

Problema 1-E3 – *Bucket list*

O problemă importantă de procesare digitală a semnalelor este problema clasificării unor valori numerice în *bucket-uri*, cu scopul realizării histogramelor. Aceste *bucket-uri* sunt intervale disjuncte în care se plasează valorile date la intrare. Spre exemplu, dacă avem numere între 0 și 1000, putem defini 3 *bucket-uri*: [0; 333], [334; 666], [667; 1000].

Cerință

Dându-se o secvență de numere pe 64 de biți, se cere să se împartă aceste numere în următoarele *bucket-uri*, numerotate de la 1 la 19:

1. [0; 9]
2. [10; 99]
3. [100; 999]
4. [1000; 9999]
5. [10000; 99999]
6. [100000; 999999]
7. [1000000; 9999999]
8. ...
19. [10000000000000000000; 9999999999999999999]

Date de intrare

Pe prima linie se află un număr întreg n , fără semn, pe 16 biți, reprezentând numărul de valori de pe a doua linie. Pe a doua linie se află o secvență de valori întregi, fără semn, pe 64 de biți, separate prin spațiu.

Date de ieșire

La ieșire, pe fiecare linie se va afișa numărul *bucket-ului* și numărul de valori de acea lungime din secvența de intrare, separate printr-un spațiu. Dacă nu există nicio valoare într-un anumit *bucket*, linia nu se afișează (vezi lungimea 4 în exemplul de mai jos).

ATENȚIE la respectarea cerinței problemei: afișarea rezultatelor trebuie făcută EXACT în modul în care a fost indicat! Cu alte cuvinte, pe stream-ul standard de ieșire nu se va afișa nimic în plus față de cerința problemei; ca urmare a evaluării automate, orice caracter suplimentar afișat, sau o afișare diferită de cea indicată, duc la un rezultat eronat și prin urmare la obținerea calificativului „Respins”.

Restricții și precizări

1. **Atenție:** În funcție de limbajul de programare ales, fișierul ce conține codul trebuie să aibă una din extensiile .c, .cpp, .java, sau .m. Editorul web **nu va adăuga automat** aceste extensii și lipsa lor duce la imposibilitatea de compilare a programului!
2. **Atenție:** Fișierul sursă trebuie numit de candidat sub forma: <nume>.<ext> unde nume este numele de familie al candidatului și extensia este cea aleasă conform punctului anterior. Atenție la restricțiile impuse de limbajul Java legate de numele clasei și numele fișierului!

Exemple

Intrare
13 154 4 325833 43 87364 42 4 64783746 64523247 34 784934109 34 6435879
Ieșire
1 2 2 4 3 1 5 1 6 1 7 1 8 2 9 1
Explicație: Sunt 13 valori de intrare, care apar la intrare pe a doua linie. În <i>bucket</i> -ul 1 (intervalul [0; 9]) sunt două valori (cei doi de 4). În <i>bucket</i> -ul 2 (intervalul [10; 99]) sunt patru valori: 43, 42, 34 și 34. În <i>bucket</i> -ul 3 este o singură valoare: 154. În <i>bucket</i> -ul 4 nu este nicio valoare, deci nu se afișează ș.a.m.d.

Intrare
10 19 10 11 12 13 14 15 16 17 18
Ieșire
2 10
Explicație: Toate valorile de la intrare sunt în intervalul [10; 99] deci se afișează doar <i>bucket</i> -ul 2, cu toate cele 10 valori.

Timp de lucru: 120 de minute