

PROBLEME PROPUSE DE INFORMATICĂ PENTRU GIMNAZIU

IG1) Se citesc de la tastatură numerele a și b . Dacă numerele sunt prime între ele să se afișeze cel mai mare număr care nu poate fi scris în forma $ax+by$, $x,y \in \mathbb{N}$.

prof. Muscalu Adrian

IG2) Broașcuța Oac joacă un joc de sărituri pe pătrate numerotate. Ea începe pe pătratul 0 și apoi sare 1 pas înainte, apoi 4 pași, apoi 7 pași, apoi din nou 1 pas, apoi 4 și așa mai departe, repetând la infinit.

Un pătrat este bun dacă Oac vizitează pătratul în timpul jocului.

a) Să se afișeze primele n pătrate pe care sare broașcuța

b) Pentru n dat să se afișeze un mesaj dacă n este sau nu termen al șirului de pătrate pe care le vizitează Oac



prof. Muscalu Adrian

IG3) Numărul 7331 este numit superprim de lungime 4 deoarece 7331, 733.73 și 7 sunt prime. Scrieți un program care citește un număr de N și afișează toate superprimele de lungime N .

CONSTRÂNGERI TEHNICE:

1. Numărul 1 nu este un număr prim.
2. Numărul N satisface condiția $1 \leq N \leq 8$.
3. Se iese din program dacă numărul de coaste este introdus ca 0.

Exemplu

Dacă $N=4$ se afișează

2333 2339 2393 2399 2939 3119 3137 3733 3739 3793 3797 5939 7193 7331 7333 7393

prof. Muscalu Adrian

IG4) Câinele de la aeroport se oprește lângă un cărucior încărcat cu n valize. Una dintre valize conține arahide de contrabandă. Câinele poate să spună dacă arahidele sunt ascunse în oricare dintre grupurile de valize, dar devine obosit dacă trebuie să miroasă prea mult. Care este cel mai mic număr de grupe de valize care trebuie să miroasă pentru a izola valiza cu arahidele?

prof. Muscalu Adrian

IG5) Sunteți un biolog care lucrează într-un laborator mare. În ultima lună ați crescut o cultură a bacteriilor preferate, *Bacillus Fortranicus*. Sunteți interesați în mod deosebit de modul în care crește în medii ostile.

Astăzi este ultima zi a experimentării și ați pierdut carnetul cu notițe. Îl sunați pe colegul dumneavoastră care spune

"Nu, nu-mi amintesc cât de multe bacterii am început, fie. Dar îmi amintesc că era un număr impar. Da, și numărul de bacterii sa dublat în fiecare zi. Deși v-ați pierdut notițele, puteți încă să numărați numărul total de bacterii pe care le aveți acum. Combinând acest total cu informațiile asistentului, trebuie să scrieți un program pentru a răspunde la cele două întrebări inițiale. Adică trebuie să calculezi

(i) câte bacterii ai început și

(ii) de câte zile a trecut experimentul.

Intrare

Fișierul de intrare va fi format dintr-o singură linie. Această linie va conține un număr întreg n reprezentând numărul de bacterii pe care le aveți acum. Sunteți garantat că $1 \leq n \leq 30.000$.

Ieșire

Fișierul de ieșire trebuie să cuprindă cele două întregi b și d pe o singură linie, unde b reprezintă numărul de bacterii la începutul experimentului și d reprezintă numărul de zile pentru care experimentul a fost rulat. Aceste două întregi trebuie separate de un singur spațiu.

Exemplu

Date de intrare

136

Date de ieșire

17 3

prof. Muscalu Adrian

PROBLEME PROPUSE DE INFORMATICĂ PENTRU LICEU

IL1) Două numere se consideră "prietene" dacă sunt diferite, iar suma divizorilor fiecărui număr (inclusiv 1, dar excluzând numărul însuși) este egală cu celălalt număr.

De exemplu: 2620 este divizibil cu 1, 2, 4, 5, 10, 20, 131, 262, 524, 655 și 1310; acestea se adună și se obține 2924. 2924 este divizibil cu 1, 2, 4, 17, 34, 43, 68, 86, 172, 731 și 1462; acestea adunate dau 2620. Prin urmare, 2620 și 2924 sunt "prietene".

a) Scrieți un program care introduce două numere (care vor fi mai mici de 10.000) și apoi imprimă "prietene" dacă sunt prietene sau "nu sunt prietene" altfel.

b) Care este cea mai mică pereche de numere amiabile?

IL2) Factorialul unui întreg pozitiv n , notat $n!$, este produsul tuturor întregilor pozitivi $\leq n$. Cu alte cuvinte $n! = n \times (n - 1) \times \dots \times 2 \times 1$. Factorialele cresc foarte repede; de exemplu $100!$ are peste 150 de cifre. Scrieți un program care citește un întreg m ($1 \leq m \leq 1000000$) și afișează două numere întregi, cea mai din dreapta cifră nenulă a lui $m!$ urmat de numărul de zerouri după acea cifră.

De exemplu pentru $n = 10$ se va afișa 8 2

IL3) Este un motiv de îngrijorare dacă sunteți treisprezecefobic - aproape fiecare număr întreg conține numărul 13. Există un singur număr întreg între 1 și 100 (și anume 13), dar există alte 19 între 100 și 1000 (cum ar fi 131 și 213). Între 1000 și 10000 există 279 de astfel de numere și există 3671 între 10000 și 100000. Rețineți că numai cele care sunt neîntrerupte sunt numărate, de exemplu, 103 nu conține modelul 13. Se citește un număr n , scrieți un program pentru a determina câte numere întregi dintr-un anumit interval conțin acel număr. Intrarea va fi pe două linii: prima linie va conține un întreg n care indică numărul ($1 \leq n \leq 99$), a doua linie va conține doi întregi a și b ($1 \leq a < b \leq 2^{30}$), indicând intervalul. Rezultatul ar trebui să fie format dintr-un singur număr, indicând numărul de întregi între a și b (inclusiv) care conțin numărul n .

De exemplu, dacă la intrare avem

13

13 1350

leșirea este 84

IL4) O secvență de șiruri de caractere este generată după cum urmează: primele două șiruri de caractere sunt date și fiecare șir succesiv se face prin imbinarea celor două șiruri de caractere precedente (în ordinea inițială). De exemplu, dacă primul șir este abc și al doilea cde, cel de-al treilea șir va fi abccde și cel de-al patrulea va fi cdeabccde. Să se determine al n -lea caracter din succesiunea șirurilor de caractere. În exemplul precedent al doilea caracter este b, așa cum este și al 11-lea. Cel de-al 13-lea caracter, care se întâmplă să fie primul caracter din șirul al 4-lea, este c. Scrieți un program care citește primele două șiruri de caractere și un număr natural n și afișează al n -lea caracter din șir

IL5) Jocul LIFE este o simulare pe calculator, jucată pe o placă dreptunghiulară de pătrate. Fiecare pătrat are două stări posibile 0-mort și 1-viu, Starea pătratelor de la următoarea generație este complet determinată de starea actuală și de câteva reguli simple.

Configurația de la generația 0 este selectată de utilizator.

Fiecare pătrat de pe tablă are opt vecini, cei care îi ating în mod direct (inclusiv pe diagonală), cu excepția pătratelor de graniță care au unele dintre aceste bucăți lipsă.

Regulile pentru generarea următoarei generații sunt:

Supraviețuirea: Orice pătrat viu, cu doi sau trei vecini vii, supraviețuiește pentru "următoarea generație".

Moartea: Orice pătrat viu cu mai puțin de doi vecini vii moare de singurătate. Orice pătrat viu cu mai mult de trei vecini vii moare din cauza aglomerației

Nașterea: Orice pătrat mort cu exact trei vecini vii învie pentru următoarea generație.

Regulile rămân aceleași pentru pătratele de frontieră, deși au mai puțini vecini - cu alte cuvinte, pătratele din afara plăcii 11x11 trebuie tratate permanent ca fiind moarte. De asemenea, toate nașterile, decese și supraviețuirea dintre generații apar simultan.

De exemplu:

	00100		00000		00000
	00010		01010		00010
Generația 1	01110	Generația 1	00110	Generația 2	01010
	00000		00100		00110
	00000		00000		00000

Trebuie să scrieți un program pentru a juca LIFE pe o placă de 11x11. Programul ar trebui să citească mai întâi într-o placă de 5x5, care va fi sub forma a cinci linii de cinci caractere (fie 0, fie 1). Această placă de 5x5 ar trebui utilizată ca secțiune centrală a plăcii 11x11 pentru generația 0, restul plăcii 11x11 este considerat mort. Aceasta va fi generația 0 pentru întreaga derulare a programului dvs.

a) Odată ce acest lucru este făcut, afișați generația 0.

b) Până la terminarea programului, trebuie să așteptați în mod repetat pentru introducerea de la tastatură astfel:

i) Dacă primiți un număr întreg cu un hash înaintea lui (cum ar fi # 5), ar trebui să calculați generația 'n' și să o afișați pe ecran.

ii) Dacă primiți un întreg "n" cu un plus înainte de acesta (cum ar fi +5), ar trebui să calculați generațiile "n" de la ultima generație afișată și apoi să afișați cea nouă pe ecran.

ii) Dacă primiți numărul -1, trebuie să terminați.

iv) Ignorați orice altă intrare.

Exemplu

Se citește

00000

00000

00010

00001

00111

Se afișează

00000000000
00000000000
00000000000
00000000000
00000000000
00000010000
00000001000
00000111000
00000000000
00000000000
00000000000

Se citește

#5

Se afișează

00000000000
00000000000
00000000000
00000000000
00000000000
00000000000
00000000000
00000000000
00000010100
00000001100
00000001000
00000000000

Se citește

+1

Se afișează

00000000000
00000000000
00000000000
00000000000
00000000000
00000000000
00000000000
00000000100
00000010100
00000001100
00000000000

Se citește

-1