	27	12	19	15	23	34	25
	+	+	+	+	+	+	+
18							
z_1	z_2	z_3	z_4	z_5	z_6	z_7	z_8

Aufgabe 1 [2]

Fügen Sie in obiger Tabelle in den leeren Kästchen, vor denen das Pluszeichen steht, die Ziffern Ihrer Matrikelnummer ein. Führen Sie die Additionen durch und ermitteln Sie die Zahlen z_2 bis z_8. (z_1 ist bereits mit dem fixen Wert 18 belegt.)

Aufgabe 2 [18]

Berechnen Sie die Konstanten $a = z_6\%3 + 2, b = z_7\%3 + 2, c = z_8\%3 + 2$. Geben Sie ein Programm in C++ ähnlicher Notation an, bei dessen Laufzeituntersuchung mit dem Mastertheorem die Formelparameter a,b und c die ermittelten Werte annehmen. Bestimmen Sie die Laufzeitkomplexität Ihres Programms.

Anmerkung: Bei Bedarf kann Ihr Programm auch aus mehreren Funktionen bestehen, die einander aufrufen.

Aufgabe 3 [20]

Die Werte z_1 bis z_8. (aus Aufgabe 1) seien in dieser Reihenfolge von links nach rechts in einem Array gespeichert.
Sortieren Sie die Werte aufsteigend mit

a. [8] Bucket Sort
b. [4] Selection Sort
c. [8] Heapsort

Aufgabe 4 [20]

a. [9] Fügen Sie die Werte z_2 bis z_8 aus Aufgabe 1 (in dieser Reihenfolge) in eine zu Beginn leere Hashtabelle der Länge 7 ein. Verwenden Sie als Hashfunktion $h(k) = k\%7$ und double hashing zur Kollisionsbehandlung. Die zweite Hashfunktion ist $g(k) = k\%5 + 2$.
   Skizzieren Sie den Zustand der Hashtabelle nach jedem Einfügeschritt.
b. [1] Löschen Sie den Wert z_3 aus der Tabelle und skizzieren Sie den Zustand der Hashtabelle.
c. [5] Geben Sie den Kollisionspfad (besuchte Indexpositionen) bei einer Suche nach dem Wert z_8 an.
d. [5] Geben Sie den Kollisionspfad (besuchte Indexpositionen) bei einer Suche nach dem Wert 42 an.

Aufgabe 5 [20]

a. [8] Fügen Sie die Werte z_2 bis z_8 aus Aufgabe 1 (in dieser Reihenfolge) in einen zu Beginn leeren binären Suchbaum ein. Skizzieren Sie den Zustand des Baums nach jedem Einfügeschritt.
b. [4] Geben Sie in C++ ähnlicher Notation die Definition einer Datenstruktur für einen binären Suchbaum an.
c. [4] Geben Sie in C++ ähnlicher Notation eine Definition einer Funktion an, die den binären Suchbaum breadth first traversiert und alle gespeicherten Werte ausgibt. Sie dürfen davon ausgehen, dass Ihnen eine Klasse Queue mit den Methoden enqueue und dequeue zur Verfügung steht.
d. [4] Notieren Sie die Ausgabe Ihrer Funktion, wenn sie für den unter Teilaufgabe a erstellten Baum aufgerufen wird. Handelt es sich dabei um eine preorder Traversierung, eine postorder Traversierung, eine inorder Traversierung oder um eine andere Art der Traversierung?

Aufgabe 6 [20]

Gegeben ist die folgende Adjazenzmatrix mit Wegekosten für einen gerichteten Graphen
(die Werte z_1 bis z_8. sind aus Aufgabe 1 zu übernehmen):

$$\begin{pmatrix} 0 & 0 & z_8 & z_7 & z_6 \\ z_5 & 0 & 0 & z_3 & 5 \\ z_2 & z_1 & 0 & 1 & z_7 \\ 0 & 0 & 3 & 0 & z_5 \\ 0 & 4 & 0 & z_2 & 0 \end{pmatrix}$$

a. [2] Skizzieren Sie den gerichteten Graphen.
b. [10] Bestimmen Sie mit dem Algorithmus von Dijkstra die jeweils kürzesten Wege vom Knoten 1 (erste Zeile, erste Spalte der Matrix) zu allen anderen Knoten des Graphen.
c. [8] Entfernen Sie aus dem Graphen möglichst wenig Kanten, so dass der resultierende Graph topologisch sortierbar wird. Führen Sie mit dem so erhaltenen Graphen eine topologische Sortierung durch.