

Übungsblatt 2

Abgabe bis 20. November 2024

Organisatorisches:

- Für das Lösen der Übungsaufgaben habt ihr *zwei Wochen* Zeit.
- Die Bearbeitung soll möglichst in Zweiergruppen erfolgen.
- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: WEIGHTED VERTEX COVER

5 Punkte

Gegeben sei ein ungerichteter Graph G mit Kostenfunktion $c : V \rightarrow \mathbb{N}$ auf den Knoten. Beim WEIGHTED VERTEX COVER-Problem ist ein Vertex Cover von G mit minimalen Gesamtkosten gesucht. In dieser Aufgabe betrachten wir nur Bäume. Gib ein dynamisches Programm an, das für einen Baum die minimalen Kosten eines Vertex Covers bestimmt. Gib außerdem die Laufzeit für die Berechnung des DPs an.

Aufgabe 2: 3-HITTING SET

10 Punkte

Eine Instanz von 3-HITTING SET besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i . Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2,562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 3: d -HITTING SET

10 Punkte

Eine Instanz von d -HITTING SET besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i . Gib einen Algorithmus mit Laufzeit $(d - 0,658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2,342^k \cdot n^{O(1)}$ für 3-HITTING SET zu erhalten.

Aufgabe 4: VERTEX COVER mit beschränkten Suchbäumen 5 Punkte + 1 Bonus-Punkt

In dieser Aufgabe sollst du in einer Programmiersprache deiner Wahl ein Programm implementieren, das einen Graphen einliest und dazu ein minimales Vertex Cover berechnet. Nutze dazu die Verfahren zu beschränkten Suchbäumen aus Vorlesung 3. Du darfst aber gerne auch weitere Regeln implementieren und Optimierungen machen.

Beschreibe in der PDF-Abgabe, welche Verfahren du implementiert hast und was du noch weiter optimiert hast. Gib außerdem in der PDF-Abgabe für jeden Graphen an, welche Größe dein gefundenes Vertex Cover hat. Gib zusätzlich den Quellcode sowie deine gefundenen Lösungen (im unten beschriebenen Format) als eine ZIP-Datei ab.

Für jedes minimale Vertex Cover kannst du einen Punkt bekommen.

Dateiformat Eingabe: In der ersten Zeile stehen n und m , die Anzahl von Knoten und Kanten des Graphen. In den nächsten m Zeilen sind jeweils zwei Knoten angegeben, die durch eine ungerichtete Kante verbunden sind. Dabei sind die Knoten von 1 bis n durchnummeriert.

Dateiformat Ausgabe: In der ersten Zeile steht die Lösungsgröße k . In den folgenden k Zeilen steht jeweils die ID eines Knotens, der Teil des Covers ist.

Hinweis: Mithilfe der Datei `validator.py` kannst du verifizieren, ob ein Vertex Cover gültig ist. Minimalität wird dabei nicht geprüft. Bitte prüfe selbst vor der Abgabe, ob deine gefundenen Vertex Cover gültig sind.