

PROBLEMA 1

100 puncte

NUME

De ziua lui, Cristi a primit în dar un animăluț mic și pufos. Evident, acesta dorește să îi dea un nume. Cristi a decis să îi dea un nume strict legat de numărul său protector. Cunoscând numărul protector, numele animăluțului se va determina astfel: va fi un șir de litere MARI ale alfabetului latin, de lungime minimă cu proprietatea că suma diferențelor în modul a literelor vecine este egală cu numărul lui protector.

Concret: dacă avem numele FLAFFY, obținem:

$$|F - L| + |L - A| + |A - F| + |F - F| + |F - Y| = |6 - 12| + |12 - 1| + |1 - 6| + |6 - 6| + |6 - 25| = 6 + 11 + 5 + 0 + 19 = 41$$

Deci codul numelui FLAFFY este 41

Cerința:

Cristi vă oferă onoarea de a afla numele micuțului său animăluț.

Date de intrare:

În fișierul *nume.in*, pe prima și singura linie, se va afla P, numărul protector dat de Cristi.

Date de ieșire:

În fișierul *nume.out*, pe prima și singura linie se va afișa un șir de litere MARI ale alfabetului latin, cu proprietățile cerute în enunț.

Restricții și precizări:

$$1 \leq P \leq 4.000.000$$

$$A = 1, B = 2, C = 3 \dots Z = 26$$

În cazul în care șirul afișat are suma corectă și număr minim de caractere, primiți 20% din punctajul pe acel test.

Dacă șirul afișat are proprietățile de mai sus și este și minim lexicografic, primiți punctajul integral pe acel test.

Un șir A este mai mic lexicografic decât un șir B dacă pe prima poziție unde $A[i] \neq B[i]$, $A[i] < B[i]$.

Exemplu:

nume.in	nume.out	Explicație
19	AT	$A = 1, T = 20$ $ A - T = 1 - 20 = 19$
808	ARAZAZAZAZAZAZAZAZAZAZAZAZAZAY	

Timp maxim de execuție: 0.1 secunde / test

Memorie totală disponibilă: 2MB și 1MB pentru stivă

Dimensiunea maximă a sursei: 5KB

PROBLEMA 2

100 puncte

RUP

Echipa de debateri pleacă la un concurs cu trenul, în altă localitate. Drumul fiind lung, antrenorul le propune copiilor următorul joc: copiii scriu pe bilețele numere pe care le pun apoi pe masa din compartiment. Antrenorul scrie la rândul său pe o bucată de hârtie un șir de cifre. Apoi le cere copiilor să găsească modalități de a rupe șirul de cifre astfel încât să se formeze numai numere scrise pe bilețele de pe masă.

Cerință

Cunoscând numerele de pe bilețele scrise de copii, șirul de cifre scris de către antrenor, scrieți un program care să determine în câte moduri poate fi rupt șirul de cifre astfel încât să obținem numai numere scrise pe bilețele, precum și care este cel mai mare număr de bilețele regăsite la o variantă de rupere.

Date de intrare

Fișierul *rup.in* are următoarea structură:

- pe prima linie un număr natural n reprezentând numărul de bilețele;
- pe a doua linie cele n numere scrise de către copii pe bilețele;
- pe a treia linie șirul de cifre scris de către antrenor.

Date de ieșire

Fișierul *rup.out* conține cele două valori cerute în enunț, scrise pe aceeași linie, separate printr-un spațiu.

Restricții și precizări:

- $2 \leq n \leq 100$;
- șirul de cifre conține maxim 20 de cifre;
- numerele scrise pe bilețele au maxim 4 cifre;
- doi copii pot să scrie același număr pe bilețel;
- datele de test furnizează cel puțin o soluție.

Exemplu:

rup.in	rup.out	Explicație
9 57 23 7 5 231 4 375 75 237 2375	3 3	Șirul 2375 poate fi rupt în 3 moduri, astfel: 237 5 23 75 23 7 5 Cele mai multe bilețele obținute sunt 3

Timp maxim de execuție: 1 secunda/test.

Memorie totală disponibilă: 2 MB, din care 2 MB pentru stivă

Dimensiunea maximă a sursei: 5 KB.

PROBLEMA 3

100 puncte

COD

Fiecarei litere mici din alfabetul englez i se asociază codul ei ASCII, adică un număr natural cuprins între 97 și 122, iar fiecărei litere mari din alfabetul englez i se asociază numărul ei de ordine în alfabet, adică lui A i se asociază 1, lui B i se asociază 2 etc. Fiind dat un număr natural n se cere numărul de variante de codificare ale acestuia folosind litere mici și litere mari.

Date de intrare :

Fișierul **cod.in** conține pe prima linie cifrele numărului n .

Date de iesire:

Fișierul **cod.out** va conține pe prima linie un singur număr reprezentând numărul de variante de codificare ale numărului n **modulo** 40517.

Restricții și precizări:

Numărul n conține minim 3 cifre și maxim 10^8 cifre.

Exemple:

cod.in	cod.out	Explicații
122	4	Cele patru variante de codificare sunt: ABB, LB, AV, z
121003	2	Cele doua variante de codificare sunt: ABdC, LdC
197	3	Cele trei variante de codificare sunt: AIG, SG, Aa
200	0	Nu exista nicio variantă de codificare
1001	1	Singura variantă de codificare este: dA

Timp maxim de execuție: 0,5 secunde/test.

Memorie totală disponibilă: 1 MB, din care 0,5 MB pentru stivă

Dimensiunea maximă a sursei: 5 KB.

PROBLEMA 4

100 puncte

ETICHETA

La atelierul de vopsitorie fiecare cuva de culoare are un cod format dintr-un singur caracter, una din literele alfabetului ('A' - 'Z', 'a' - 'z') sau o cifră (0..9). Fiecare produs, după ce iese dintr-o cuvă de vopsire, este etichetat corespunzător. Etichetarea se face prin adaugarea pe etichetă a caracterului corespunzător cuvei de vopsire. De exemplu eticheta „As6d” arată că produsul a trecut prin 4 cuve de vopsire: 'A', apoi 's', apoi '6' și apoi 'd'.

În procesul de vopsire este esențial să respecti o anumită ordine a culorilor „critice” pentru a nu compromite produsul. De obicei, ordinea culorilor critice este dată pe baza unor relații de precedență. De exemplu produsul poate să treacă prin cuva 'A' după ce a trecut prin cuva '2' și acest lucru va fi exprimat printr-o relație de forma „2A”. Dacă în plus, poate să treacă prin cuva 'f' după

ce a trecut prin cuva 'A' vom condensa exprimarea acestor 2 relații de ordine sub forma șirului „2Af”.

Pentru relația de ordine „2Af” dacă pe eticheta produsului cuva '2' va apare după cuva 'A' atunci produsul este compromis.

Managerul atelierului de vopsitorie are probleme uriase după ultima serie de angajări pentru că noii veniți au încurcat seriile de produse și vopsirea materialelor a fost compromisă.

Cerință:

Cunoscând relațiile de precedență care ar fi trebuit respectate în procesul de vopsire trebuie să îl ajutați pe manager să descopere pe bază etichetelor care este starea fiecărui produs – compromis sau nu.

Date de intrare:

Fișierul *eticheta.in* conține pe prima linie un număr natural N, pe a doua linie un șir de caractere O reprezentând relațiile de precedență pentru culorile „critice”, în formă condensată; pe următoarele N linii câte un șir de caractere reprezentând eticheta unui produs.

Date de ieșire:

Fișierul *eticheta.out* conține N linii cu răspunsurile YES sau NO corespunzătoare celor N etichete, în ordinea în care au apărut în fișierul de intrare.

Restricții și precizări:

- numărul de caractere al unei etichete este mai mic decât 256
- caracterele dintr-o etichetă se pot repeta
- caracterele din șirul O sunt distincte
- $1 \leq N \leq 1000$

Exemple:

eticheta.in	eticheta.out	Explicații
2 1d3 st1aweedbvnccx3n st1aweedbvn1ccx3n	YES NO	primul sir: chiar dacă a trecut prin mai multe camere de vopsire a păstrat ordinea relațiilor de precedență - a trecut prin camera 1 înainte de a trece prin camera d, și a trecut prin camera d înainte de a trece prin camera 3, răspuns YES al doilea sir: prin camera 1 nu are voie să treacă după ce trece prin camera d, răspuns NO
1 1d3 st1aweedbccx3nnn3	YES	Nu a încălcat nici o regulă de precedență
1 1d3 staweedbccx3nnn3	YES	Nu a încălcat nici o regulă de precedență

Timp maxim de execuție: 0,1 secunde/test

Memorie totală disponibilă: 2MB și 1 MB pentru stivă

Dimensiunea maximă a sursei: 5 Kb