

## Prințesa

### Soluție

Citirea rudelor o vom face în vectorul rude. Pentru a găsi pretendenții la mâna prințesei construim numere prime (începând cu 2) numere pe care le introducem în vectorul pretendent dacă nu sunt rude. Apoi afișăm vectorul pretendent în ordine inversă. Programul Pascal care rezolvă problema este următorul:

```
var rude:array[1..100]of integer;
    pretendent:array[1..100]of integer;
    aux,h,k,n,p,i,j,g,L:integer;prim,gasit:boolean;
begin
write('k=');readln(k);
write('h=');readln(h);
for i:=1 to h do readln(rude[i]);
p:=2;
L:=0;
while L<k do begin
    prim:=true;
    for j:=2 to p div 2 do
        if p mod j=0 then prim:=false;
    if prim then begin
        gasit:=false;
        for g:=1 to h do
            if p=rude[g] then gasit:=true;
        if not gasit then begin
            L:=L+1;
            pretendent[L]:=p;
            end;
        end;
    p:=p+1;
    end;
for i:=1 to L div 2 do begin
aux:=pretendent[i];
pretendent[i]:=pretendent[L-i+1];
pretendent[L-i+1]:=aux;
end;
for i:=1 to L do
write(pretendent[i],' ');
end.
```

## Produs

**100 puncte**

### Soluție

Problema poate fi rezolvată prin două metode: folosind un vector care reține cifrele numărului de telefon sau fără a folosi vectori (în acest caz se construiesc repetat numerele din expresie prin prelucrarea cifrelor din numărul de telefon).

Pentru varianta cu vector, după construirea vectorului cu cifrele din numărul de telefon, se consideră pe rând numerele formate cu prima cifră și numărul cu celelalte cifre, apoi cel cu primele doua cifre și celelalte, etc. și se formează expresiile corespunzătoare. Dintre aceste expresii se reține cea cu valoarea maximă.

Pentru exemplul din enunț se formează expresiile:

$$2*301=602$$

$$23*1=23$$

$$230*1=230$$

și se afișează prima pentru că are valoarea maximă.

Programul Pascal care rezolvă problema este următorul:

```
var x:array[1..20]of byte;
    a,e1,e2,e1max,e2max:longint;
    k,i,j,aux:byte;
begin
  write('Numar de telefon: ');
  readln(a);
  k:=0;
  while a<>0 do
    begin
      k:=k+1;
      x[k]:=a mod 10;
      a:=a div 10;
    end;
  for i:=1 to k div 2 do
    begin
      aux:=x[i];
      x[i]:=x[k-i+1];
      x[k-i+1]:=aux;
    end;
  e1max:=0;
  e2max:=0;
  for i:=1 to k-1 do
    begin
      e1:=0;
      for j:=1 to i do
        e1:=e1*10+x[j];
      e2:=0;
      for j:=i+1 to k do
        e2:=e2*10+x[j];
      if e1*e2>=e1max*e2max then
        begin
          e1max:=e1;
          e2max:=e2;
        end;
    end;
  writeln(e1max,'*',e2max,'=',e1max*
e2max);
end.
```

**Timp maxim de execuție:** 1 secundă/test

**Doar o notă să-ți mai pun...**  
**Soluție**

**100 puncte**

Problema poate fi rezolvată folosind doi vectori: primul vector reține notele obținute de elevi și se construiește cel de-al doilea vector care păstrează frecvența notelor obținute în lucrare. Prin parcurgerea celui de-al doilea vector se vor determina nota (notele) care apare de cele mai multe ori și notele care nu s-au acordat la lucrare.

```
#include<stdio.h>
#include<math.h>
main()
{int n,i,z,max,v[25],w[10];
//printf(numarul elevilor: ");
scanf("%d", &n);
for(i=1;i<=10;i++) w[i]=0;
//printf("notele obtinute de elevi la lucrarea de control:");
for(i=1; i<=n;i++)
{scanf("%d", &v[i]);
w[v[i]]+=1;
}
max=w[1];
for(i=1;i<=10;i++)
if (w[i]>max) max=w[i];
for(i=1; i<=10;i++)
if(max==w[i]) printf("%d ",i);
printf("\n");
for(i=1;i<=10;i++)
if(!w[i]) printf("%d ",i);
printf("\n");
}
```