

DESCRIERE SOLUȚII SECȚIUNEA 7-8 ÎNCEPĂTORI

PROBLEMA 1

autor prof. Constantin Scheau

BARA

Se vor elimina pe rând piesele aflate la distanțe minime de găurile din bară. Pentru fiecare piesa se va calcula distanța față de gaura cea mai apropiată și se vor contoriza deplasările.

PROBLEMA 2

propunator prof. Luminita Nastase

CLĂTITE

Parcurgem toate numerele dintre N și M, iar pentru fiecare număr calculăm simultan valorile pentru primele 3 tipuri de clătite astfel: descompunem în factori primi numărul (mergând până la radical din acesta) și ne folosim de următoarele formule/ proprietăți:

Tipul 1:

Dacă nu am găsit niciun divizor al numărului, acesta e *prim* și îl contorizăm.

Tipul 2:

Un număr este *pătrat perfect* dacă toți exponenții (d_1, d_2, \dots, d_k) sunt divizibili cu 2.

Un număr este *cub perfect* dacă toți exponenții (d_1, d_2, \dots, d_k) sunt divizibili cu 3.

Folosind aceste două relații putem determina pe parcurs dacă numărul este sau nu pătrat respectiv cub perfect.

Tipul 3:

Calculăm pe parcurs suma divizorilor numărului cu următoarea formulă: fie numărul N și descompunerea sa în factori primi, de forma: $p_1^{d1} * p_2^{d2} * \dots * p_k^{dk}$.

Suma divizorilor numărului N este dată de relația: $\frac{p_1^{d1+1}-1}{p_1-1} * \frac{p_2^{d2+1}-1}{p_2-1} * \dots * \frac{p_k^{dk+1}-1}{p_k-1}$.

Astfel, verificăm dacă suma calculată – numărul propriu-zis = numărul propriu-zis (suma de mai sus include și numărul).

Dacă niciuna dintre relațiile de mai sus nu este îndeplinită verificăm pur și simplu paritatea numărului (tipul 4 dacă e *par* respectiv tipul 5 dacă e *impar*).

Complexitatea totală: $O(N \sqrt{N})$

PROBLEMA 3

autor prof. Radu Visinescu

INTERVALE

Descriere soluție: Se sortează crescător acele elemente distincte. Apoi vom folosi două cicluri while, incluse unul în altul. În cadrul primului ciclu, vom determina pozițiile de început ale secvențelor de numere consecutive. În ciclul interior parcurgem, una câte una, aceste secvențe. La sfârșitul unei astfel de parcurgeri, dacă secvența este formată din cel puțin două numere, se va tipări și ultimul element al secvenței. Programul este următorul:

```
{sortarea vectorului X}
i:=1;
while i <= n do
  begin
    write('[',x[i]);
    L:=0;
    while x[i]+1=x[i+1] then
      begin
        i:=i+1;
        L:=L+1
      end;
    if L >0 then write ('.',x[i]);
    write(']');
    i:=i+1
  end;
```