

Aufgabe 7.1 $2+1+1=4$ Punkte.

Gegeben sei ein Text mit den Zeichen $\Sigma := \{a, e, i, k, l, m, p, s\}$. Eine Analyse des Textes führe zu folgender Häufigkeitstabelle (Zeichen, Häufigkeit):

$(a, 7), (e, 10), (i, 42), (k, 5), (l, 9), (m, 16), (p, 11), (s, 22)$.

- Erstellen Sie mittels Huffman-Algorithmus einen Codebaum unter Berücksichtigung der oben angegebenen Häufigkeiten. Notieren Sie an jedem inneren Knoten die Summe der Häufigkeiten seiner Kindknoten.
- Geben Sie den Präfixcode für den Codebaum an und kodieren Sie das Wort `mississippi`.
- Wieviel Bit spart das Kodieren von `mississippi` mit Ihrem Huffman-Code im Vergleich zur Verwendung gleich großer Blöcke für jedes Zeichen? Geben Sie Ihren Rechenweg an.

(Hinweis: Die Wahrscheinlichkeit eines Buchstaben entspricht hier der Häufigkeit des Buchstaben geteilt durch die Anzahl der Buchstaben)

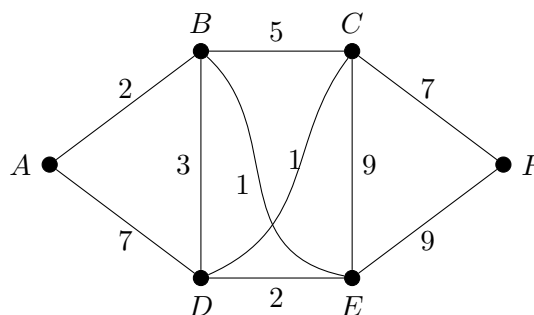
Aufgabe 7.2 4 Punkte.

Beim Huffman Algorithmus werden in jedem Schritt die zwei Symbole mit den kleinsten Häufigkeiten gesucht. Die Symbole werden dann durch ein neues Symbol ersetzt, das als Häufigkeit die Summe der Häufigkeiten der beiden gefundenen Symbole zugewiesen bekommt.

- Begründen Sie, dass nie ein neues Symbol erzeugt wird, das eine geringere Häufigkeit zugewiesen bekommt als ein davor neu erzeugtes Symbol.
- Angenommen, die ursprünglichen Symbole sind bereits nach ihrer Häufigkeit sortiert. Wie kann man dann den Huffman-Algorithmus mit Komplexität $\mathcal{O}(n)$ implementieren?

Aufgabe 7.3 5 Punkte.

Gegeben sei folgender Graph. Bestimmen Sie mit Hilfe des Dijkstra-Algorithmus die Länge der kürzesten Pfade von A zu allen anderen Knoten. Geben Sie auch die Zwischenschritte an.



— (Bitte beachten Sie die nächste Seite) —

Aufgabe 7.4 3 Punkte.

Student Oberhuber arbeitet neben Studium als Taxifahrer. Um seine Einnahmen zu optimieren hat er eine Modifikation des Dijkstra-Algorithmus entworfen, der den längsten Pfad zwischen einem ausgezeichneten Knoten u und allen anderen Knoten, findet. Die Modifikationen sind:

- Die Längen werden mit $-\infty$ initialisiert.
- Anstelle den Knoten mit der geringsten Distanz, besucht der modifizierte Algorithmus den Knoten mit der höchsten Distanz.
- Zudem, wird die längste Weglänge $l(v')$ genau dann aktualisiert, wenn $l(v) + w(v, v') > l(v')$.

Zeigen Sie, dass Oberhubers Algorithmus nicht den **längsten Pfad findet**.

Aufgabe 7.5 4 Punkte.

Lösen Sie die Programmier-Aufgabe "Weihnachtsmann & Co. KG".

Geben Sie Ihren Domjudge-Teamnamen bei Ihrer Abgabe an, damit Ihnen Ihre Lösung zugeordnet werden kann.