**Săptămâna 7. Probleme de concurs cu instrucțiunile decizionale**

**#2014 Canguri**

**Enunț**

La proba de atletism s-au calificat trei canguri care trebuie să efectueze sărituri pe o pistă de **800** metri.

Fiecare cangur se aliniază la linia de start, își lipește pe marsupiu eticheta cu numărul său de concurs iar apoi când se dă stratul efectuează săritură după săritură până trece linia de sosire.

Cunoscând lungimea săriturii fiecărui cangur (exprimată în metri) și faptul că nu există doi canguri cu sărituri de aceeași lungime, scrieți un program care să determine:

**Cerința**

1. Numărul de concurs al cangurului care ajunge primul la linia de sosire;
2. Numărul săriturilor efectuate de cangurul care a trecut primul linia de sosire.

**Date de intrare**

Din fișierul de intrare **canguri.in** se citesc:

* de pe primele trei linii, câte două numere naturale despărțite prin câte un spațiu, ce reprezintă numărul de concurs respectiv lungimea săriturii fiecărui cangur.

**Date de ieșire**

În fișierul **canguri.out** se va scrie:

* pe prima linie, numărul de concurs al cangurului care trece primul linia de sosire;
* pe cea de-a doua linie, un număr natural ce reprezintă numărul săriturilor efectuate de cangurul care a trecut primul linia de sosire.

**Restricții și precizări**

* Numerele scrise pe etichetele de concurs sunt numere naturale cu maximum trei cifre
* Pentru determinarea corectă a numărului de concurs se acordă **60%** din punctaj; pentru determinarea corectă a numărului de sărituri se acordă **40%** din punctaj

**Exemplu**

|  |  |
| --- | --- |
| **canguri.in** | **canguri.out** |
| 769 2  354 6  129 4 | 354  134 |

**Explicație**

* Primul cangur are numărul de concurs **769** și lungimea săriturii egală cu **2** metri.
* Cel de-al doilea cangur are numărul de concurs **354** și lungimea săriturii de **6** metri.
* Cel de-al treilea cangur are numărul de concurs **129** și lungimea săriturii de **4** metri.
* Primul va trece linia de sosire cangurul cu numărul de concurs **354**.
* El va efectua **134** de sărituri pentru a trece linia de sosire

**#2311 Rapunzel**

**Cerința**

Rapunzel s-a născut într-un ținut îndepărtat într-un regat necunoscut. Mama vitregă a închis-o pe Rapunzel într-un turn foarte înalt ce avea **N** metri. Aici Rapunzel urma să-și petreacă toată copilăria.

Pentru a-i trece timpul mai ușor, Rapunzel cânta din zori și până în seară, cântecul ei auzindu-se în tot ținutul. Atras de vocea de privighetoare a fetei, Flynn Rider și-a propus să se cațere pe pereții exterior ai turnului și în fiecare zi să procedeze astfel: de la răsăritul până la asfințitul soarelui să parcurgă **M1** metri iar apoi, de la asfințit până la miezul nopții încă **M2** metri.

Scrieți un program care determină după câte zile ajunge Flynn Rider la Rapunzel.

**Date de intrare**

De pe prima linie a fișierului **rapunzel.in** se citesc trei numere naturale **N**, **M1** și **M2**, în această ordine, despărțite prin câte un spațiu, având semnificația din enunț.

**Date de ieșire**

În fișierul **rapunzel.out** se va afișa, pe prima linie, un număr natural ce reprezintă numărul de zile necesar lui Flynn Rider pentru a ajunge la Rapunzel.

**Restricții și precizări**

* **1 ≤ n ≤ 5 000 000 000**
* **1 ≤ M1, M2 ≤ 2 500**

**Exemplul 1:**

|  |  |
| --- | --- |
| **rapunzel.in** | **rapunzel.out** |
| 7 3 4 | 1 |

**Explicație**

* Turnul are înălțimea de **7** metri. Flynn urcă până la asfințit **3** metri iar apoi încă **4** metri. După o zi el va ajunge la Rapunzel.

**Exemplul 2:**

|  |  |
| --- | --- |
| **rapunzel.in** | **rapunzel.out** |
| 10 1 3 | 3 |

**Explicație**

* Turnul are înălțimea de **10** metri. Flynn urcă până la asfințit **1** metru iar până la miezul nopții încă **4** metri. După **3** zile el va ajunge la Rapunzel.

**#1309 Gresie**

**Cerința**

Se consideră o încăpere de formă dreptunghiulară cu dimensiunile **a×b**. Încăperea trebuie pavată cu gresie de formă pătratică, de dimensiune **d**. Știind că o bucată de gresie se poate folosi întreagă sau tăiată, să se determine numărul minim de bucăți de gresie sunt necesare pentru pavarea încăperii.

**Date de intrare**

Programul citește de la tastatură numerele naturale **a b d**.

**Date de ieșire**

Programul afișează pe ecran numărul minim de bucăți de gresie necesare pentru pavarea încăperii.

**Restricții și precizări**

* **1 ≤ d ≤ a,b ≤ 32000**
* din motive estetice, pentru fiecare piesă de gresie tăiată se folosește o bucată întreagă.

**Exemplu**

|  |  |
| --- | --- |
| **Intrare** | **Ieșire** |
| 12 17 3 | 24 |

**#1943**[**Bacterie**](https://www.pbinfo.ro/probleme/1943/bacterie)

**Enunț**

O echipă de arheologi a descoperit o hartă străveche a Ținutului de Nord, care era locuit de o civilizație condusă după reguli matematice foarte riguroase. Conform acestei hărți, Ținutul de Nord era împărțit în **n** rânduri a câte **m** comitate, fiecare comitat ocupând o suprafață pătrată de un hectar.

Însă descoperirile au mai arătat că această civilizație a fost atacată de la sud-vest de o bacterie periculoasă, ce a acționat astfel: în primul an, a infectat comitatul din colțul din stânga jos al hărții, în al doilea an a infectat cele două comitate vecine cu primul, în al treilea an a infectat cele trei comitate vecine cu anterioarele două și așa mai departe, infecția oprindu-se când bacteria a ajuns la marginea de sus sau la marginea din dreapta a hărții.

|  |  |  |
| --- | --- | --- |
| Primul an | Al doilea an | Al treilea an |
|  |  |  |

**Cerința**

Scrieți un program care să determine numărul de comitate rămase neinfectate după oprirea expansiunii bacteriei.

**Date de intrare**

Fișierul de intrare **bacterie.in** conține separate printr-un spațiu, **n** și **m**, dimensiunile hărții.

**Date de ieșire**

Fișierul de ieșire **bacterie.out** va conține numărul de comitate rămase neinfectate după oprirea expansiunii bacteriei.

**Restricții și precizări**

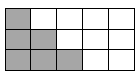
1 ≤ n ≤ 1.000.000.000, 1 ≤ m ≤ 1.000.000.000, numere naturale

**Exemplu**

|  |  |
| --- | --- |
| **bacterie.in** | **bacterie.out** |
| 3 5 | 9 |

**Explicație**

Harta Ținutului de Nord cuprinde **3** rânduri a câte **5** comitate fiecare, având în total **15** comitate. Expansiunea bacteriei s-a oprit după **3** ani, deoarece a atins marginea de sus; au rămas **9** comitate neinfectate.



**#2908 planta**

Ghiță a primit de ziua lui o plantă exotică, ce se comportă foarte ciudat. El a măsurat-o când a primit-o și a constatat că are **D** cm, apoi a văzut că se dezvoltă într-un ritm special:

* În prima zi, planta crește cu **A** cm
* În a doua zi, descrește cu **B** cm
* În a treia zi, iar crește cu **A** cm
* În a patra zi, descrește din nou cu **B** cm

etc.

Pe scurt, în zilele cu număr de ordine impar crește cu **A** cm, iar în cele cu număr de ordine par, descrește cu **B** cm.

**Cerința**

Știind **D**, înălțimea inițiala a plantei și valorile **A** și **B** cu care aceasta crește, respectiv descrește, să se afla ce înălțime va avea planta lui Ghiță la finalul celei de-a **N**-a zile.

**Date de intrare**

Pe prima linie a fișierului **planta.in** se vor afla patru numere naturale **D A B N** în aceasta ordine, separate prin câte un spațiu, cu semnificațiile din enunț.

**Date de ieșire**

Pe prima linie a fișierului **planta.out** se va afla un număr **H**, semnificând înălțimea finală a plantei în **cm** la finalul celei de-a **N**-a zile.

**Restricții și precizări**

* **0 ≤ D ≤ 100**
* **1 ≤ B ≤ A ≤ 1 000 000**
* **1 ≤ N ≤ 1 000 000 000**

Pentru **50%** dintre teste, **1 ≤ N ≤ 1 000 000**

Se garantează că pentru toate testele valorile se încadrează în tipul **int**.

Exemplul 1:

|  |  |
| --- | --- |
| **planta.in** | **planta.out** |
| 4 5 2 3 | 12 |

**Explicație**

După prima zi: **H = 4 + 5 = 9**.

După a doua zi: **H = 9 – 2 = 7**.

După a treia zi: **H = 7 + 5 = 12**.

Deci la finalul celei de-a **3**-a zile, înălțimea plantei o sa fie **12** cm.

**Exemplul 2:**

|  |  |
| --- | --- |
| **planta.in** | **planta.out** |
| 57 1000 1000 120 | 57 |

**#1308 CifreComune**

**Cerința**

Se citesc două numere naturale **n** și **m** cu exact două cifre fiecare. Să se decidă dacă cele două numere au cifre comune.

**Date de intrare**

Programul citește de la tastatură numerele **n m**.

**Date de ieșire**

Programul afișează pe ecran mesajul **DA** dacă cele două numere au cifre comune, respectiv **NU** în caz contrar.

**Restricții și precizări**

* **10 ≤ n, m ≤ 99**

**Exemplu**

**Intrare:**

27 62

**Ieșire:**

da

**#1310 CifDiv**

**Cerința**

Se citesc două numere naturale **n** și **m** cu exact trei cifre fiecare. Să se afle câte cifre din **n** divid pe **m**.

**Date de intrare**

Programul citește de la tastatură numerele **n m**.

**Date de ieșire**

Programul afișează pe ecran valoarea cerută.

**Restricții și precizări**

* **100 ≤ n,m ≤ 999**

**Exemplu**

|  |  |
| --- | --- |
| **Intrare** | **Ieșire** |
| 273 600 | 2 |

**Explicație**

Cifrele **2** și **3** din **273** îl divid pe **600**.

**#1002 Paginare\*\*\***

**Cerința**

Pentru numerotarea paginilor unei serii enciclopedice formate din unul sau mai multe volume se presupune că se folosesc **n** cifre. Fiecare volum are **300** de pagini, eventual cu excepția ultimului volum care ar putea avea mai puține.

Pentru **n** dat, să se determine numărul de volume din serie **V** și numărul de pagini **P** ale ultimului volum. Dacă nu este posibilă numerotarea paginilor folosind **n** cifre, se va afișa mesajul **IMPOSIBIL**.

**Date de intrare**

Programul citește de la tastatură numărul **n**.

**Date de ieșire**

Programul va afișa pe ecran numerele **V P**, separate prin exact un spațiu, sau mesajul **IMPOSIBIL**.

**Restricții și precizări**

* **1 ≤ n ≤ 1 000 000 000**
* numerotarea paginilor se face începând cu **1**

**Exemplul 1**

|  |  |
| --- | --- |
| **Intrare** | **Ieșire** |
| 999 | 2 105 |

**Exemplul 2**

|  |  |
| --- | --- |
| **Intrare** | **Ieșire** |
| 900 | IMPOSIBIL |

**#1470 Parcare**

**Enunț**

În parcările din Iași, intrarea în parcare este permisă dacă șoferul care dorește să parcheze solicită acest lucru.

Solicitarea se face prin apăsarea unui buton la automatul aflat la intrarea în parcare, acțiune care are ca efect eliberarea unui tichet de parcare și deschiderea barierei. Pe tichet sunt trecute data și ora intrării în parcare.

La plecare, șoferul scanează tichetul la automatul de plată. În urma scanării automatul calculează și afișează suma de plată. Calculul sumei se face astfel:

* se citește de la ceasul automatului *data* și *ora* plecării
* se calculează diferența de timp dintre *ora* *plecării* și *ora* *intrării* în parcare
* diferența calculată se rotunjește la număr întreg de *ore*, în sus
* se înmulțește timpul exprimat în *ore* cu *tariful* orar

**Cerința**

Cunoscându-se *data* și *ora* *intrării* în parcare, *data* și *ora* plecării din parcare și *tariful* orar, să se determine *timpul* cât a stat mașina în parcare și *suma* de plată.

**Date de intrare**

Fișierul de intrare **parcare.in** conține pe prima linie *data* și *ora* *intrării* în parcare, separate printr-un spațiu, sub forma:

**zz ll hh mm**.

Pe linia a doua a fișierului de intrare se găsesc *data* și *ora* plecării din parcare, separate printr-un spațiu, sub forma:

**zz ll hh mm**.

Linia a treia conține un număr natural **t** reprezentând tariful orar exprimat în *lei*.

Pe linia a patra a fișierului de intrare se găsește una dintre valorile **1** sau **2** reprezentând cerința: **1**, dacă se cere determinarea timpului cât a stat mașina în parcare, timp exprimat în minute, respectiv **2**, dacă se cere determinarea sumei de plată pentru parcare.

**Date de ieșire**

Fișierul de ieșire **parcare.out** conține pe prima linie o valoarea naturală reprezentând timpul cât mașina a stat în parcare, exprimat în minute, dacă cerința a fost **1**, respectiv suma de plată, dacă cerința a fost **2**.

**Restricții și precizări**

* Mașina stă în parcare cel puțin **1** minut dar nu poate sta mai mult de **7** zile.
* Dacă timpul de staționare nu depășește **15** minute, nu se percepe taxă.
* Cele două date sunt din același an, care nu este bisect.
* Minutul de plecare din parcare nu se contorizează.
* Minutele unei ore sunt numerotate de la **0** la **59**.
* **1 <= t <= 100**

Pentru cerința **1** se acordă **50%** din punctaj, iar pentru cerința **2** se acordă încă **50%** din punctaj.

**Exemplul 1**

|  |  |
| --- | --- |
| **parcare.in** | **parcare.out** |
| 30 01 17 33  30 01 21 01  3  1 | 208 |

**Explicație**

Intrarea în parcare s-a făcut în data de **30** *ianuarie* ora **17:33**. Plecarea s-a făcut în aceeași zi la ora **21:01**. Deci timpul cât mașina a stat în parcare este **27 + 3\*60 + 1 = 208** minute.

**Exemplu 2**

|  |  |
| --- | --- |
| **parcare.in** | **parcare.out** |
| 30 01 17 33  30 01 21 01  3  2 | 12 |

**Explicație**

Tariful orar este **3** lei/oră. Se deduce că **208 minute = 3 ore și 28 minute**, deci se plătesc **4** ore, **4\*3=12**.

**#3480 9lan**

**Cerința**

Se dă un număr natural **n**. Calculați ultima cifră a lui **9n**.

**Date de intrare**

Fișierul de intrare **9lan.in** conține pe prima linie numărul **n**.

**Date de ieșire**

Fișierul de ieșire **9lan.out** va conține ultima cifră a lui **9n**.

**Restricții și precizări**

* **0 ≤ n ≤ 100000**

**Exemplu**

|  |  |
| --- | --- |
| **9lan.in** | **9lan.out** |
| 6 | 1 |

**#3479 2lan**

**Cerința**

Se da un număr **n**. Calculați ultima cifră a lui **2n**.

**Date de intrare**

Fișierul de intrare **2lan.in** conține pe prima linie numărul **n**.

**Date de ieșire**

Fișierul de ieșire **2lan.out** va conține ultima cifra a lui **2n**.

**Restricții și precizări**

* **0 ≤ n ≤ 100.000**

**Exemplu**

|  |  |
| --- | --- |
| **2lan.in** | **2lan.out** |
| 6 | 4 |

**Explicație**

**26=64** și ultima sa cifră este **4**.

**#2987 Buletin**

**Cerința**

Se dă un număr natural **n** cu exact **13** cifre reprezentând un cod numeric personal. Să se afișeze anul, luna și ziua nașterii deținătorului.

**Date de intrare**

Programul citește de la tastatură numărul natural **n**, reprezentând codul.

**Date de ieșire**

Programul va afișa pe ecran numărul **A**, reprezentând anul nașterii, **L**, reprezentând luna nașterii și **Z**, reprezentând ziua nașterii, separate prin spații.

**Restricții și precizări**

* **n** este un număr natural cu exact **13** cifre;
* **A**, **L**, **Z**, vor fi afișate sub forma **AA LL ZZ**.
* Dacă unul dintre numere **x** este de o cifră, se va afișa sub forma **0x**.

**Exemplu**

|  |  |
| --- | --- |
| **Intrare** | **Ieșire** |
| 5070521071145 | 07 05 21 |
| 2961103042367 | 96 11 03 |

**Explicație**

Deținătorul primului **CNP** are data nașterii **21 mai, 2007**, iar deținătorul celui de-al doilea **CNP** are data nașterii **3 noiembrie, 1996**.