
Average of consecutive subvectors**X12415_en**

Given a vector $v[0], v[1], \dots, v[n-1]$ of integer values, the average of the segment starting at position i and of size k is $(v[i] + v[i+1] + \dots + v[i+k-1])/k$. Note that i and k must satisfy $0 \leq i \leq n-k$ and $0 < k$ in order to have a correct definition. For example, for the vector $v = [3, 2, 5, 1, 4]$, the average of the segment starting at position 1 of size 3 (i.e. the average of the subvector $[2, 5, 1]$) is $(2 + 5 + 1)/3 = 2.6667$ written with four digits after the decimal point.

Write a function `print_average_of_segments(v, k)` that, given a vector v of integers and a positive natural number k , prints in the standard output (`cout`) the average of all segments of v of size k from left to right. Note that for the above example of v and $k = 3$ we have the following consecutive subvectors (segments) of size k : $[3, 2, 5]$, $[2, 5, 1]$, $[5, 1, 4]$. Thus, in this case the function must print `3.33332.66673.3333`.

Think of a solution that avoids unnecessary computations. Solutions that do not take this into account will likely be rejected by the Jutge with a TLE error ("time limit exceeded") when n and k are large numbers.

Use the following C++ code, completing the missing parts. Recall that the operator `/` makes integer division if both of its operands are integers.

```
#include <iostream>
#include <vector>
using namespace std;

// reads and returns the contents of a vector of n reals from cin, n > 0
vector<int> read_vector(int n) {
    vector<int> v(n);
    for (int i = 0; i < n; ++i)
        cin >> v[i];
    return v;
}

// add here functions or procedures if you need them

// see the stament of the problem
void print_average_of_segments(const vector<int>& v, int k) {
    ...
}

int main() {
    cout.setf(ios::fixed);
    cout.precision(4);
    int n, k;
    while (cin >> n >> k) {
        vector<int> v = read_vector(n);
        print_average_of_segments(v, k);
    }
}
```

=====

CASTELLANO

=====

Dado un vector $v[0], v[1], \dots, v[n-1]$ de enteros, la media del segmento de tamaño k que comienza en la posición i es $(v[i] + v[i+1] + \dots + v[i+k-1])/k$. Observa que i y k deben satisfacer $0 \leq i \leq n-k$ y $k > 0$ para que la definición sea correcta. Por ejemplo, para el vector $v = [3, 2, 5, 1, 4]$, la media del segmento de tamaño 3 comenzando en la posición 1 (es decir, la media del subvector $[2, 5, 1]$) es $(2 + 5 + 1)/3 = 2.6667$ cuando se escribe con cuatro dígitos tras el punto decimal.

Escribe un procedimiento `print_average_of_segments(v, k)` que, dado un vector v de enteros y un entero positivo k , imprime en la salida estándar (`cout`) la media de todos los segmentos de tamaño k en v , de izquierda a derecha. En el ejemplo anterior de vector con $k = 3$ tenemos los siguientes subvectores (segmentos) de tamaño k : $[3, 2, 5]$, $[2, 5, 1]$, $[5, 1, 4]$. Por lo tanto en este caso el procedimiento imprime `3.3333 2.6667 3.3333`.

Piensa una solución que evite cálculos innecesarios. Las soluciones que no tengan esta observación en cuenta tendrán muy probablemente un veredicto de error TLE ("time limit exceeded") del Jutge cuando n y k son números grandes.

Usa el código C++ dado más arriba, completando las partes que faltan. Recuerda que el operador `/` hace la división entera si sus dos operandos son ambos enteros.

=====

CATALÀ

=====

Donat un vector $v[0], v[1], \dots, v[n-1]$ d'enters, la mitjana del segment de mida k que comença a la posició i és $(v[i] + v[i+1] + \dots + v[i+k-1])/k$. Observa que i i k han de satisfer $0 \leq i \leq n-k$ i $k > 0$ perquè la definició sigui correcta. Per exemple, per al vector $v = [3, 2, 5, 1, 4]$, la mitjana del segment de mida 3 començant en la posició 1 (és a dir, la mitjana del subvector $[2, 5, 1]$) és $(2 + 5 + 1)/3 = 2.6667$ quan s'escriu amb quatre dígits després del punt decimal.

Escriu un procediment `print_average_of_segments(v, k)` que, donat un vector v d'enters i un enter positiu k , imprimeix a la sortida estàndard (`cout`) la mitjana de tots els segments de mida k a v , d'esquerra a dreta. En l'exemple anterior de vector amb $k = 3$ tenim els següents subvectores (segments) de mida k : $[3, 2, 5]$, $[2, 5, 1]$, $[5, 1, 4]$. Per tant en aquest cas el procediment imprimeix `3.3333 2.6667 3.3333`.

Pensa una solució que eviti càlculs innecessaris. Les solucions que no tinguin aquesta observació en compte tindran molt probablement un veredict de d'error TLE ("time limit exceeded") del Jutge quan n i k són nombres grans.

Utilitza el codi C++ donat més amunt, completant les parts que falten. Recorda que l'operador `/` fa la divisió entera si els seus dos operands són tots dos enters.

Exam score: 2.5 Automatic part: 30%

Input

The input consists of several cases. Each case has a first line with two integers n, k such that $0 < k \leq n$, and a second line with n integer numbers v_0, \dots, v_{n-1} .

=====

CASTELLANO

=====

La entrada consiste en varios casos. Cada caso comienza con una línea con dos enteros n y k tales que $0 < k \leq n$, seguida de una segunda línea con n números enteros v_0, \dots, v_{n-1} .

=====
CATALÀ

=====
L'entrada consisteix en diversos casos. Cada cas comença amb una línia amb dos enters n i k tals que $a0 < k \leq n$, seguida d'una segona línia amb n nombres enters v_0, \dots, v_{n-1} .

Output

For each case, the program outputs the average of all segments of v_0, \dots, v_{n-1} of size k from left to right, followed by a line break. The averages are printed with four digits of precision and each two consecutive values are separated by a blank space.

=====
CASTELLANO

=====
Para cada caso, el programa imprime la media de todos los segmentos de tamaño k del vector $[v_0, \dots, v_{n-1}]$, de izquierda a derecha, finalizando con un salto de línea. Las medias se imprimen con cuatro dígitos de precisión y cada dos valores consecutivos se separan mediante un espacio en blanco.

=====
CATALÀ

=====
Per a cada cas, el programa imprimeix la mitjana de tots els segments de mida k del vector $[v_0, \dots, v_{n-1}]$, d'esquerra a dreta, finalitzant amb un salt de línia. Les mitjanes s'imprimeixen amb quatre dígitos de precisió i cada dos valors consecutius es separen mitjançant un espai en blanc.

Sample input

```
5 3
3 2 5 1 4
10 3
1 2 3 4 5 6 7 8 9 10
10 5
2 3 4 3 4 2 1 1 0 0
8 3
-1 -2 -2 -1 -1 0 0 1
3 3
1 1 1
6 2
1 -1 1 -1 1 -1
5 1
1 2 3 4 5
```

Sample output

```
3.3333 2.6667 3.3333
2.0000 3.0000 4.0000 5.0000 6.0000 7.0000 8.0000 9.0000
3.2000 3.2000 2.8000 2.2000 1.6000 0.8000
-1.6667 -1.6667 -1.3333 -0.6667 -0.3333 0.3333
1.0000
0.0000 0.0000 0.0000 0.0000 0.0000
1.0000 2.0000 3.0000 4.0000 5.0000
```

Problem information

Author : PRO1

Generation : 2016-05-08 14:58:17

© Jutge.org, 2006–2016.

<http://www.jutge.org>