

**Descrierea problemelor****Ziua 1, Clasele 7 – 9**

<b>Denumirea problemei</b>	<b>Numărul de puncte alocat problemei</b>	<b>Denumirea fișierului sursă</b>
Formule	100	formule.pas formule.c formule.cpp
Pagini	100	pagini.pas pagini.c pagini.cpp
Arta plastica	100	arta.pas arta.c arta.cpp
<b>Total</b>	<b>300</b>	-

## Formule

Se consideră următoarele formule metalingvistice:

$\langle \text{Cifră} \rangle ::= 0|1|2|3|4|5|6|7|8|9$

$\langle \text{Număr} \rangle ::= \langle \text{Cifră} \rangle \{ \langle \text{Cifră} \rangle \}$

$\langle \text{Semn} \rangle ::= +|-|*|/$

$\langle \text{Expresie} \rangle ::= \langle \text{Număr} \rangle | \langle \text{Expresie} \rangle \langle \text{Semn} \rangle \langle \text{Expresie} \rangle$

**Sarcină.** Scrieți un program care returnează valoarea DA dacă șirul de caractere  $S$  este conform definiției unității lexicale  $\langle \text{Expresie} \rangle$  și NU în caz contrar.

**Date de intrare.** Intrarea standard conține pe fiecare linie câte un șir de caractere  $S$ .

**Date de ieșire.** Ieșirea standard va conține pe fiecare linie valoarea DA dacă șirul de caractere din linia respectivă a fișierului de intrare corespunde definiției și NU în caz contrar.

### Exemplu.

#### Intrare

```
1+3014-4+629/1235-3*2
+1+3469
4509/293+1
```

#### Ieșire

```
DA
NU
DA
```

**Restricții.** Șirul nevid  $S$  conține cel mult 250 de caractere. Timpul de execuție nu va depăși 0,1 secunde. Programul va folosi cel mult 1 Megaoctet de memorie operativă. Intrarea standard conține cel mult 100 de linii. Fișierul sursă va avea denumirea `formule.pas`, `formule.c` sau `formule.cpp`.

## Pagini

În biblioteca școlii se păstrează o ediție rară a uneia din primele cărți ce conține problemele propuse la Olimpiada de Informatică din anul 1987. În general, fiecare din paginile acestei cărți ar fi trebuit să aibă tipărită în subsol următoarea informație:

*<Numărul paginii curente>/<Numărul total de pagini în carte>.*

Evident, numerele din subsol sunt scrise fără zerouri ne semnificative.

Întrucât pe timpurile de atunci imprimantele mecanice nu erau așa de perfecte ca cele de astăzi, pe unele din pagini caracterul "/" nu a fost tipărit. În consecință, în unele subsoluri de pagină apar șiruri de caractere  $S$ , formate, evident, doar din cifre zecimale, fără prefixe din zerouri.

De exemplu, în subsolul uneia din paginile cărții în cauză apare șirul de caractere 2348. De fapt, în anul 1987, pe această pagină se intenționa să fie tipărită informația 23/48.

Virginia, o elevă pasionată de informatică, a descoperit astfel de erori de tipar și a decis să elaboreze un algoritm de corectare a acestora. Pentru început, ea dorește să afle numărul variantelor posibile  $V$  de inserare a caracterului "/" în șirul  $S$  în așa mod, încât perechile de numere astfel formate să fie verosimile.

De exemplu, în cazul șirului de caractere 2348, perechile verosimile de numere sunt: 2/348 și 23/48. Prin urmare,  $V = 2$ .

**Sarcină.** Scrieți un program, care, cunoscând șirul de caractere  $S$ , determină numărul de variante  $V$ .

**Date de intrare.** Prima linie a intrării standard conține șirul de caractere  $S$ .

**Date de ieșire.** Ieșirea standard va conține pe o singură linie numărul întreg  $V$ .

**Restricții.** Șirul de caractere  $S$  va fi format din cel mult 9 cifre zecimale, fără prefixe din zerouri. Timpul de execuție nu va depăși 0,1 secundă. Programul va folosi cel mult 1 Megaoctet de memorie operativă. Fișierul sursă va avea denumirea `pagini.pas`, `pagini.c` sau `pagini.cpp`.

### Exemplul 1.

*Intrare*

2348

*Ieșire*

2

### Exemplul 2.

*Intrare*

23507645

*Ieșire*

3

## Artă plastică

Abstracționismul este un curent în artele plastice europene, a cărui trăsătură o constituie încercarea de a elimina din reprezentarea operei de artă orice referire la realitățile exterioare, ideile și sentimentele fiind relatate prin pete de culoare sau prin figuri geometrice.

Un renumit pictor tocmai a terminat o nouă operă de artă plastică abstracționistă. Ea este executată pe o pânză albă, dreptunghiulară, cu lățimea  $W$  și înălțimea  $H$ . Pe pânză sunt desenate  $n$  dreptunghiuri colorate, laturile cărora sunt paralele laturilor corespunzătoare ale pânzei. Dreptunghiurii, nu neapărat toate, se pot intersecta.

Pictorul este mândru de opera lui, dar, vai, colecționarul care intenționează să cumpere tabloul, insistă ca prețul lui să fie stabilit ca fiind invers proporțional ariei suprafeței necolorate a pânzei.

Ajutați pictorul să calculeze aria  $S$  a suprafeței necolorate a pânzei.

În scopuri didactice, fiecare din dreptunghiurile  $i$ ,  $i = 1, 2, 3, \dots, n$ , este descris prin coordonatele  $(a_i, b_i)$  ale colțului stânga-sus și coordonatele  $(c_i, d_i)$  ale colțului dreapta-jos.

Amintim, că în sistemul de coordonate carteziene, prima coordonată este abscisa  $x$  (coordonata orizontală), iar a doua – ordonata  $y$  (coordonata verticală) a punctului pe plan. Evident, lățimile se măsoară pe orizontală (axa  $X$ ), iar înălțimile – pe verticală (axa  $Y$ ). Originea sistemului de coordonate se află în colțul stânga-jos al pânzei.

**Sarcină.** Scrieți un program, care, cunoscând dimensiunile pânzei și descrierile dreptunghiurilor colorate de pe ea, calculează aria suprafeței necolorate a pânzei.

**Date de intrare.** Prima linie a intrării standard conține numerele întregi  $W$  și  $H$ , separate prin spațiu. Linia a doua a intrării standard conține numărul întreg  $n$ . Fiecare din următoarele  $n$  linii ale intrării standard conține câte patru numere întregi  $a_i, b_i, c_i, d_i$ , separate prin spațiu. Linia  $(i + 2)$  a intrării standard conține descrierea dreptunghiului  $i$ .

**Date de ieșire.** Ieșirea standard va conține pe o singură linie numărul întreg  $S$ .

**Restricții.**  $1 \leq W, H \leq 100$ ;  $1 \leq n \leq 5000$ ;  $0 \leq a_i, c_i \leq W$ ;  $0 \leq b_i, d_i \leq H$ . Coordonatele colțurilor de dreptunghiuri sunt numere întregi. Timpul de execuție nu va depăși 0,1 secundă. Programul va folosi cel mult 1 Megaoctet de memorie operativă. Fișierul sursă va avea denumirea `arta.pas`, `arta.c` sau `arta.cpp`.

### Exemplul 1.

*Intrare*

5	5
2	
1	1 3 3
2	2 4 4

*Ieșire*

18
----

### Exemplul 2.

*Intrare*

6	7
3	
0	0 5 5
1	1 4 4
2	2 3 3

*Ieșire*

17
----