Esercizio 1

Data la classe Array, dotata di inserimento/rimozione/svuotamento array e relativa stampa degli elementi:

* scrivere un metodo che restituisca la somma di tutti gli elementi dell’array
* scrivere un metodo che verifichi se un dato elemento è presente nell’array, restituendo *true* se presente, altrimenti *false*

Esercizio 2

Dato l’array dell’esercizio precedente, scrivere l’ordinamento tramite insertion sort (inserimento per ordinamento).

1. Inizializzazione:

* Considera l'array non ordinato.
* Definisci un indice **i** che inizia da 1 (considerando il primo elemento già "ordinato").

2. Iterazione sull'Array:

* Itera attraverso l'array da sinistra a destra, partendo dall'elemento all'indice **i**.
* Assegna il valore dell'elemento corrente a una variabile temporanea.

3. Confronto e Spostamento:

* Mentre l'indice corrente **j** è maggiore di 0 e l'elemento corrente è minore del valore temporaneo:
  + Sposta l'elemento a sinistra (elemento corrente diventa **array[j]**).
  + Riduci **j** di 1.

4. Inserimento:

* Assegna il valore temporaneo all'elemento successivo a **array[j]**.

5. Incremento dell'Indice:

* Incrementa l'indice **i** per passare all'elemento successivo.

6. Ripetizione:

* Ripeti i passi 2-5 fino a quando hai esaminato tutti gli elementi nell'array.

Esercizio 3

Dato l’array dell’esercizio precedente, scrivere l’ordinamento tramite bubble sort.

1. Inizializzazione:
   * Considera l'array non ordinato.
   * Ripeti il processo finché nessuno scambio è stato effettuato durante un'iterazione completa dell'array.
2. Iterazione sull'Array:
   * Itera attraverso l'array da sinistra a destra.
   * Confronta coppie di elementi adiacenti.
   * Se l'elemento successivo è minore di quello corrente, effettua uno swap.
3. Ripetizione:
   * Continua l'iterazione sull'array finché nessuno scambio è stato effettuato durante un'iterazione completa.
4. Array Ordinato:
   * L'array è ordinato quando non sono necessari ulteriori scambi.

Esercizio 4

Dato l’array dell’esercizio precedente, scrivere l’ordinamento tramite Shell Sort.

1. Inizializzazione:
   * Considera l'array non ordinato.
   * Definisci un intervallo iniziale.
2. Suddivisione e Ordinamento:
   * Dividi l'array in sotto-array di dimensioni definite dall'intervallo.
   * Ordina ciascun sotto-array utilizzando l'ordinamento per inserimento.
3. Riduzione dell'Intervallo:
   * Riduci l'intervallo e ripeti il processo di suddivisione e ordinamento.
4. Conclusione:
   * L'array è ora ordinato.