'Cajas de dulces'

|  |  |  |  |
| --- | --- | --- | --- |
| Puntos | 11.44 | Límite de memoria | 32MB |
| Límite de tiempo (caso) | 1s | Límite de tiempo (total) | 60s |
| Límite de entrada (bytes) | 10 KiB |  |  |

**Descripción**

Carlitos fue a la tienda con su mama a comprar el mandado, y vio un estante con NN cajas de dulces, entonces Carlitos le pide a su mama que si le puede comprarle algunas cajas, su mama dijo que si pero que solo agarrara KK cajas y que si agarraba una caja tenia que agarrar las que estában junto para que no se tarde en ver cual escoger, Carlitos acepta el trato y va a agarrarlos, como su mama solo le permitio llevarse KK cajas y que solo se podia llevar cajas que estuvieran juntas, entonces decide llevarse lo maximo de dulces posible, pero no sabe cuanto es el maximo de dulces que se puede llevar, entonces decides ayudarlo para saber la suma total de dulces que Carlitos se puede llevar.

**Entrada**

Recibirás 2 números NN y KK, que NN representa el numero de cajas que hay en el estante y KK las cajas que puede agarrar. En la siguiente linea seguirán los NiNi números que es el numero de dulces que contiene cada caja

**Salida**

Un numero que represente la suma maxima de dulces que puede llevarse Carlitos.

**Ejemplo**

| **Entrada** | **Salida** | **Descripción** |
| --- | --- | --- |
| 10 4  3 9 9 1 9 10 8 7 2 16 | 34 | Si agarramos las cajas que estan en medio, en especial las que contienen "9,10,8,7", obtendremos la suma maxima de dulces que carlitos se puede llevar que en este caso son 34 |
| 5 2  9 5 2 10 3 | 14 |  |

**Límites**

* 1<=N<=10001<=N<=1000
* 1<=K<=N1<=K<=N
* 1<=Ni<=5001<=Ni<=500

*Fuente: SS*

Problema subido por: [Snoopy](https://omegaup.com/profile/Avela64/)

[Reportar contenido inapropiado en este problema.](https://omegaup.com/arena/problem/Cajas-de-dulces)

<https://omegaup.com/arena/problem/Cajas-de-dulces#problems>

#include <iostream>

#include <stdio.h>

#include <set>

#include <map>

#include <vector>

#include <algorithm>

using namespace std;

int main() {

int N, K;

scanf("%d%d", &N, &K);

int arr[N];

for(int i =0; i<N; i++) {

int elem;

scanf("%d", &elem);

arr[i] = elem;

}

int sum = 0;

int i;

for (i = 0; i < K; i++)

{

sum += arr[i];

}

int max\_sum = sum;

for (i =0 ;i + K < N; i++)

{

sum -= arr[i];

sum += arr[i + K];

max\_sum = max(max\_sum, sum);

}

cout << (max\_sum) << endl;

return 0;

}