OLIMPIADA LA INFORMATICĂ

etapa raională/municipală, 13.02.2022, Clasa a 11-a

Timp de lucru: 240 de minute *Mult succes!*

Descrierea generală a problemelor

Nr. crt.	Denumirea problemei	Denumirea fișierului sursă	Restricția referitoare la volumul utilizat de memorie	Restricția referitoare la timpul de execuție	Numărul de teste	Numărul de puncte alocate unui test	Numărul de puncte alocate problemei
1.	Domino	`domino.pas`, `domino.c` sau `domino.cpp`	≤ 8 Mb	≤ 0,1 s	20	10	200
2.	Energie electrică	`energie.pas`, `energie.c` sau `energie.cpp`	≤ 8 Mb	≤ 0,1 s	10	10	100
3.	Expresii aritmetice romane	`numere.pas`, `numere.c` sau `numere.cpp`	≤ 8 Mb	≤ 0,1 s	10	10	100

Total: 3 probleme, 400 de puncte

Notă. În caz de egalitate de punctaje, mai bune vor fi considerate soluțiile cu un timp de execuție mai mic.

În caz de egalitate și a timpilor de execuție, mai bune vor fi considerate soluțiile ce utilizează un volum mai mic de memorie.

Domino

În jocul *dominoul matematic* se utilizează un set complet de piese distincte plate, dreptunghiulare. Fiecare piesă se identifică prin două numere $\{i, j\}$, $i, j \in \{0, 1, 2, ..., m\}$, simbolizate prin puncte (*Fig. 1*). Însemnarea acestor numere pe fiecare placă se efectuează prin divizarea feței respective în două părți egale.

De exemplu, pentru $m = 3$, setul complet include 10 piese distincte: $\{0, 0\}$, $\{0, 1\}$, $\{0, 2\}$, $\{0, 3\}$, $\{1, 1\}$, $\{1, 2\}$, $\{1, 3\}$, $\{2, 2\}$, $\{2, 3\}$, $\{3, 3\}$.

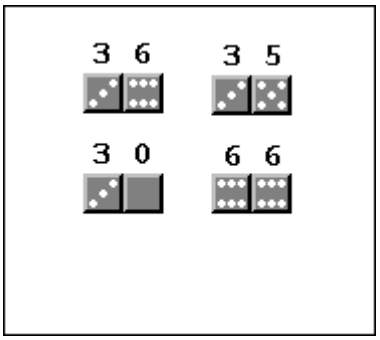

Fig. 1

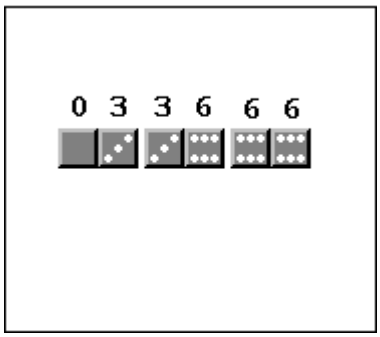

Fig. 2

Utilizând piese distincte de domino în calitate de vagoane imaginare, pe masă de joc pot fi construite „trenuri” cu proprietatea ca cele două numere de pe părțile vecine ale oricăror două piese adiacente sunt egale între ele (*Fig. 2*).

O partidă de domino matematic constă în următoarele:

1) arbitrul indică valoarea concretă a numărului *m*;

2) conform indicațiilor arbitrului, din setul complet sunt eliminate *n* piese de domino;

3) jucătorul trebuie să construiască un număr minim *t* de „trenuri” în așa mod, încât fiecare din piesele rămase să fie inclusă în unul din „trenuri”.

Sarcină. Scrieți un program care calculează numărul *t*.

Date de intrare. Fișierul text `domino.in` conține pe prima linie numărul întreg *m*. Linia a doua a fișierului conține numărul întreg *n*. Următoarele *n* linii ale fișierului de intrare conțin câte două numere întregi i, j separate prin spațiu, fiecare linie reprezentând o piesă, exclusă din setul complet de piese de domino.

Date de ieșire. Fișierul text `domino.out` va conține pe o singură linie numărul întreg *t*.

Restricții. $1 \le m \le 100$. Restricțiile referitoare la timpul de execuție și volumul utilizat de memorie sunt date în Fișa “Descrierea generală a problemelor”. Fișierul sursă va avea denumirea `domino.pas`, `domino.c` sau `domino.cpp`.

Exemplu.

`domino.in`

```
7
2
7 5
3 4
```

`domino.out`

```
2
```

Energie electrică

Ne dorim cu toții o planetă verde. În acest scop este foarte important să economisim energia electrică.

Se consideră abonații rețelei de aprovizionare cu energie electrică dintr-o anumită localitate. Pentru fiecare din acești abonați este cunoscut numele, prenumele și consumul anual, exprimat în kilowați-oră.

Sarcină. Elaborați un program, care:

1) calculează consumul mediu anual de energie electrică per abonat,
2) sortează abonații în ordinea creșterii consumului anual.

Pentru exemplificare, în scopuri didactice, mai jos este prezentat un tabel cu informația despre abonații dintr-o localitate.

Numele, prenumele abonatului	**Consumul electric anual, kilowați-oră**
Munteanu Ion	25
Prisăcaru Viorica	59
Poștașul Valentina	38

Consumul mediu anual de energie per abonat se calculează în felul următor:

$$(26 + 59 + 38) : 3 = 40.6666666666667 \text{ kilowați-oră.}$$

Date de intrare. Fișierul text `energie.in` conține pe prima linie un numărul întreg *n* – numărul de abonați. Fiecare din următoarele *n* linii ale fișierului de intrare conține un șir de caractere: numele, prenumele și consumul anual al unui abonat, separate prin spațiu. Prin alte cuvinte, informația despre fiecare din abonați este structurată în felul următor:

<Nume><Spațiu><Prenume><Spațiu><Număr întreg>

În această descriere *<Nume>* și *<Prenume>* reprezintă șiruri de caractere fără spații, iar *<Număr întreg>* – consumul anual al abonatului respectiv.

Date de ieșire. Fișierul text `energie.out` va conține pe prima linie un număr real – consumul mediu anual de energie per abonat. Numărul real va fi scris pe 10 poziții, aliniat la dreapta, fără factor de scală, cu două cifre după punctul zecimal.

Următoarele *n* linii ale fișierului de ieșire vor conține informațiile despre abonați, în ordinea creșterii consumului anual. Informațiile din aceste linii vor fi structurate exact la fel ca informațiile din fișierul de intrare.

Restricţii. Fișierul de intrare va conține cel mult 10 abonați. Consumul anual al fiecăruia din abonați nu va depăși 4000 de kilowați-oră. Nu există abonați cu unul și același consum. Restricțiile referitoare la timpul de execuție și volumul utilizat de memorie sunt date în Fișa "Descrierea generală a problemelor". Fișierul sursă va avea denumirea `energie.pas`, `energie.c` sau `energie.cpp`.

Exemplu.

`energie.in`

```
3
Munteanu Ion 25
Prisacaru Viorica 59
Postasul Valentina 38
```

`energie.out`

```
      46.7
Munteanu Ion 25
Postasul Valentina 38
Prisacaru Viorica 59
```

Expresii aritmetice romane

În sistemul roman de numerație se folosesc 7 cifre, reprezentate prin următoarele litere ale alfabetului latin: `I`, `V`, `X`, `L`, `C`, `D` și `M`. În sistemul zecimal, aceste cifre au următoarea semnificație:

Cifra din sistemul roman de numerație	**Semnificația cifrei romane în sistemul zecimal de numerație**
`I`	1
`V`	5
`X`	10
`L`	50
`C`	100
`D`	500
`M`	1000

Amintim faptul că sistemul roman de informație este unul nepozițional. Numărul scris în sistemul roman se citește de la stânga spre dreapta. Valoarea zecimală a numărului se determină însumând valorile zecimale individuale ale fiecăreia din cifre, cu excepția cazurilor în care cifra curentă precede o cifră de valoare superioară. În acest caz, se scade valoarea zecimală a primei cifre din valoarea zecimală a cifrei următoare.

Exemple:

Numărul în roman de numerație	**Numărul în sistemul zecimal de numerație**
`X`	10
`IX`	9
`XIII`	13
`XIV`	14
`MMCIX`	2109
`MMXXII`	2022

În sistemul roman de numerație, expresiile aritmetice se definesc cu ajutorul următoarelor formule metalingvistice:

<*Cifră romană*> ::= `I` | `V` | `X` | `L` | `C` | `D` | `M`

<*Număr roman*> ::= <*Cifră romană*> {<*Cifră romană*>}

<Operator> ::= + | −

<*Expresie aritmetică romană*> ::= <*Număr roman*> {<*Operator*> <*Număr roman*>}

Exemple:

```
MMXXII
IX+XIII
XIV+MMCIX-MMXXII+V-XL
```

Sarcină. Elaborați un program, care calculează valorile expresiilor aritmetice romane.

Date de intrare. Fișierul text `expresii.in` conține pe o singură linie o expresie aritmetică romană.

Date de ieşire. Fișierul text `expresii.out` va conține pe o singură linie un număr întreg – valoarea expresiei aritmetice romane din fișierul de intrare, scrisă în sistemul zecimal de numerație.

Restricții. Expresia aritmetică din fișierul de intrare va conține cel mult 255 de caractere. Restricțiile referitoare la timpul de execuție și volumul utilizat de memorie sunt date în Fișa "Descrierea generală a problemelor". Fişierul sursă va avea denumirea `expresii.pas`, `expresii.c` sau `expresii.cpp`.

Exemplu.

expresii.in	expresii.out
XIV+MMCIX-MMXXII+V-XL	66