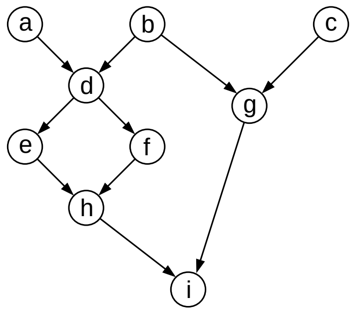
Übungsblatt 7

Bemerkung: Aufgrund der in der letzten Übung aufgetretenen Diskrepanz zwischen den Beispielnummern in TUWEL und jenen auf den Übungsblättern werden zum Ausgleich die Nummern 44-48 übersprungen.

49.) Erklären Sie die Funktionsweise von Topological-Sort (in der VO Top-Sort genannt) für das topologische Sortieren eines gerichteten azyklischen Graphens an Hand des nachfolgend angegeben Graphen G:



50.) Geben Sie einen Algorithmus mit Laufzeit $\mathcal{O}(|V|+|E|)$ an, dessen Eingabe ein gerichteter azyklischer Graph $G = (V, E)$ und zwei Knoten s und t sind und der die Anzahl der (gerichteten) Wege (= einfache Pfade) von s nach t in G berechnet (Whg.: ein Weg ist eine Kantenfolge, wo alle Knoten entlang der Kantenfolge verschieden sind). Beispielsweise enthält der gerichtete azyklische Graph aus Bsp. (1) drei Wege vom Knoten b zum Knoten i, nämlich $b-d-e-h-i$, $b-d-f-h-i$ und $b-g-i$. Anmerkung: Ihr Algorithmus soll die Pfade nur zählen, nicht ausgeben.

51.) Beweisen oder widerlegen Sie die folgende Behauptung: Falls ein gerichteter Graph G Zyklen enthält, dann liefert Topological-Sort(G) eine lineare Ordnung der Knoten, für die die Anzahl der Kanten, die inkonsistent mit dieser Ordnung sind, minimal ist.

52.) Eine alternative Möglichkeit, um einen gerichteten azyklischen Graphen G zu sortieren, ist wie folgt: Der Algorithmus sucht einen Knoten mit Hingrad 0 in G, gibt ihn aus und entfernt ihn aus dem Graphen. Der so veränderte Graph heiße G'. Dann wird der Algorithmus rekursiv auf G' angewendet. Schreiben Sie eine Pseudocode-Implementierung dieser Idee, deren Laufzeit $O(|V| + |E|)$ beträgt. Wie sieht das Ergebnis, wenn der Eingabegraph Zyklen besitzt?

53.) Modifizieren Sie Memoized-Cut so, dass nicht nur der Wert, sondern auch die tatsächliche Lösung ausgegeben wird.

54.) Entwerfen Sie einen Algorithmus, der dynamische Programmierung benutzt und in linearer Zeit die Fibonacci-Zahlen berechnet, d.h. zur Bestimmung von F_n sind $O(n)$ Schritte nötig. Bestimmen Sie auch den zugehörigen Teilproblemgraph sowie dessen Knoten- und Kantenanzahl.

55.) Sei e eine Kante mit maximalem Gewicht auf einem Zyklus eines zusammenhängenden, kantenbewerteten Graphen $G = (V, E)$. Beweisen Sie, dass es einen minimalen Spannbaum von

$G' := (V, E - \{e\})$ gibt, der gleichzeitig ein minimaler Spannbaum von G ist. Das heißt, es gibt einen minimalen Spannbaum von G, der e nicht enthält.

56.) Alternative Algorithmen zur Bestimmung minimaler Spannbäume. Nachfolgend sind die Pseudocodes von drei verschiedenen Algorithmen angegeben. Alle drei erhalten als Eingabe einen zusammenhängenden kantenbewerteten Graphen und geben eine Kantenmenge T zurück. Beweisen oder widerlegen Sie für jeden der drei Algorithmen die Behauptung, dass T in jedem Fall ein minimaler Spannbaum ist.

```
MAYBE-MST-A(G, w)
sortiere die Kanten in nichtsteigender Reihenfolge nach ihren Gewichten w
T := E
for e ∈ E (in der soeben berechneten Reihenfolge) do
  if T \ {e} ist ein zusammenhängender Graph then
    T := T \ {e}
  end if
end do
return T
```

```
MAYBE-MST-B(G, w)
T := ∅
for e ∈ E (in einer beliebigen Reihenfolge) do
  if T ∪ {e} kreisfrei then
    T := T ∪ {e}
  end if
end do
return T
```

```
MAYBE-MST-C(G, w)
T := ∅
for e ∈ E (in einer beliebigen Reihenfolge) do
  T := T ∪ {e}
  if T enthält einen Zyklus c then
    sei e' eine Kante von c mit maximalem Gewicht
    T := T \ {e'}
  end if
end do
return T
```