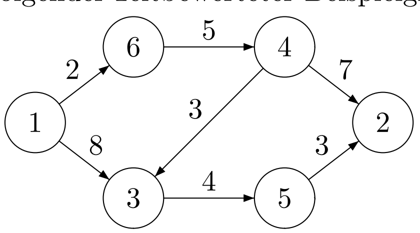
Aufgabenblatt zur 14. Übung

Zeitraum: 01.02. bis 05.02.2010

1. Aufgabe: (AGS 9.33*)

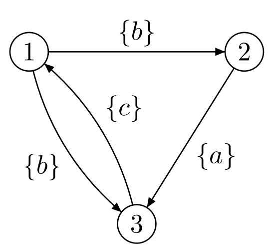
Zur Überwachung von Bauprojekten bedient man sich sehr häufig der Netzplantechnik. Die Knoten des (gerichteten) Netzplangraphen sind wichtige Ereignisse (Teilfertigstellungen, Beendigung von Planabschnitten bis hin zur Projektfertigstellung) im Prozess der Realisierung eines Projektes, die Kanten sind jeweils auszuführende Aktivitäten. Ein Ereignis kann nur dann eintreten, wenn alle notwendigen Aktivitäten, dargestellt durch die eingehenden Kanten, ausgeführt wurden. Ist ein Ereignis eingetreten, so werden alle nachfolgenden Aktivitäten, dargestellt durch die ausgehenden Kanten, in Gang gesetzt. Im einfachsten Fall sind nun ausschließlich die Kanten zeitbewertet (Zeitverbrauch bei Realisierung der Aktivität) und man interessiert sich für den Zeitabstand zwischen zwei Ereignissen. Im Besonderen kann damit die Dauer des Projektes, also der Fertigstellungszeitpunkt berechnet werden.
Folgender zeitbewerteter Beispielgraph G sei nun gegeben:



(a) Geben Sie einen geeigneten Semiring für die Lösung dieses Problems an.
(b) Sei Knoten 1 der Startknoten (Projektbeginn) und Knoten 2 der Zielknoten (Projektabschluss); berechnen Sie mithilfe des Aho-Algorithmus die Dauer bis zum Projektabschluss. Geben Sie zunächst die modifizierte Adjazenzmatrix mA_G an.
(c) Welche Aktivitäten sind zeitunkritisch? Geben Sie für diese Aktivitäten die jeweiligen Zeitpuffer an, die keine Verlängerung der Projektdauer bewirken.

2. Aufgabe: Klausuraufgabe 07.2009 (AGS 9.35)

Der gewichtete Graph $G = (V, E, c)$ über dem Semiring $(\mathcal{P}(\{a, b, c\}^*), \cup, \circ, \emptyset, \{\varepsilon\})$ sei durch folgende graphische Darstellung gegeben:



Es soll für den Graph G das Prozessproblem mit Hilfe des Aho-Algorithmus gelöst werden.
(a) Geben Sie für G die modifizierte Adjazenzmatrix mA_G an.
(b) Berechnen Sie für den Aho-Algorithmus die Matrizen $D_G^{(1)}$ und $D_G^{(2)}$. Schreiben Sie hierbei nur die Matrixelemente auf, die sich gegenüber mA_G geändert haben. Nutzen Sie dafür die Notation (i, j, L) mit $i \ldots$ Anfangsknoten, $j \ldots$ Endknoten und $L \ldots$ Sprache.
(c) Geben Sie die letzte Zeile der Ergebnismatrix D_G des Aho-Algorithmus an.
(d) Wie verändert sich der Eintrag $D_G(3, 3)$, wenn zu dem Graphen G eine Kante von Knoten 3 zum Knoten 2 mit dem Gewicht $\{b\}$ zugefügt wird?

3. Aufgabe: (AGS 10.14*)
Im Schach darf der Springer in einem Zug immer zwei Felder geradeaus und dann ein Feld links oder rechts davon gesetzt werden. Je nach Position des Springers ergeben sich somit bis zu acht verschiedene Zugmöglichkeiten. Diese seien mit a bis h bezeichnet und in Abbildung 1 dargestellt (wobei der Springer hier in der Mitte steht).
Betrachten Sie nun Abbildung 2. Finden Sie mit Hilfe des Backtracking-Verfahrens einen Weg (Zugfolge), um mit einem Springer von Feld 1 zum Feld 7 zu gelangen.
Hierbei ist zu beachten, dass:

- schraffierte Felder nicht benutzt werden dürfen,
- kein Feld auf einem Weg mehrfach betreten werden darf.

Zeichnen Sie dazu den optimierten Berechnungsbaum bis zur ersten Lösung. Gehen Sie bei der Wegeerweiterung immer in der Reihenfolge a bis h vor, wenn es in einer Situation mehrere Zugmöglichkeiten gibt.

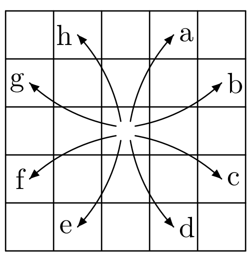


Abbildung 1

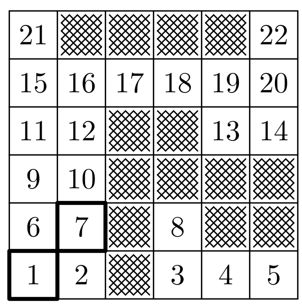


Abbildung 2

Zusatzaufgabe 1: (AGS 10.13*)
Gegeben sei folgendes Labyrinth, und es soll ein nichtzirkulärer Weg von Platz 3 nach Platz 34 gefunden werden.

42	41	40	39	38	37	36
35	34	X	32	X	30	29
28	X	26	25	24	23	22
X	20	X	18	17	16	15
14	13	12	11	10	9	8
X	6	X	X	3	2	1

Plätze, die mit einem X gekennzeichnet sind, dürfen hierbei nicht betreten werden. Von einem Platz darf die Fortsetzung des Weges nach links, nach oben, nach rechts und nach unten erfolgen (diagonale Richtungen sind also verboten).

(a) Geben Sie mit Hilfe des BACKTRACK- Algorithmus, beginnend mit dem Startplatz 3, den optimierten Berechnungsbaum der zulässigen Wegeerweiterungen bis zur 1. Lösung an.
Grundsätzlich soll gelten:

> Erweiterungen des Weges von einem aktuellen Platz müssen <u>immer</u> in der Reihenfolge nach links, nach oben, nach rechts, nach unten getätigt werden.
>
> Eine Erweiterung ist zulässig, wenn der Erweiterungsplatz
> – nicht außerhalb einer Randlinie liegt,
> – nicht durch ein X gesperrt ist und
> – nicht auf dem bisherigen Weg liegt.

(b) Setzen Sie die Suche bis zum Auffinden der 2. Lösung fort; geben Sie dazu die notwendigen Erweiterungen im Berechnungsbaum der Aufgabenstellung (a) an. Gibt es noch weitere Lösungen? Begründen Sie kurz Ihre Antwort.

Zusatzaufgabe 2: (AGS 10.10*)
Sei M eine Menge von n Elementen. Eine *Permutation* von M ist dann eine (beliebige) Anordnung aller n Elemente von M (entspricht einem n-Tupel, wo jedes Element von M genau einmal vorkommt). Die Anzahl der bildbaren Permutationen ist bekanntlich n-Fakultät.
Die Menge $\{1, 2, 3\}$ besitzt zum Beispiel die Permutationen: 123, 132, 213, 231, 312, 321.
a) Entwickeln Sie einen Algorithmus mit Hilfe des Backtracking–Prinzips, der <u>alle</u> Permutationen von M ermittelt und auflistet. Definieren Sie hierzu eine rekursive Vorschrift, bei der in jedem Rekursionsschritt versucht wird, eine bereits vorhandene Teilfolge von verschiedenen Zahlen um eine weitere Zahl zu ergänzen, die von allen Zahlen der Teilfolge verschieden ist. Nutzen Sie für die Reihefolge der Erweiterungen von Teillösungen die Ordnungsrelation von $\mathbb{N}$.
b) Schreiben Sie den Berechnungsbaum bis zur 4. Lösung auf.
c) Schreiben Sie zu Ihrem Algorithmus ein C-Programm, das nach Eingabe von $n \in \mathbb{N}$ alle Permutationen von $\{1, \ldots, n\}$ berechnet und ausgibt.
Hinweis: Begrenzen Sie n zunächst auf maximal 9, um die Aufgabe besser handhabbar zu machen.