Fara pereche: Anca

Tăierea copacilor

Pe drumul național E81 sunt copaci numerotați în zig-zag (pe partea dreaptă a drumului copacii sunt numerotați cu numere pare și pe stânga cu numere impare). La un moment dat au fost aleși copaci pentru tăiere de pe fiecare parte a drumului după următoarea regulă: se va tăia primul acel copac care este numerotat cu un număr ale cărui cifre însumate formează un pătrat perfect, apoi următorul copac având aceeași proprietate și tot așa mai departe până la capătul drumului.

Citindu-se numărul copacilor, să se determine câți copaci au fost aleși pentru tăiere, pe care parte a drumului sunt aceștia, precum și numărul cu care au fost numerotați. Numerotarea copacilor începe de la 1.

Exemplu

Intrare

$n = 20$

Ieșire

6 copaci

DE PE DREAPTA: 4 10 18

DE PE STANGA: 1 9 13

Ionuț Stan este elev la Școala Generală Nr. 1, "Ovidiu Hulea" din Aiud.

Poate fi contactat prin e-mail la adresa: aliennt@yahoo.com.

Probleme cu Diablo II Online...

Serverele care găzduiesc varianta online a jocului Diablo II au căzut victime atacurilor unor hackeri care au profitat de o slăbiciune a sistemului de securitate. Hackerii au preluat

controlul asupra personajelor diferiților jucători, deseori eliminându-le din joc. Blizzard a înlocuit aceste personaje cu cele salvate pe data de 19 decembrie 2000.