

– Simulazione di Prova Intermedia – 26/01/2021

Questa è una simulazione di prova intermedia.

Nello svolgimento degli esercizi si invitano gli studenti a considerare aspetti di “buona programmazione”, “pulizia del codice” ed efficienza. Ad es.: formattazione corretta del codice, rendere il codice modulare aggiungendo ove necessario altri metodi rispetto a quelli richiesti dall'esercizio, soprattutto se questi rendono il codice più pulito e leggibile, o se evitano duplicazione di codice. Inoltre, non ci devono essere warning nel codice scritto.

Esercizio Java-- n. 1: Riarrangia

Scrivere un metodo Java--, chiamato **riarrangia**, che dato un array a di numeri interi diversi da zero, restituisca un array con gli stessi elementi di a , ma riarrangiati in modo “alternato” così che, partendo da un numero negativo, ogni numero negativo sia seguito da uno positivo e viceversa, mantenendo però l’ordine di apparizione in a . Il numero di elementi positivi e negativi non è necessariamente uguale: se ci sono più numeri positivi, questi devono apparire alla fine dell’array, e, ugualmente, se ci sono più numeri negativi, questi devono apparire alla fine dell’array.

Ad esempio,

- con input l’array {1, 2, 3, -4, -1, -5}, il metodo deve restituire l’array {-4, 1, -1, 2, -5, 3},
- con input l’array {1, 2, 3, -4, -1, 4}, il metodo deve restituire l’array {-4, 1, -1, 2, 3, 4},
- con input l’array {-5, -2, 5, 2, -4, 7, -1, -8, 2, -8}, il metodo deve restituire l’array {-5, 5, -2, 2, -4, 7, -1, 2, -8, -8},
- con input l’array {6, 2, 7, 4, 1, 4}, il metodo deve restituire l’array {6, 2, 7, 4, 1, 4},
- con input l’array {-5, -2, -7, -8}, il metodo deve restituire l’array {-5, -2, -7, -8}.

L’array in input non deve essere modificato ed è possibile usare array di appoggio, se necessario.

Esercizio Java-- n. 2: Conta sottomatrici

Sia T una matrice di dimensione $m \times n$ (con $m > 0$ e $n > 0$) contenente numeri interi, e sia S una matrice di dimensione $p \times q$ (con $0 < p \leq m$ e $0 < q \leq n$) contenente numeri interi.

Scrivere un metodo Java-- chiamato `contaSottomatrici` che, dato in input una matrice T di numeri interi di dimensione $m \times n$ (m righe e n colonne), con $m > 0$ e $n > 0$, e una matrice S di numeri interi di dimensione $p \times q$ (p righe e q colonne), con $0 < p \leq m$ e $0 < q \leq n$, restituisca il numero di occorrenze della matrice S all'interno della matrice T .

Ad esempio, sia $T = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 5 & 6 \\ 1 & 2 & 3 & 4 \end{pmatrix}$:

- se $S = (1)$ il metodo deve restituire 3 occorrenze: $\begin{pmatrix} \textcircled{1} & 2 & 3 & 4 \\ \textcircled{1} & 2 & 5 & 6 \\ \textcircled{1} & 2 & 3 & 4 \end{pmatrix}$
- se $S = \begin{pmatrix} 1 & 2 \\ 1 & 2 \end{pmatrix}$ il metodo deve restituire 2 occorrenze: $\begin{pmatrix} \boxed{1} & \boxed{2} & 3 & 4 \\ \boxed{1} & \boxed{2} & 5 & 6 \\ \boxed{1} & \boxed{2} & 3 & 4 \end{pmatrix}$
- se $S = (1 \ 2 \ 3 \ 4)$ il metodo deve restituire 2 occorrenze: $\begin{pmatrix} \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} \\ 1 & 2 & 5 & 6 \\ \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} \end{pmatrix}$
- se $S = \begin{pmatrix} 5 \\ 3 \end{pmatrix}$ il metodo deve restituire 1 occorrenza: $\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & \boxed{5} & 6 \\ 1 & 2 & \boxed{3} & 4 \end{pmatrix}$
- se $S = T$ il metodo deve restituire 1 occorrenza: $\boxed{\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 5 & 6 \\ 1 & 2 & 3 & 4 \end{pmatrix}}$

Sia $T = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$ e $S = (1 \ 1)$, il metodo deve restituire 6 occorrenze:

$$\begin{pmatrix} \boxed{1} & \boxed{1} & 1 \\ \boxed{1} & \boxed{1} & 1 \\ \boxed{1} & \boxed{1} & 1 \end{pmatrix}$$