

186.866 Algorithmen und Datenstrukturen VU

Übungsblatt 6

für die Übung am Montag den 4. bzw. Dienstag den 5. Juni 2018.

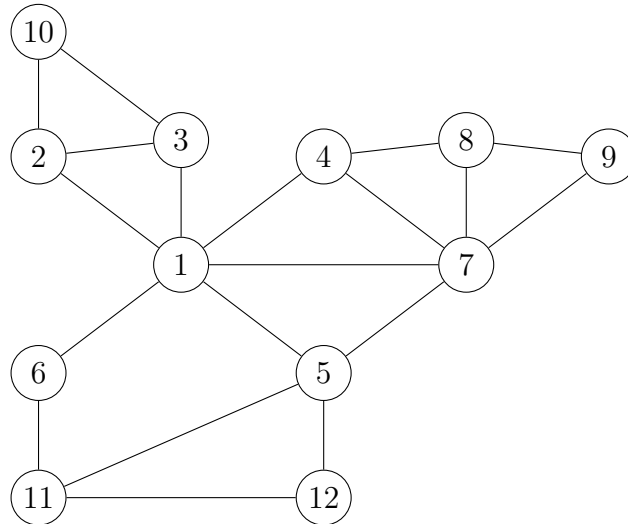
Geben Sie bis **spätestens Sonntag, 3.6.2018, 13:00 Uhr** über TUWEL an, welche Beispiele Sie bearbeitet und gelöst haben. Gehen Sie dabei folgendermaßen vor:

- TUWEL (<https://tuwel.tuwien.ac.at>)
Kurs *186.866 Algorithmen und Datenstrukturen (VU 5.5)* im Abschnitt *Übungsblätter*
- Bearbeitete Beispiele ankreuzen **und** abgeben
 - Link *Ankreuzen Übungsblatt 6*
Button *Abgabe bearbeiten*
Bearbeitete Beispiele anhängen und *Änderungen speichern*.
 - Link *Hochladen Lösungen Übungsblatt 6*
Button *Abgabe hinzufügen*
PDF-Datei mit Lösungen hochladen und *Änderungen sichern*.
 - Wenn Sie das Programmierbeispiel gelöst haben:
Link *Hochladen Source-Code Übungsblatt 6*
Button *Abgabe hinzufügen*
Java-Datei (**E6.java**) hochladen und *Änderungen sichern*.

Bitte beachten Sie:

- Sie können **vor** der Deadline beliebig oft Ihre Auswahl an Beispielen und das zugehörige Lösungs-PDF verändern, aber **nach** der Deadline gibt es **keine** Veränderung ihrer angekreuzten Beispiele **und** der abgegebenen Dateien!
- Sie können Ihre Lösungen entweder direkt in einem Textverarbeitungsprogramm erstellen, oder aber auch gut leserliche Scans bzw. Fotos von handschriftlichen Ausarbeitungen einreichen.
- Bitte geben Sie Ihren Namen, Matrikelnummer und E-Mail-Adresse in den Ausarbeitungen an.
- Wenn Sie das Programmierbeispiel kreuzen, muss sowohl die theoretische Ausarbeitung im PDF, als auch der Source-Code im Tuwel abgegeben werden.
- Beachten Sie die Richtlinien für das An- und Aberkennen von Beispielen in den Vorbesprechungsfolien. Neben der Überprüfung in der Übungseinheit werden danach stichprobenartig weitere Abgaben auf Spekulation und Plagiate überprüft.

Aufgabe 1 Wenden Sie den Branch and Bound Algorithmus zum Finden eines minimalen Vertex Covers auf dem folgenden Graphen an. Geben Sie den Ablauf des Algorithmus als Baum wieder. Geben Sie für jeden Schritt die obere Schranke U' und die untere Schranke L' an, sowie die aktuelle Teilinstanz und die aktuell beste gefundene Lösung S .



Aufgabe 2 Lösen Sie die folgende Instanz des Gewichteten Interval Scheduling Problems mittels Dynamischer Programmierung. Geben Sie das Array M an, sowie jeweils die Werte $w_j + M[p(j)]$ und $M[j - 1]$. Geben Sie außerdem für jeden Job j den Wert $p(j)$ an. Berechnen Sie anschließend aus diesem Array eine Lösung und erklären Sie wie Sie zu dieser Lösung gekommen sind.

Job	Start	Ende	Gewicht
1	4	8	1
2	2	7	5
3	9	11	1
4	1	3	2
5	2	17	7
6	8	10	3
7	3	4	1
8	3	5	2
9	13	16	9
10	10	14	10
11	7	12	11

Aufgabe 3 Entwerfen und beschreiben Sie einen Branch-and-Bound-Algorithmus für das Optimierungsproblem MAX 2SAT, welches wie folgt definiert ist:

Gegeben sei eine Menge von Booleschen Variablen $V = \{x_1, x_2, \dots, x_n\}$ sowie eine Menge $C = \{c_1, c_2, \dots, c_m\}$ von Klauseln mit genau zwei Literalen, d.h. für jede Klausel c_i gilt $c_i = (l_i^1 \vee l_i^2)$ und $l_i^k = x$ oder $l_i^k = \bar{x}$ für $k \in \{1, 2\}$ und ein $x \in V$. Ziel ist eine Belegung der Variablen zu finden, die möglichst viele Klauseln erfüllt.

Betrachten Sie dazu zunächst folgende Punkte:

1. Wie repräsentieren Sie ein Teilproblem und wie erfolgt das Branching, d.h. das Aufteilen eines (Teil-)Problems in weitere Teilprobleme?
 2. Beschreiben Sie eine sinnvolle Möglichkeit für die Auswahl des nächsten Teilproblems. Nach welcher Strategie gehen Sie dabei vor?
 3. Wie erstellen Sie eine gültige initiale Lösung? Entspricht diese Lösung einer unteren oder einer oberen Schranke für die gegebene Probleminstanz?
-

Aufgabe 4 Im nächsten Schritt sollen Sie Ihren Branch-and-Bound Algorithmus für MAX 2SAT durch geeignete Heuristiken zur Berechnung von unteren und oberen Schranken verfeinern.

Betrachten Sie dazu die folgenden beiden Punkte:

1. Was ist eine geeignete untere Schranke für die Anzahl erfüllbarer Klauseln im aktuellen Teilproblem?
2. Wie kann eine geeignete obere Schranke für den Lösungswert des aktuellen Teilproblems berechnet werden?

Bestimmen Sie möglichst gute, nicht-triviale untere und obere Schranken und beschreiben Sie, wie diese berechnet werden. Ihre untere Schranke soll größer sein als die Anzahl durch fixierte Variablenwerte bereits sicher erfüllter Klauseln. Ihre obere Schranke soll kleiner sein als die Gesamtzahl noch offener Klauseln mit unbestimmtem Wahrheitswert.

Hinweis: Betrachten Sie insbesondere diejenigen Klauseln, für die im aktuellen Teilproblem bereits eines der beiden Literale mit einem Wahrheitswert belegt ist und nutzen Sie dies für die Berechnung von Schranken.

Aufgabe 5 Programmieraufgabe (2 Punkte)

Ihre Aufgabe ist es, den **Branch-and-Bound Algorithmus für Vertex Cover** zu implementieren.

Die **detaillierte Angabe** zu dieser Programmieraufgabe finden Sie **im TUWEL** (PDF-Dokument „Programmieraufgabe 6“). Dieses Dokument enthält auch eine Beschreibung der zu implementierenden Funktionen und die Theoriefragen dieser Aufgabe.

Sie bekommen das benötigte Framework („AlgoDat_E6.zip“) ebenfalls im TUWEL zur Verfügung gestellt. Implementieren Sie darin in der Datei „exercise/E6.java“ die geforderten Methoden.

Wichtig: Damit das Beispiel als positiv absolviert gilt, müssen **alle** geforderten Methoden korrekt implementiert und die Theoriefragen in der PDF-Abgabe ausführlich beantwortet werden!
