

Maxi-server (ricostruzioni)

Limite di tempo: 1.0 secondi
Limite di memoria: 256 MiB

Ora che le gare a squadre di informatica stanno coinvolgendo sempre più la nazione, il quartier generale delle Olimpiadi ha bisogno di costruire un nuovo maxi-server in modo da gestire efficientemente la competizione. Questo maxi-server dovrà essere lungo K metri, e dovrà essere collocato in un punto segreto in cui però la corrente elettrica sia facilmente disponibile. Per questo motivo William ha selezionato, tra tutte le linee elettriche italiane, la linea $A/C51$ che attraversa le alpi da una parte all'altra garantendo la giusta dose di riservatezza. Ora rimane solo da individuare il punto esatto in cui costruire il maxi-server!

A complicare la scelta, tuttavia, rimane il fatto che la linea $A/C51$ attraversa zone con altitudini molto diverse, e che per giunta cambiano molto rapidamente, mentre il maxi-server andrà costruito necessariamente in piano. William sa che nel metro i della linea (lunga complessivamente N metri) l'altitudine sul livello del mare è di A_i metri. Sa inoltre che una volta scelto un intervallo di K metri all'interno della linea, il costo per spianare quell'intervallo sarà proporzionale alla *massima differenza di altitudine* presente in quell'intervallo, vale a dire la differenza tra l'altitudine più alta e quella più bassa.

Aiuta William a trovare il punto più pianeggiante della linea $A/C51$!

Implementazione

Dovrai sottoporre esattamente un file con estensione `.c`, `.cpp` o `.pas`.

📎 Tra gli allegati a questo task troverai un template (`ricostruzioni.c`, `ricostruzioni.cpp`, `ricostruzioni.pas`) con un esempio di implementazione da completare.

Se sceglierai di utilizzare il template, dovrai implementare la seguente funzione:

C/C++	<code>int spiana(int N, int K, int A[]);</code>
Pascal	<code>function spiana(N, K: longint; var A: array of longint): longint;</code>

In cui:

- L'intero N rappresenta la lunghezza totale in metri della linea $A/C51$.
- L'intero K rappresenta la lunghezza in metri del maxi-server da costruire.
- L'array A , indicizzato da 0 a $N - 1$, contiene le altitudini dei vari metri della linea.
- La funzione dovrà restituire la minima differenza di altitudine per un intervallo di K metri, che verrà stampata sul file di output.

Dati di input

Il file `input.txt` è composto da due righe. La prima riga contiene i due interi N e K . La seconda riga contiene gli N interi A_i separati da uno spazio.

Dati di output

Il file `output.txt` è composto da un'unica riga contenente un unico intero, la risposta a questo problema.

Assunzioni

- $1 \leq K \leq N \leq 100\,000$.
- $0 \leq A_i \leq 1\,000\,000$ per ogni $i = 0 \dots N - 1$.

Assegnazione del punteggio

Il tuo programma verrà testato su diversi test case raggruppati in subtask. Per ottenere il punteggio relativo ad un subtask, è necessario risolvere correttamente tutti i test relativi ad esso.

- **Subtask 1 [10 punti]:** Casi d'esempio.
- **Subtask 2 [30 punti]:** $N \leq 100$.
- **Subtask 3 [20 punti]:** $K \leq 100$.
- **Subtask 4 [20 punti]:** $A_i \leq 100$ per ogni $i = 0 \dots N - 1$.
- **Subtask 5 [20 punti]:** Nessuna limitazione specifica.

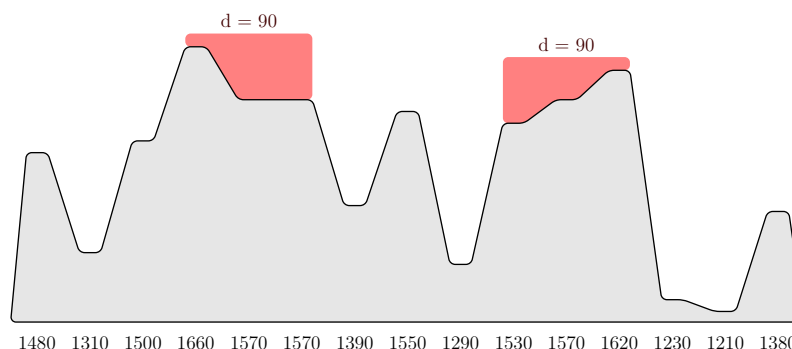
Esempi di input/output

Esempi di input/output

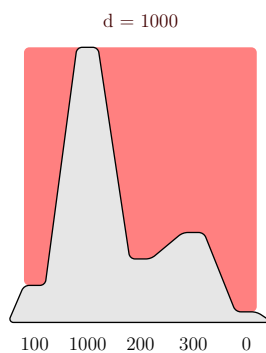
input.txt	output.txt
15 3 1480 1310 1500 1660 1570 1570 1390 1550 1290 1530 1570 1620 1230 1210 1380	90
5 5 100 1000 200 300 0	1000
10 5 6870 7829 8708 6060 1918 1956 5344 3666 3850 5899	3426

Spiegazione

Nel **primo caso di esempio**, si può ottenere una differenza di 90 sia selezionando le altitudini 1660 – 1570 – 1570 che le altitudini 1530 – 1570 – 1620.



Nel **secondo caso di esempio**, siamo costretti a selezionare tutta quanta la lunghezza della linea per una differenza di altitudine di 1000.



Nel **terzo caso di esempio**, si può ottenere una differenza di 3426 selezionando le altitudini 1918 – 1956 – 5344 – 3666 – 3850.

