

Programmazione di sistema

Anno accademico 2017-2018

Esercitazione 3

Dato un elenco ordinato di N numeri interi, è possibile generare un nuovo elenco di cardinalità $N-1$ scegliendo una coppia di valori al suo interno e combinandoli tra loro tramite una delle quattro operazioni aritmetiche fondamentali (somma, sottrazione, moltiplicazione e divisione), ammesso che essa risulti possibile: l'elenco prodotto conterrà il risultato dell'operazione e tutti i valori che non sono stati utilizzati, opportunamente riordinati.

Applicando ricorsivamente la stessa funzione sul risultato ottenuto è possibile ridurre l'elenco finale ad un solo numero.

Applicando la funzione ad un insieme di N elementi, il numero delle scelte possibili risulta compreso tra 0 (se nessuna operazione risultasse lecita) e $N \cdot (N-1) \cdot 4$ (se tutte le operazioni fossero lecite).

Ad ogni elenco è anche associato un punteggio. La funzione che trasforma l'elenco, trasforma anche il punteggio secondo la seguente regola: il punteggio risultante è pari alla somma del punteggio originale con il prodotto tra il risultato dell'operazione effettuata e una costante dipendente dall'operatore utilizzato (somma: 2, sottrazione: 1, moltiplicazione: 4, divisione: 3).

I seguenti vincoli sono imposti sulle operazioni:

- le sottrazioni possono essere effettuate solo se il risultato è positivo
- le moltiplicazioni possono essere effettuate solo se nessuno degli operandi è zero
- le divisioni possono essere effettuate solo se nessuno degli operandi è zero e la divisione non produce resto
- le somme sono invece sempre possibili.

Si scriva un programma che generi, in modo efficiente, un file riportante, tra tutte le combinazioni ordinate di 7 numeri interi compresi tra 0 e 9, quelle che permettono di ottenere come risultato finale il valore 0 e, per ciascuna combinazione, la sequenza ottimale di scelte necessarie ad ottenere il massimo punteggio.

Visto l'ampio spazio di ricerca, si consideri l'utilizzo di tecniche di caching allo scopo di evitare di calcolare lo stesso risultato più volte.

Si proceda in modo ricorsivo, formulando dapprima un algoritmo atto a valutare il caso più semplice (un solo numero) a cui aggiungere la gestione dei casi più complessi (combinazione di due, tre, ... numeri).

Competenze da acquisire

- Uso delle librerie di I/O basate sui flussi
- Modellazione delle informazioni
- Uso delle librerie standard (set, map, iteratori, algoritmi ...)
- Uso delle funzioni lambda