**Insertion Sort (Sortieren durch Einfügen)**

Der SelectionSort-Algorithmus ist ein Sortierverfahren, das durch Einfügen von Zahlen in den bereits sortierten Teil des Arrays funktioniert. Das Verfahren wird z.B. hier erklärt: <https://youtu.be/OGzPmgsI-pQ>

**Aufgabe 1: Beschreibung des Verfahrens**

Ergänzen Sie die Beschreibungen zu dem Insertion Sort-Algorithmus.

|  |  |
| --- | --- |
| Wiederhole, solange noch nicht alle Zahlen einsortiert (grün) sind … | |
| Ein Bild, das Reihe, Screenshot, Zahl, Rechteck enthält.  Automatisch generierte Beschreibung | Markiere die nächste noch nicht einsortierte zahl zum Einsortieren (gelb). |
| Ein Bild, das Reihe, Screenshot, Zahl enthält.  Automatisch generierte Beschreibung | Finde die richtige Stelle |
| Ein Bild, das Zahl, Reihe enthält.  Automatisch generierte Beschreibung | Verschiebe |
| Ein Bild, das Screenshot, Reihe, Rechteck enthält.  Automatisch generierte Beschreibung | Färbe |

**Aufgabe 2: Sortieren**

Führe das Sortierverfahren für das Array zahlen=[13,71,6,1,15,81,100,66,7,48] durch. Markieren Sie die Zahlen analog zu der obigen Beschreibung.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **zahlen[0]** | **zahlen[1]** | **zahlen[2]** | **zahlen[3]** | **zahlen[4]** | **zahlen[4]** | **zahlen[5]** | **zahlen[6]** | **zahlen[7]** | **zahlen[8]** |
| 13 | 71 | 6 | 1 | 15 | 81 | 100 | 66 | 7 | 48 |
| 13 | 71 | 6 | 1 | 15 | 81 | 100 | 66 | 7 | 48 |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

**Aufgabe 3: Der Inserttion Sort-Algorithmus**

Fügen Sie die Pseudocode-Schnipsel auf der rechten Seite so in das Struktogramm ein, dass das Struktogramm den korrekten Algorithmus repräsentiert.

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte Beschreibung

**Aufgabe 4:** **Durchlaufen des Algorithmus**

Führen Sie einen Schreibtischtest für den Insertion Sort-Algorithmus durch.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Eingabe: [5, 3, 4, 2]** | | | | | | |
| i | sortierZahl | einfuegePos | while-Durchlauf | zahlen[einfuegePos + 1] | einfuegePos | zahlen[einfuegePos +1] |
| 1 |  |  | 1 |  |  |  |
| 2 |  |  | 1 |  |  |  |
| 3 |  |  | 1 |  |  |  |
| 2 |  |  |
| 3 |  |  |
| **Ausgabe:** | | | | | | |

**Aufgabe 5: Effizienzbetrachtung**

Geben Sie die Anzahl der notwendigen Vergleiche im schlechtesten Fall an.

|  |  |  |
| --- | --- | --- |
| **Beispiel** | **Anzahl der Elemente** | **Anzahl der Vergleiche** |
| [5, 3] | 2 | 0 + 1 = 1 |
| [5, 4, 3] |  |  |
| [5, 4, 3, 2] |  |  |
| [5, 4, 3, 2, 1] |  |  |
| / | n |  |