Selection sort

Tom Egermann

6. Februar 2022

1 Prinzip

Sortiert von links nach rechts.
Das Gesamte Array arr wird in zwei Bereiche unterteilt, Bereich S und U. S ist der sortierte Bereich des Arrays (vorne) und U der unsortierte Bereich (dahinter). In jedem Durchgang wird das kleinste Element in U gesucht und mit dem ersten Element von U getauscht. Jetzt ist das Array im Bereich S+1 sortiert, da das kleinste Element nach vorne von U gebracht wurde. Deswegen kann die Länge von S um 1 erhöht werden. Dieses Verfahren wird arr.length-1 mal wiederholt.

2 Code

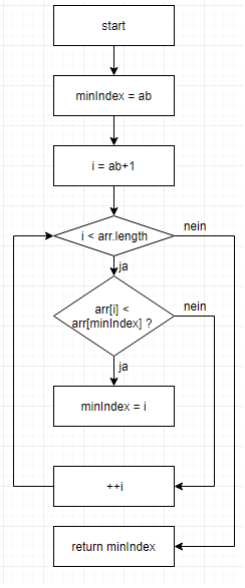
2.1 Int array

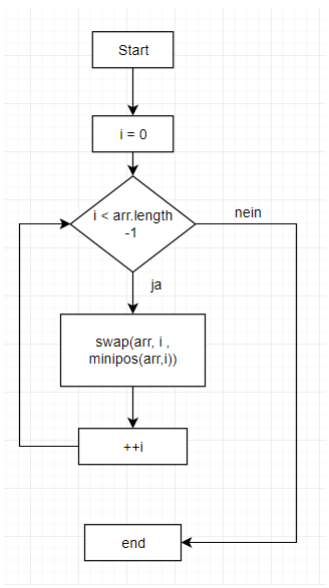
```
1  static void selectionSort(int[] arr){
2      for(int i = 0; i < arr.length-1; ++i){
3          swap(arr, i, minipos(arr, i));
4      }
5  }
6
7  static int minipos(int[] arr, int start){
8      int minIndex = start;
9      for (int i = start+1; i < arr.length; i++){
10         if (arr[i] < arr[minIndex])
11             minIndex = i;
12     }
13     return minIndex;
14 }
15
16 static void swap(int[] arr, int a, int b){
17     int temp = arr[a];
18     arr[a] = arr[b];
19     arr[b] = temp;
20 }
```

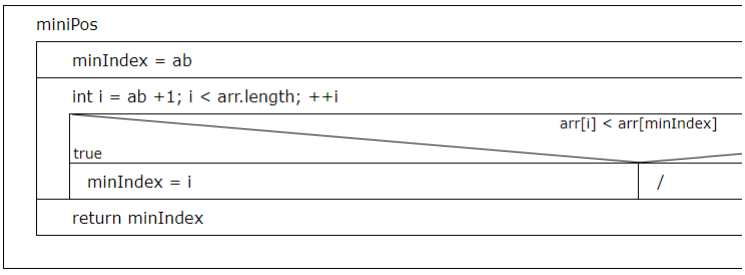
2.2 String array

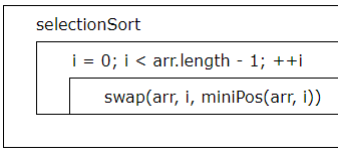
```
static void selectionSort(String[] arr){
    for(int i = 0; i < arr.length-1; ++i){
        swap(arr, i, minipos(arr, i));
    }
    show(arr);
}

static int minipos(String[] arr, int ab){
    int minIndex = ab;
    for (int i = ab+1; i < arr.length; i++)
    {
        if(arr[i].compareTo(arr[minIndex]) < 0)
            minIndex=i;
    }
    return minIndex;
}

static void swap(String[] arr, int a, int b){
    String temp = arr[a];
    arr[a] = arr[b];
    arr[b] = temp;
}
```

3 PAP, Strukto









4 Beispiel

97 93 74 65 48 13 61 1 86 80
1 93 74 65 48 13 61 97 86 80
1 13 74 65 48 93 61 97 86 80
1 13 48 65 74 93 61 97 86 80
1 13 48 61 74 93 65 97 86 80
1 13 48 61 65 93 74 97 86 80
1 13 48 61 65 74 93 97 86 80
1 13 48 61 65 74 80 97 86 93
1 13 48 61 65 74 80 86 97 93
1 13 48 61 65 74 80 86 93 97
1 13 48 61 65 74 80 86 93 97

5 Laufzeit

5.1 Best-Case

1	2	3	4	5	6	Input
1	2	3	4	5	6	5V + 3O
1	2	3	4	5	6	4V + 3O
1	2	3	4	5	6	3V + 3O
1	2	3	4	5	6	2V + 3O
1	2	3	4	5	6	1V + 3O

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

$$=> \frac{n * (n-1)}{2}$$

$$=> \frac{n^2 - n}{2}$$

$$=> \frac{n^2}{2} - \frac{(n-1)}{2}$$

$O(n^2)$

5.2 Worst-Case

6	5	4	3	2	1	Input
1	5	4	3	2	6	5V
1	2	4	4	5	6	4V
1	2	3	4	5	6	3V
1	2	3	4	5	6	2V
1	2	3	4	5	6	1V

$O(n^2)$

5.3 Average-Case

$O(n^2)$