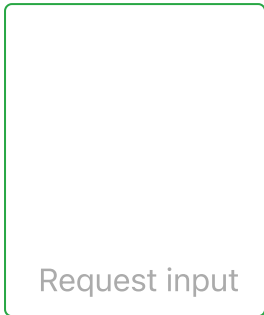


Mangiata di carote



Scarica la traccia nel linguaggio che preferisci, poi modifica il programma aggiungendo le parti che mancano!

- Scarica la traccia in Scratch: [monologo.sb3](#)
- Scarica la traccia in pseudocodice: [monologo.srs](#)
- Scarica la traccia in C: [monologo.c](#)
- Scarica la traccia in C++: [monologo.cpp](#)
- Scarica la traccia in Python: [monologo.py](#)
- Scarica la traccia in Java: [monologo.java](#)
- Scarica la traccia in C#: [monologo.cs](#)
- Scarica la traccia in JavaScript: [monologo.html](#)

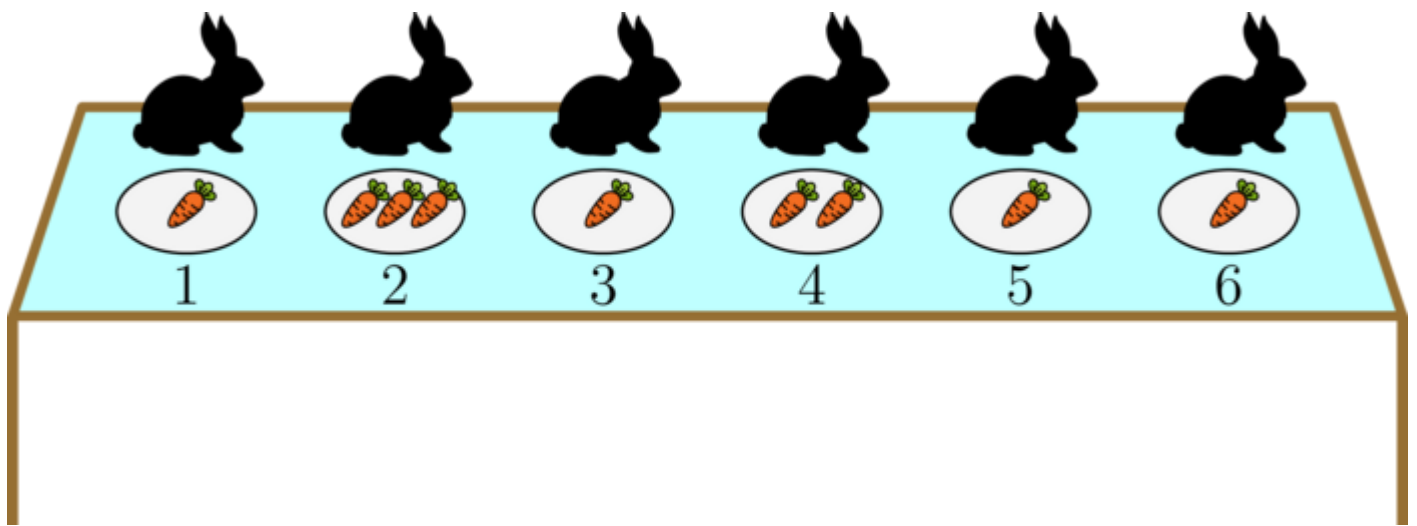
Bunny ha organizzato una mangiata di carote! I suoi N amici sono tutti in fila in una lunga tavolata, con i posti numerati da 1 a N , e il coniglio nel posto numero i ha C_i carote nel piatto.

Tutti aspettano con educazione il via di Bunny prima di mangiare le proprie carote: purtroppo Bunny non ha una voce molto forte, quindi quando dice a tutti di cominciare solo i conigli più vicini lo sentono!

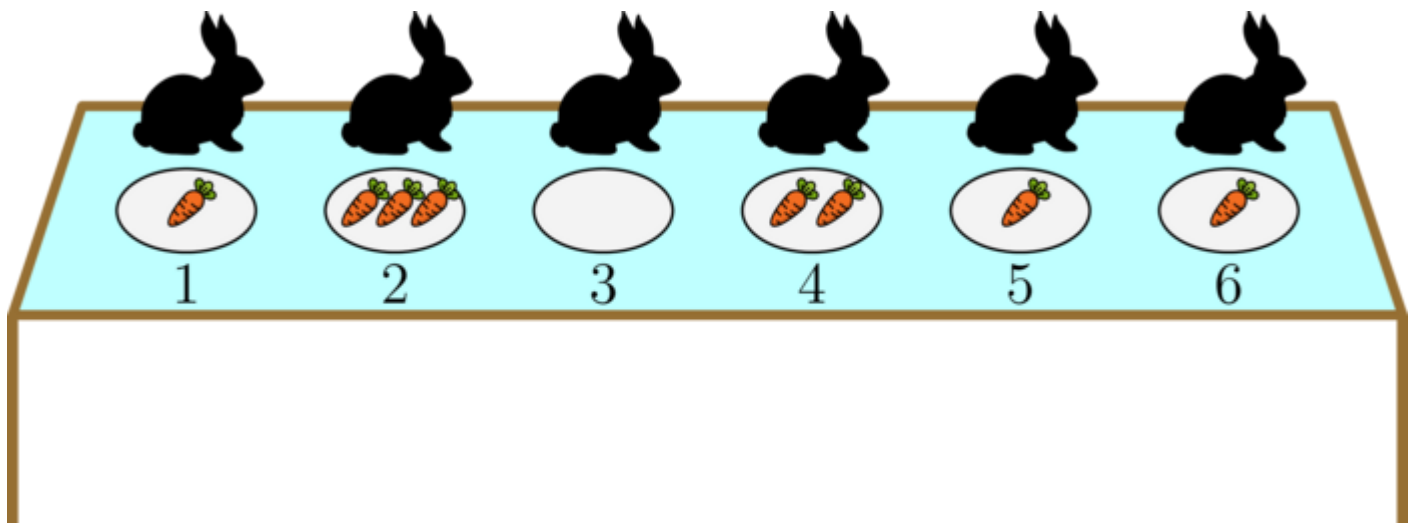
Dopo un po' Bunny controlla, e nota che ora il coniglietto nel posto numero i ha F_i carote nel piatto, meno o uguale a quante ne aveva prima. Sapendo che tutti i conigli che lo hanno sentito hanno mangiato almeno una carota, mentre gli altri non ne hanno ancora mangiate, quanti sono i conigli che lo hanno sentito?

Esempi

Come **primo esempio**, gli amici di Bunny potrebbero all'inizio avere queste carote:

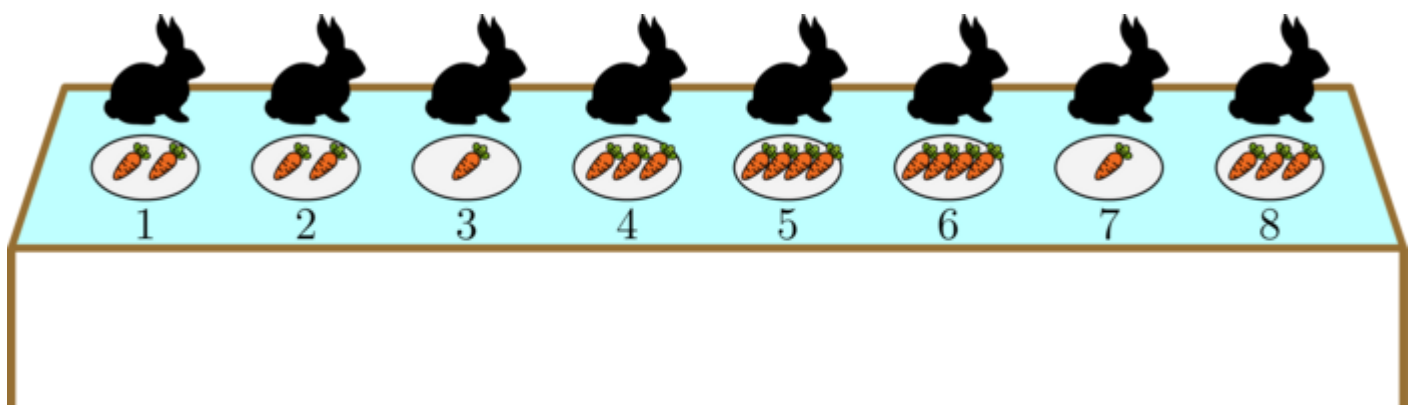


Quanto Bunny controlla, potrebbero invece avere queste carote:

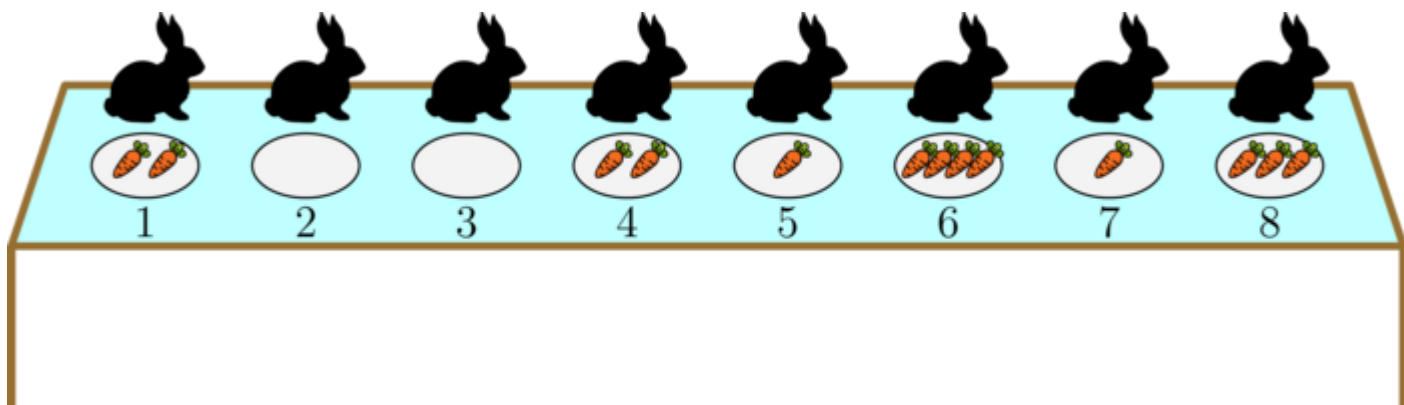


Allora solo il coniglio nel posto 3 ha sentito Bunny!

Come **secondo esempio**, gli amici di Bunny potrebbero all'inizio avere queste carote:



Quanto Bunny controlla, potrebbero invece avere queste carote:



Allora hanno sentito Bunny in quattro: i conigli nei posti 2, 3, 4, 5.

Dettagli tecnici

Limiti numerici

Nel file di input che scaricherai saranno presenti esattamente 10 casi da risolvere. In tutti sarà vero che:

- $1 \leq N \leq 100$
- Le carote presenti alla fine sono meno che all'inizio: $0 \leq F_i \leq C_i \leq 10$
- Almeno un coniglio mangia almeno una carota

Formato di input

La prima riga del file di input contiene il numero T di casi da risolvere. Seguono T casi da risolvere, numerati da 1 a T , separati da una riga vuota. Ogni caso da risolvere è descritto da tre righe:

- la prima riga contiene il numero N ,
- la seconda riga contiene la lista di N numeri C_i ,
- la terza riga contiene la lista di N numeri F_i .

Per esempio, i due casi descritti prima si rappresentano così:

Input:

2

6

1 3 1 2 1 1

1 3 0 2 1 1

8

2 2 1 3 4 4 1 3

2 0 0 2 1 4 1 3

Formato di output

Il file di output deve contenere la risposta ai casi che sei riuscito a risolvere. Per ogni caso che hai risolto, il file di output deve contenere una riga con scritto:

Case #t: x

dove t è il numero del caso da risolvere (a partire da 1) e il valore x è il numero di conigli che hanno sentito Bunny. Per esempio, le risposte ai due casi descritti prima si indicano così:

Output:

Case #1: 1

Case #2: 4