

Demo003

Il metodo `sumVectors` somma due array di lunghezze diverse posizione per posizione, gestendo elegantemente il caso in cui uno dei due sia più corto dell'altro. Il codice determina prima quale array è più lungo e crea un array risultato di quella dimensione. Poi scorre tutte le posizioni e, per ciascuna, controlla se l'indice è valido per entrambi gli array. Se un array è troppo corto e non ha un elemento in quella posizione, il codice usa semplicemente 0 come valore di default. In questo modo evita errori di accesso fuori range e tratta gli elementi mancanti come zeri nella somma.

Il risultato finale è un array lungo quanto il più lungo dei due input, dove ogni posizione contiene la somma degli elementi corrispondenti, con gli spazi "vuoti" trattati come zeri.

Demo004

Il codice filtra una lista di 7 nomi per estrarre solo quelli con esattamente 6 caratteri. Scorre tutti i nomi uno per uno, controlla la lunghezza di ciascuno con `name.length() == 6`, e aggiunge alla nuova lista `shortNames` solo quelli che soddisfano la condizione. Il risultato finale è un sottoinsieme di 2 nomi ("Simone" e "Andrea") su 7 originali, dimostrando come un filtro riduca la cardinalità mantenendo solo gli elementi che passano il test specificato.