

Übung 5

Tutoraufgabe 1 (Quicksort):

Sortieren Sie das folgende Array mithilfe von Quicksort. Geben Sie dazu das Array nach jeder Partition-Operation an und markieren Sie das jeweils verwendete Pivot-Element.

2	3	9	6	7	4	1	5	8

Tutoraufgabe 2 (Mergesort):

Sortieren Sie das folgende Array mithilfe von Mergesort. Geben Sie dazu das Array nach jeder Merge-Operation an.

2	3	9	6	7	4	1	5	8

Tutoraufgabe 3 (Heapsort):

Sortieren Sie das folgende Array mithilfe von Heapsort. Geben Sie dazu das Array nach jeder Swap-Operation an.

[illegible]

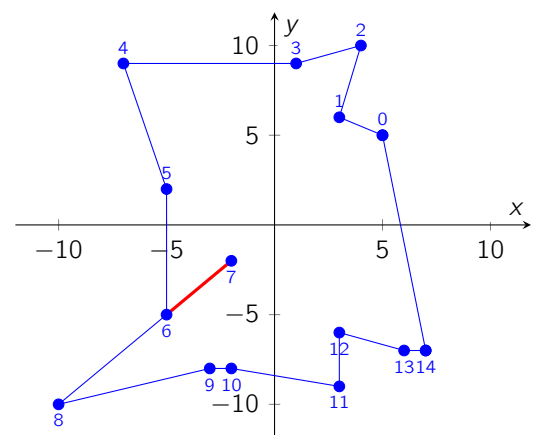
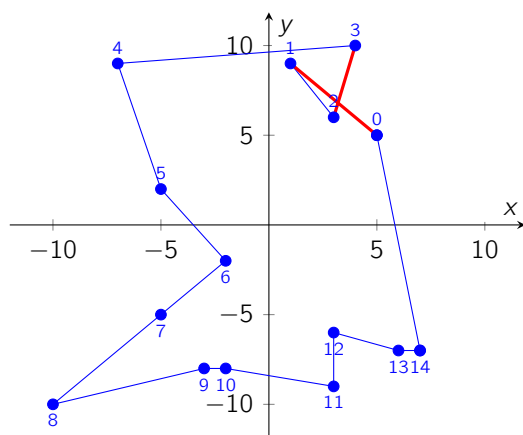
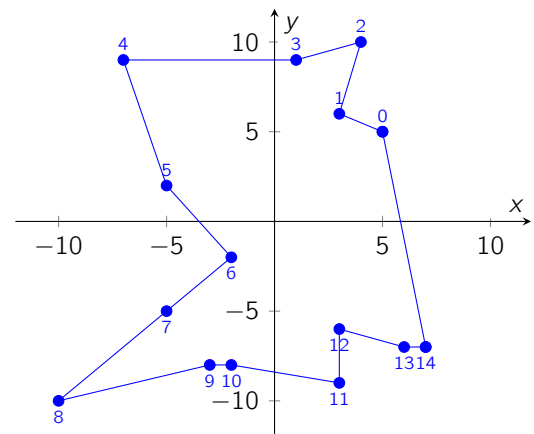
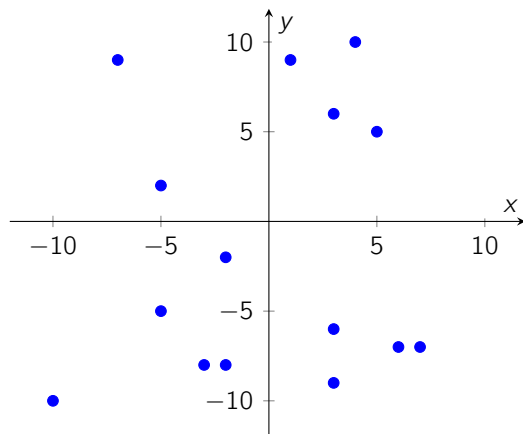
Tutoraufgabe 4 (Sortieren von Polarkoordinaten):

In dieser Aufgabe möchten wir Punkte $(x, y) \in \mathbb{Z}^2$ im 2-dimensionalen Raum nach verschiedenen Kriterien (Schlüsseln) sortieren.

- Beschreiben Sie, wie man die Sortierverfahren aus der Vorlesung abändern kann, um ein gegebenes Array von Punkten aufsteigend nach dem Euklidischen Abstand zum Ursprung zu sortieren.
- Beschreiben Sie, wie man die Sortierverfahren aus der Vorlesung abändern kann, um ein gegebenes Array von Punkten aufsteigend nach dem Winkel zwischen der x -Achse und der Linie die den Ursprung mit einem Punkt verbindet zu sortieren.
- Beschreiben Sie einen Algorithmus der als Eingabe ein Array A von n Punkten bekommt und dieses Array so sortiert, dass wir jeweils die Punkte $A[i]$ und $A[i+1]$ (für $i \in \{0, \dots, n-2\}$ sowie $A[n-1]$ und $A[0]$) mit geraden Linien verbinden können, **ohne** dass sich die Linien kreuzen bzw. überlappen.

Wir gehen davon aus, dass sich in jedem Quadranten des Koordinatensystems mindestens ein Punkt befindet. Begründen Sie kurz, warum ihr Algorithmus korrekt ist.

Beispiel. In der unten stehenden Abbildung sehen wir eine mögliche Eingabe (oben links), eine korrekte Ausgabesortierung mit eingezeichneten Linien (oben rechts), sowie zwei falsche Ausgabesortierungen bei der sich die Linien sowohl kreuzen (unten links) als auch überlappen (unten rechts).



Aufgabe 5 (Quicksort):

12 Punkte

Sortieren Sie das folgende Array mithilfe von Quicksort. Geben Sie dazu das Array nach jeder Partition-Operation an und markieren Sie das jeweils verwendete Pivot-Element.

7	3	6	4	5	1	8	9	2

Aufgabe 6 (Mergesort):

8 Punkte

Sortieren Sie das folgende Array mithilfe von Mergesort. Geben Sie dazu das Array nach jeder Merge-Operation an.

7	3	6	4	5	1	8	9	2

Aufgabe 7 (Heapsort):

11 Punkte

Sortieren Sie das folgende Array mithilfe von Heapsort. Geben Sie dazu das Array nach jeder Swap-Operation an.

[illegible]

Aufgabe 8 (d-Heaps):**3 + 2 + 2 + 2 = 9 Punkte**

Ein Baum mit Grad d ist ein solcher Baum, bei dem jeder Knoten v maximal d viele Kinder hat. Ein d -Heap ist ein Baum mit Grad d in dem die erweiterte Heap-Bedingung gilt, d.h. für alle Knoten v und alle deren Teilbäume $t_1 \dots t_n$ gilt $\max(t_i) \leq v$.

- a) Wir wollen zuerst eine Einbettung eines d -Heaps in das Array A finden. Wir wollen genauso wie bei regulären Heaps, dass wir links-vollständige Bäume mit Grad d in ein Array einbetten können und zwischen Einträgen im Array keine leeren Einträge sind.
- i. Geben Sie nun eine Abbildung an, die für einen Elternknoten $A[i]$ die Position aller ihrer Nachfolger $A[j] \dots A[k]$ in einem vollständigen Baum mit Grad d angibt.
 - ii. In welchem Eintrag müssen Sie die Wurzel speichern?
 - iii. Begründen Sie auch die Korrektheit ihrer Abbildung!
- b) Was ist die Höhe eines d -Heaps mit n vielen Knoten? Begründen Sie ihre Antwort!
- c)
- i. Erklären Sie, wie Sie die Methode `sink` aus der Vorlesung so erweitern können, dass diese auch auf d -Heaps arbeitet.
 - ii. Geben Sie außerdem eine Laufzeitanalyse für den Worstcase in O-Notation ihrer erweiterten Methode `sink` in Abhängigkeit von dem Grad des Baumes d und der Anzahl der Knoten im Baum n an.
- d)
- i. Erklären Sie schließlich noch, wie Sie die Methode `heap_sort` anpassen müssen, damit diese mit d -Heaps arbeiten können.
 - ii. Geben Sie auch für diese beide eine Laufzeitanalyse für den Worstcase in O-Notation in Abhängigkeit von dem Grad des Baumes d und der Anzahl der Knoten im Baum n an.