Karlsruher Institut für Technologie

Tutorium 11

Algorithmen I SS 14

Institut für Theoretische Informatik

Mittsemesterklausur

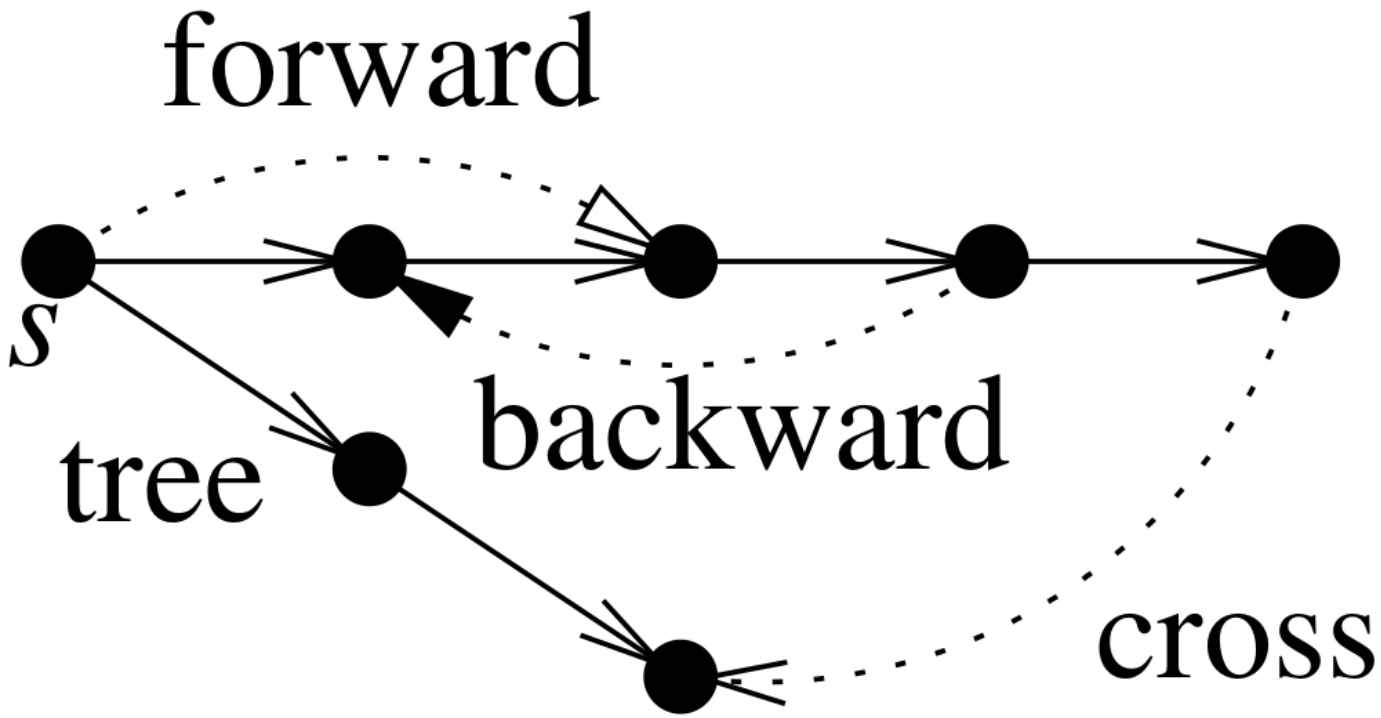
Duplikaterkennung

- Gegeben: Array der Größe n mit Elementen $\in 0, \ldots, n-1$
- Gesucht: Duplikat finden in $\mathcal{O}(n)$ und *in-place*
- Bucketsort ist **nicht** in-place
- Betrachte Array als Permutation, löse schrittweise Zykel auf

Urlaubsdatenbank

- Gegeben: Paare mit ID und Urlaubstag
- Gesucht: IDs der Mitarbeiter mit mindestens 30 Tagen Urlaub in *erwartet* $\mathcal{O}(n)$
- Wichtig: IDs können sehr groß werden. Array über Wertebereich wird zu groß!
- Lösung: Hashtabelle
- Gesucht (2): Tag mit maximalen Urlaubsnehmern in *deterministisch* $\mathcal{O}(n)$
- Lösung mit Array, nur 365 Tage

Aufgabe mit Graphen



Familie von Graphen

- Familie von DAGs mit $\Omega(n^2)$ Kanten
- $\Omega \approx$ „mindestens“

Summe von Paaren

- Gegeben: Array mit n Zahlen
- Gesucht: i, j mit $A[i] + A[j] = x$ in $\mathcal{O}(n \log n)$
- Zunächst Liste sortieren
- Lösung 1: für jedes Element in der Liste mit binärer Suche entsprechende andere finden
- Lösung 2 (schneller): Bewege i und j vom linken bzw. rechten Rand aufeinander zu, bis Summe erreicht ist
- Beweis der Korrektheit, nicht Laufzeit begründen!

Aufgabe: Bottleneck Shortest Path

Sei $G = (V, E)$ ein zusammenhängender ungerichteter gewichteter Graph und $s, t \in V$. Ein kreisfreier Pfad P zwischen s und t heiße ein Bottleneck Shortest Path (BSP) für s und t, wenn das größte in P auftretende Kantengewicht minimal ist für alle Pfade zwischen s und t

1. Zeigen Sie: Ist T ein MST in G, dann ist der in T eindeutige Pfad P zwischen zwei Knoten $s, t \in V$ ein BSP in G für s und t.
2. Geben Sie einen Algorithmus an, der für gegebenen $G = (V, E)$, gegebene $s, t \in V$ und einen gegebenen MST T in G einen BSP P zwischen s und t ausgibt. Die Laufzeit soll dabei in $\mathcal{O}(|P|)$ liegen. Nehmen Sie an T liege in Form des Array *parent* vor.
3. Argumentieren Sie kurz warum ihr Algorithmus korrekt und die geforderte Laufzeit hat.

Kreativaufgabe: Streaming MST

Gegeben sei ein zusammenhängender Graph mit n Knoten und m Kanten. Die Knoten sind lokal gespeichert, während die Kanten über eine Netzwerkverbindung gestreamt werden. Sie können nicht lokal gespeichert werden, da nur $\mathcal{O}(n)$ Speicherplatz vorhanden ist.

Aufgabe 1: Gib einen Algorithmus an, der einen MST von G unter diesen Einschränkungen bestimmt.

Aufgabe 2: Verbessere diesen Algorithmus so, dass er nur $\mathcal{O}(m \log n)$ Rechenzeit benötigt.