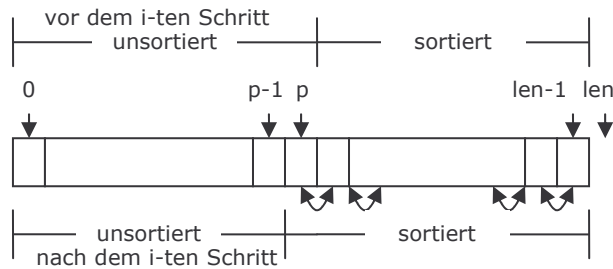




Übersichtsblatt: Selection Sort

Idee:

Der Selection Sort ist ein Sortieralgorithmus der nach dem folgenden Prinzip funktioniert. Man hat ein Feld mit unsortierten Elementen. Man geht das Feld durch, sucht das größte Element und setzt es ganz ans ende. Dann geht man das Feld wieder durch, aber nur noch bis zum vorletzten Element (Denn das letzte ist ja schon sortiert) und wiederholt den Vorgang, solange bis das unsortierte Feld leer ist, bzw. sich alle Elemente im sortierten Bereich befinden.



Pseudocode:

Eingabe: Feld A mit n Komponenten
Ergebnis: Sortiertes Feld A' mit $A'[0] \leq A'[1] \leq \dots \leq A'[n-1]$

```
int SelectionSort(A,n) {
    p = n-1;
    while (p > 0) {
        for ( j=0,max=j ; j<p ; j++ ) { /*Bis alles im sortierten Bereich ist*/
            if (A[max] < A[j+1]) { /*Suche im unsortierten Bereich*/
                max = j+1; /*Wenn A[max] größer als A[j+1]*/
            } /*Das ist das Max bei j+1*/
        }
        swap(A[max],A[p]);
        p = p-1;
    }
    return A;
}
```

Analyse:

Sei $V_{\text{Sel}}(n)$ die benötigte Anzahl von Vergleichen um n Element mit Selection Sort zu sortieren.

$$V_{\text{Sel}}(n) = \sum_{i=0}^{n-1} (n-i-1) = \sum_{i=1}^n (n-i)$$

Umdrehen der Summationsreihenfolge:

$$= \sum_{i=0}^{n-1} i$$

Gaußsche Summe:

$$= \frac{n \cdot (n-1)}{2} = \binom{n}{2} = O(n^2)$$

