

Tornei di scherma (scherma)

Kenan, amico di Noemi, è un fortissimo schermidore¹. Talmente forte, da essere sicuro di vincere qualsiasi torneo. In particolare, nel prossimo mese ci saranno N tornei diversi, ognuno con un premio al vincitore di P_i euro.



Figure 1: Kenan durante un suo scontro.

Tuttavia, Kenan ha un livello E di energia, ed ogni torneo gli consumerà L_i punti di energia. Nel caso l'energia rimasta fosse inferiore di quella richiesta da un torneo, Kenan non potrà prenderne parte. Trova quanti soldi può vincere Kenan se sceglie in modo ottimale i tornei a cui partecipare.

Implementazione

Dovrai sottoporre un unico file, con estensione `.cpp`.

📎 Tra gli allegati a questo task troverai un template `scherma.cpp` con un esempio di implementazione.

Il file di input è composto da $N + 1$ righe:

- Riga 1: gli interi N e E , separati da uno spazio.
- Riga 2 ... $N+1$: gli interi P_i e L_i .

Il file di output è composto da 1 riga:

- Riga 1: la risposta al problema.

Assunzioni

- $1 \leq N \leq 1\,000\,000$.
- $1 \leq E \leq 1\,000\,000$.

¹Giocatore di scherma.

- $1 \leq N \cdot E \leq 3\,000\,000$.
- $1 \leq P_i \leq 1\,000\,000\,000$ per ogni $i = 0 \dots N - 1$.
- $1 \leq L_i \leq E$ per ogni $i = 0 \dots N - 1$.
- Kenan può partecipare ad ogni torneo una volta sola.

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 che lo compongono.

- **Subtask 1** [0 punti]: Casi d'esempio.
- **Subtask 2** [10 punti]: $N \cdot E \leq 10$.
- **Subtask 3** [10 punti]: L_i è uguale per tutti gli $i = 0 \dots N - 1$.
- **Subtask 4** [13 punti]: $N \leq 20$.
- **Subtask 5** [27 punti]: $N \cdot E \leq 1\,000$.
- **Subtask 6** [40 punti]: Nessuna limitazione aggiuntiva.

Esempi di input/output

stdin	stdout
3 20 10 7 13 9 6 4	29

Spiegazione

Nel caso d'esempio a Kenan conviene partecipare a tutti e tre i tornei, ottenendo in totale 29 euro.