

ABC 2014

Bug (bug)

Descrizione del problema

Il professor Tarboto sta per terminare di scrivere il suo nuovo libro di reti e sistemi informatici, ma si è accorto di essere un po' indietro con il lavoro: la fine dell'anno scolastico si sta avvicinando e la casa editrice vuole che il lavoro sia assolutamente pronto entro giugno.

In particolare, restano da sistemare ancora **M** bug nei codici degli algoritmi presenti nel libro.

Grazie alla sua proverbiale esperienza, il professor Tarboto decide quindi di chiedere la collaborazione dei suoi **N** studenti affinché risolvano i problemi al suo posto. Ovviamente, gli studenti **possono lavorare in parallelo**.

Gli studenti del professore però, saputo dell'urgenza con cui il loro docente deve finire il lavoro, chiedono in cambio per l'essere impiegati (**indipendentemente dal volume di lavoro**, ovvero da quanti bug devono risolvere) un certo numero di punti (**P**) in più nella prossima verifica di Sistemi e Reti.

Il professor Tarboto è disposto a fare questa concessione purché il **totale** dei punti aggiuntivi che dovrà regalare ai suoi studenti non ecceda la somma **S** (non vuole che tutti prendano 10 alla prossima verifica... il preside potrebbe insospettirsi!).

Il professore sa che i bug, come gli studenti, non sono tutti uguali: ogni bug è caratterizzato dall'avere una complessità (**C**) e può essere risolto **in un giorno** (indipendentemente dalla complessità) da uno studente con un'abilità (**A**) tale che $A \geq C$.

Aiuta il professor Tarboto a stabilire a quale studente assegnare ciascun bug in modo da risolverli tutti **nel minor numero di giorni** possibile.

Dati di input

L'input è composto da 4 righe.

La prima riga contiene tre interi: **N**, **M**, **S**; rispettivamente il numero degli studenti, il numero dei bug e la massima quantità di punti che Tarboto può distribuire alla prossima verifica.

La seconda riga contiene **M** interi separati da uno spazio: C_i è la complessità dell'*i*-esimo bug.

La terza riga contiene **N** interi separati da uno spazio: A_i è l'abilità dell'*i*-esimo studente.

La quarta riga contiene **N** interi separati da uno spazio: P_i è la quantità di punti che l'*i*-esimo studente chiede in cambio del proprio lavoro.

Dati di output

La prima riga dell'output deve contenere la stringa "SI" (senza virgolette e senza accento) oppure "NO" (senza virgolette), a seconda se gli studenti del professor Tarboto riusciranno a risolvere tutti i problemi.

Se la risposta è "SI", l'output deve contenere una seconda riga costituita da **M** interi separati da spazio. L'*i*-esimo di questi numeri deve essere il numero dello studente che corregge l'*i*-esimo bug nella soluzione ottimale.

Se ci sono più soluzioni ottimali, è sufficiente stamparne una qualsiasi.

Assunzioni

- $1 \leq N, M \leq 10^5$
- $0 \leq S \leq 10^9$
- $1 \leq C_i, A_i \leq 10^9$
- $0 \leq P_i \leq 10^9$

Nel 50% dei casi vale che $N, M, S, A_i, C_i, P_i \leq 100$.

Nota importante per la valutazione

Soluzioni che stampano **solo** la prima riga di output ("SI" o "NO") verranno valutate con il 30% dei punti per ogni caso di prova.

Esempi di input/output

Input (stdin)	Output (stdout)
3 4 9 1 3 1 2 2 1 3 4 3 6	SI 2 3 2 3

Chiarificazione dell'esempio

Il terzo studente (con abilità 3) risolve il secondo e il quarto bug (con complessità 3 e 2, rispettivamente) e il secondo studente (con abilità 1) risolve il primo e il terzo bug (le loro complessità sono 1).

Risolvere ciascun bug richiede un giorno per ciascuno studente, quindi i giorni totali sono 2 (gli studenti possono lavorare in parallelo).

Il secondo studente vuole 3 punti, il terzo 6 punti e ciò incontra la disponibilità del professore che è pronto a distribuire al massimo 9 punti.