

Problema 1

100 puncte

Codif

Pentru scrierea mesajelor soldații dintr-o unitate militară folosesc 9 litere mici: **a, e, i, o, u, m, n, r, s** și caracterul spațiu. Aceste litere sunt codificate cu ajutorul cifrelor **1, 2, ..., 9** (în ordinea de mai sus), iar pentru caracterul spațiu se folosește cifra **0**. Astfel codificarea textului **ana are mere** se poate realiza prin numărul natural **171018206282**.

Pentru a mări gradul de securitate a mesajelor transmise soldații reelizează o supracodificare, înlocuind fiecare cifră **k** folosită la codicare cu puterea 2^k . Astfel textul anterior se supracodifică astfel: **2128212256416442564**.

Cerință

Să se scrie un program care pentru o supracodificare dată, determină textul inițial. Dacă există mai multe astfel de texte se vor determina toate.

Date de intrare

Fișierul de intrare `codif.in` conține pe prima linie, **n** numărul de cifre folosite la textul supracodificat, iar pe a doua linie conține supracodificarea.

Date de ieșire

Fișierul de ieșire `codif.out` va conține pe prima linie numărul **m** de texte ce corespund supracodificării date, iar pe următoarele **m** linii, textele, fiecare dintre acestea pe câte o linie, în ordine *lungo-lexicografică*.

Restricții și precizări

- $0 < n < 83$
- Pentru două texte $u = u_1 u_2 \dots u_a$ și $v = v_1 v_2 \dots v_b$, avem că **u** este înaintea lui **v** în ordine *lungo-lexicografică*, dacă $a < b$ sau dacă $a = b$ și există un indice $1 \leq i \leq a$ cu proprietatea că $u_1 = v_1, \dots, u_{i-1} = v_{i-1}$, $u_i < v_i$. Ordinea caracterelor este cea dată de codurile ASCII. De exemplu, în ordine *lungo-lexicografică*, textul **aerss** este înaintea textului **aeumr**, și după textul **sun**.
- Fiecare linie din fișierul de ieșire va avea sfârșit de linie.
- Pentru toate testele numărul de texte care corespunde aceleiași supracodificări este mai mic sau egal cu 70.

Exemplu

<code>codif.in</code>	<code>codif.out</code>
19 2128212256416442564	4 ana are mere ana areoeere a aia are mere a aia areoeere

Tim maxim de executare/test 0.1 secunde

Limita de memorie 16MB

Mărimea maximă a sursei 5 KB

**Tabăra de pregătire a Lotului Național de Informatică
Alba Iulia, 11-17 iunie, 2010**

Baraj 4 Juniori

Sursa: `codif.c`, `codif.cpp`, `codif.pas`

Codif – descrierea soluției

Prof. Doru Popescu Anastasiu, C. N. “Radu Greceanu” - Slatina

Problema se poate rezolva folosind metoda backtracking. Se va genera un vector x cu componente indici ale literelor din alfabet, folosind doi vectori constanti:

$v=(1,2,4,8,16,32,64,128,256,512)$;

$c=(' ','a','e','i','o','u','m','n','r','s')$.

Pentru a usura verificarea conditiilor de continuare se poate utiliza un indice p , pentru pozitia curenta din supracodificare. Cand $p > \text{lungime supracodificare}$, se obtine un text, care se retine intr-un tablou a .

Dupa ordonarea tabloului a , conform relatiei de ordine *lungo-lexicografica* se afiseaza.

Cand se construiește vectorul x , la fiecare pas, incepand cu pozitia p se utilizeaza, un caracter, doua sau trei caractere, formandu-se un string s . Apoi se cauta in v stringul s si se obtine o pozitie ce se va retine in componenta curenta din x .