

## Accensione dei PC (accensione)

Limite di tempo: 1.0 secondi

Limite di memoria: 256 MiB

Alle OII, gli  $N$  computer dei partecipanti sono numerati da 1 a  $N$ . Al momento solo alcuni di essi sono accesi, e Gabriele deve accenderli tutti prima dell'inizio della gara!

Sfortunatamente, l'accensione e lo spegnimento dei computer può avvenire solamente attraverso un sofisticato sistema di pulsanti. Nella Sala Controllo vi sono infatti  $N$  pulsanti, anch'essi numerati da 1 a  $N$ . Il pulsante  $K$  ha, come effetto, quello di cambiare lo stato di tutti i computer il cui numero identificativo sia un divisore di  $K$ . Ad esempio, il pulsante 12 cambia lo stato dei computer 1, 2, 3, 4, 6, 12.

Ogni pulsante può essere premuto al massimo una volta, ed è necessario premere i pulsanti in ordine crescente (non si può premere il pulsante  $K_1$  dopo aver premuto il pulsante  $K_2$ , se  $K_1 < K_2$ ). Un computer può venire acceso e/o spento anche varie volte; l'importante è che alla fine tutti i computer risultino accesi. Sapendo quali sono i computer inizialmente accesi, dovete decidere quali pulsanti premere in modo da accenderli tutti.

Ad esempio, supponiamo che i computer siano  $N = 6$ . Ci sono quindi  $N = 6$  pulsanti, che cambiano lo stato dei computer secondo la tabella seguente.

Pulsante	Effetto
1	Computer 1
2	Computer 1 e 2
3	Computer 1 e 3
4	Computer 1, 2 e 4
5	Computer 1 e 5
6	Computer 1, 2, 3 e 6

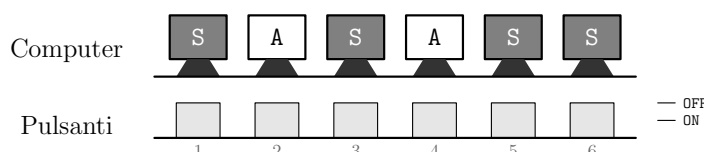
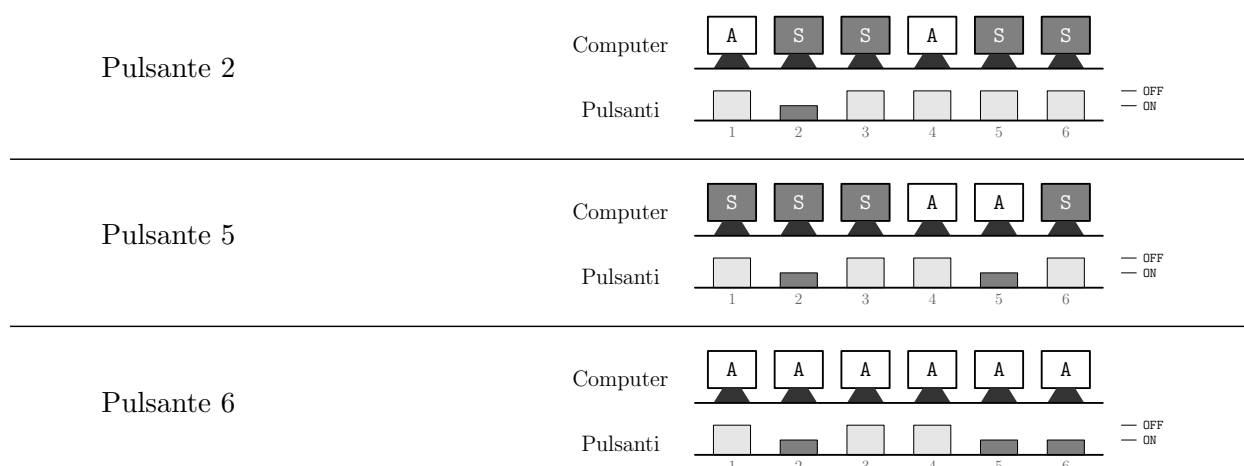


Figura 1: Configurazione iniziale dei computer.

Supponiamo che i computer siano inizialmente nel seguente stato della Figura 1 (dove S significa spento e A significa acceso). In questo caso, per poterli accendere tutti, Gabriele deve premere in sequenza i seguenti pulsanti:



Aiuta Gabriele a determinare la sequenza di pulsanti da premere per poter accendere tutti i computer.

## 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 [5 punti]:** Caso d'esempio.
- **Subtask 2 [10 punti]:**  $N \leq 10$ .
- **Subtask 3 [8 punti]:** I computer spenti all'inizio sono tutti e soli quelli contrassegnati da un numero primo (2, 3, 5, 7...).
- **Subtask 4 [26 punti]:**  $N \leq 1000$ .
- **Subtask 5 [28 punti]:**  $N \leq 100\,000$ .
- **Subtask 6 [23 punti]:** Nessuna limitazione specifica (vedi la sezione **Assunzioni**).

## Implementazione

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

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

Dovrai implementare la seguente funzione:

C/C++	<code>void Accendi(int N, int acceso[], int pulsante[]);</code>
Pascal	<code>procedure Accendi(N: longint; var acceso, pulsante: array of longint);</code>

dove:

- L'intero  $N$  rappresenta il numero di computer.
- L'array `acceso` descrive lo stato iniziale dei computer: se `acceso[i]` vale 1, all'inizio l' $i$ -esimo computer è acceso; se vale 0 è spento.
- La funzione dovrà riempire l'array `pulsante`, inserendo nella posizione  $i$  dell'array il valore 1 se è necessario premere l' $i$ -esimo pulsante, 0 altrimenti. È garantito che inizialmente `pulsante[i]` valga 0 per ogni  $i$ .

**Attenzione:** `acceso[0]` e `pulsante[0]` non contengono informazioni utili (in quanto i computer e i pulsanti sono numerati da 1 a  $N$ ).

## Grader di prova

Nella directory relativa a questo problema è presente una versione semplificata del grader usato durante la correzione, che potete usare per testare le vostre soluzioni in locale. Il grader di esempio legge i dati di input dal file `input.txt`, a quel punto chiama la funzione `Accendi` che dovete implementare. Il grader scrive sul file `output.txt` quali pulsanti sono stati premuti dalla vostra funzione.

Nel caso vogliate generare un input per un test di valutazione, il file `input.txt` deve avere questo formato:

- Riga 1: contiene l'intero  $N$ , il numero di computer.

- Riga 2: contiene  $N$  valori; l' $i$ -esimo di questi vale 1 se l' $i$ -esimo computer è acceso, vale 0 se è spento.

Il file `output.txt` invece ha questo formato:

- Riga 1: contiene  $N$  valori; l' $i$ -esimo di questi vale 1 se l' $i$ -esimo pulsante è stato premuto, vale 0 altrimenti.

## Assunzioni

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

## Esempi di input/output

input.txt	output.txt
6 0 1 0 1 0 0	0 1 0 0 1 1

Questo caso di input corrisponde all'esempio spiegato nel testo.