Propunător: Narcisa Daniela Ștefănescu
Inspectoratul Școlar Județean Suceava

Săptămâna 20:

Calculul multiplicității unui număr prim în n!

Conținuturi:

- Să calculăm n!
- Descompunerea unui număr în factori primi
- Folosirea formulelor lui Euler și Legendre

În matematică factorialul unui număr întreg n este notat cu n! și este egal cu produsul numerelor naturale mai mici sau egale cu n.

Exemple:

5! = 1*2*3*4*5

Factorialul unui număr oarecare n indică numărul de permutări (numărul de posibilități de rearanjare) ale unei mulțimi finite având n elemente. Algoritmul de calcul pe care îl prezintă manualul de informatică este:

```
#include <iostream>
using namespace std;
int main()
{
    int n,i,p;
    cout<<"n:"; cin>>n;
    p=1;
    for (i=2; i<=n; i++)
        p=p*i;
    cout<<p;
    return 0;
}
```

Dar dacă vom calcula factorialul pentru valori mici a lui n, constatăm că valoarea este foarte mare. Nu vom putea să calculăm n!, decât pentru valori foarte mici ale lui n.

10 ! = 3628800

15! = 1307674368000

42! = 1405006117752879898543142606244511569936384000000000

În concursuri au apărut cerințe legate de calcularea unor divizori ai lui n!. Cum procedăm? Trebui să învățăm anumite formule de la matematică.

Dacă dorim să calculăm la ce putere apare un număr prim în descompunerea în factori primi a lui n!, vom folosi o formulă. Matematicianul ***Adrien-Marie Legendre*** a descoperit că multiplicitatea (exponentul) unui număr prim *p* care apare în descompunerea în factori primi a lui *n*! poate fi exprimată exact ca:

$$\boldsymbol{exponentul\ lui\ p\ \hat{i}n\ n!} = \left[\frac{n}{p^1}\right] + \left[\frac{n}{p^2}\right] + \left[\frac{n}{p^3}\right] + \cdots.$$

Pe scurt putem scrie $\boldsymbol{exp_{n!}(p)} = \left[\frac{\boldsymbol{n}}{\boldsymbol{p^1}}\right] + \left[\frac{\boldsymbol{n}}{\boldsymbol{p^2}}\right] + \left[\frac{\boldsymbol{n}}{\boldsymbol{p^3}}\right] + \cdots.$

Aceasta este o formulă matematică, notațiile sunt folosite la matematică.

De exemplu pentru n =1000 și p =2

$exp_{n!}(p)=[1000/2] + [1000/2^2]+ [1000/2^3]+ [1000/2^4]+ [1000/2^5]+\ldots.$

Ne vom opri când $[1000/2^k]$ devine 0.

$exp_{n!}(p)=[1000/2] + [1000/4] + [1000/8] + [1000/16] + [1000/32] + [1000/64] + [1000/128] + [1000/256] + [1000/512] = 500 + 250 + 125 + 62 + 31 + 15 + 7 + 3 +1 =994$

Formula II

Dacă p este prim, iar α este un număr natural, atunci:

$$\boldsymbol{exp_{n!}(p^{\propto})} = \left[\frac{\boldsymbol{exp_{n!}(p)}}{\propto}\right]$$

Ex: $exp_{50!}(3) = [50/3] + [50/9] + [50/27] + [50/81] = 16 +5+1+0 ==22$

$exp_{50!}(3^7) = [22/7] = 3$

Formula III – Calculul multiplicității unui număr oarecare în n!

Dacă a este un număr natural oarecare, el se poate descompune în factori primi:

a = $p1^{\propto 1} * p2^{\propto 2} * \ldots * pk^{\propto k}$

atunci

$$exp_{n!}(\boldsymbol{a}) = \boldsymbol{min}\{exp_{n!}(\boldsymbol{p1^{\propto 1}}), exp_{n!}(\boldsymbol{p2^{\propto 2}}), \ldots exp_{n!}(\boldsymbol{pk^{\propto k}})\}$$

Aplicație propusă spre rezolvare

Se dă un număr natural n, de cel mult 5 cifre. Scrieți un program care calculează cu câte zerouri se termină n!.

Date de intrare: Fișierul text **numar.in** conține pe primul rând, un numărul de prelucrat, număr natural de cel mult 5 cifre.

Date de ieșire: Numărul de zerouri cu care se termină n! se afișează pe primul rând al fișierul **numar.out.**
Restricții: numărul dat în fișierul de intrare are cel mult 5 cifre

numar.in	numar.out
1958	467

Cum putem aplica formula:

Numărul de zerouri este dat de $exp_{1958!}(10) = \min \{ exp_{1958!}(2), exp_{1958!}(5)\}$

$exp_{1958!}(5) = [1958/5] + [1958/25] + [1958/125] + [1958/625] = 467$

Descompunerea unui număr în factori primi

Orice număr natural nenul, care nu este număr prim poate fi scris ca un produs de numere naturale prime. Scrierea unui număr natural ca un produs de numere naturale prime se numeşte **descompunere în factori primi** a numărului natural respectiv. **Exemple**:

720	2
360	2
180	2
90	2
45	3
15	3
5	5
1	

Formulă – numărul de divizori ai unui număr

Prin formula lui Euler bazată pe descompunerea în factori primi a lui n

Dacă în urma descompunerii în factori primi se obţine produsul $d1^{e1} * d2^{e2} * \ldots dk^{ek}$ unde d1, d2, ...dk sunt factorii primi iar e1, e2,... exponenţii acestora, numărul divizorilor lui n se calculează după formula:

nrdiv = (1+e1)* (1+e2)*...* (1+ek)

Algoritmul în pseudocod:

```
citeşte n (număr natural)  // numărul care se va descompune în factori primi
scrie "Numărul ", n, " descompus în factori primi este egal cu: "
d = 2                      // primul factor la care incercăm sa împărţim pe n
┌cât timp n>=1 execută
│   s=0                    //număr de câte ori factorul d îl divide pe n
│   ┌cât timp n%d=0 execută
│   │   s=s+1
│   │   n=n/d
│   └■
│   ┌dacă s ≠ 0 atunci
│   │  scrie d," la puterea ", s
│   └■
│   d=d+1
└■
```

Aplicație propusă spre rezolvare: 3 Divizori (varena.ro)

Se dă un şir de N numere naturale. Să se spună câte dintre aceste numere au exact trei divizori.

Date de intrare:
Fişierul de intrare treidiv.in conţine pe prima linie numărul natural N, iar pe următoarea linie N numere naturale, reprezentând elementele şirului.

Date de ieşire:
În fişierul de ieşire treidiv.out se va găsi un singur număr natural, reprezentând numărul de numere din şir care au exact trei divizori.
Restricții:

- $1 \le N \le 100.000$
- Elementele şirului sunt numere naturale mai mici sau egale cu 1.000.000

treidiv.in	treidiv.out

3 4 12 3	1