

## Problema 1 - grad

100 puncte

Avem la dispoziție un text format din litere mici ale alfabetului englez și spații. În text cuvintele sunt separate prin unul sau mai multe spații. Fiecare literă din alfabet are asociat un număr ce reprezintă ordinea literei în alfabet. Astfel **a** are asociat numărul 1, **b** numărul 2, **c** numărul 3 și așa mai departe. Folosind această asociere definim *gradul* unui cuvânt ca fiind suma numerelor asociate fiecărei litere. Spre exemplu cuvântul **bac** are gradul  $2+1+3=6$ .

Folosind cuvintele unui text putem forma grupe de cuvinte. Două cuvinte fac parte din aceeași grupă, dacă au același *grad*.

### Cerință

Să se scrie un program care, pentru un text dat determină numărul de cuvinte și numărul de grupe.

### Date de intrare

Fișierul de intrare **grad.in** are pe prima linie un număr natural **n** reprezentând numărul de caractere din text, iar pe linia a doua textul.

### Date de ieșire

Fișierul de ieșire **grad.out** va conține pe prima linie numărul de cuvinte, iar pe linia a doua numărul de grupe.

### Restricții și precizări

- $0 < n < 255$
- Textul conține cel puțin un cuvânt.
- Primul și ultimul caracter din text este literă.
- Pentru determinarea corectă a numărului de cuvinte se obține 30% din punctaj.
- Nu se obțin puncte dacă numărul de cuvinte este incorect.

### Exemplu

<b>grad.in</b>	<b>grad.out</b>	<b>Explicație</b>
20 bac    daca    aaac    bbb	4 2	Textul conține 4 cuvinte și două grupe. Prima grupă este formată din cuvintele <b>bac</b> , <b>aaac</b> și <b>bbb</b> , care au fiecare gradul 6, iar a doua grupă conține doar cuvântul <b>daca</b> , cu gradul 9.

**Timp maxim de execuție/test:** 1 secundă

### Soluție

Pentru a determina numărul de ordine al unei litere în alfabet se poate folosi codul ASCII sau un sir de caractere constant cu toate literele alfabetului (cautând litera în acest sir de caractere, poziția este chiar numărul de ordine). Gradul unui cuvânt se obține sumând numărul de ordine al fiecărei litere.

Se formează un vector cu gradul fiecărui cuvânt:

$g=(g[1],g[2],\dots,g[k])$ .

Nu este nevoie să se rețină fiecare cuvânt.

Se determină numărul de grupe **nrg**, eventual ordonant crescător vectorul **g**.

Se afișează **k** și **nrg**.