

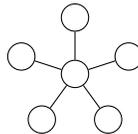
## Übungsblatt 5

Abgabe bis 15. Januar 2025

### Aufgabe 1: Sterne

8 Punkte

Ein 5-Stern ist folgender Graph:



Gegeben einem ungerichteten Graphen  $G = (V, E)$  und Parameter  $k$  soll entschieden werden, ob  $G$  mindestens  $k$  knotendisjunkte induzierte 5-Sterne enthält. Verwende *color coding* um zu zeigen, dass dieses Problem in FPT liegt.

### Aufgabe 2: LONGEST CYCLE

12 Punkte

Gegeben sei ein Graph  $G = (V, E)$  und ein Parameter  $k$ . Bei LONGEST CYCLE geht es darum, zu entscheiden ob es in  $G$  einen Kreis der Länge *mindestens*  $k$  in  $G$  gibt. Gib einen FPT-Algorithmus für dieses Problem an.

*Hinweis:* Beachte, dass es Graphen gibt, die keinen Kreis der Länge genau  $k$  enthalten, aber dennoch einen Kreis der Länge mindestens  $k$ .

### Aufgabe 3: Dominating Set auf speziellen Graphen

10 Punkte

Betrachte folgende parametrisierte Variante von DOMINATING SET. Der Graph  $G$  ist gegeben zusammen mit einer Knotenordnung  $V = \{v_1, \dots, v_n\}$ . Die Länge einer Kante  $\{v_i, v_j\}$  ist  $|i - j|$ . Der Parameter  $k$  ist die maximale Kantenlänge in  $G$ . Gib einen FPT-Algorithmus an, der diese Parametrisierung von DOMINATING SET löst.

### Aufgabe 4: CLOSEST STRING

5 Bonus-Punkte

In dieser Aufgabe sollst du erneut ein Programm implementieren, das das Problem CLOSEST STRING (in einer leichten Variante) löst. Gegeben sind  $k$  Strings der Länge  $n$ . Gesucht ist ein minimales  $D$ , sodass ein String  $s^*$  existiert, der zu jedem anderen String Hamming-Distanz höchstens  $D$  hat.

Löse das Problem mittels eines ILP wie in der 3. Übung. Du darfst dir gerne weitere Optimierungen überlegen. Beschreibe in der PDF-Abgabe, welche Änderungen du an der ILP Formulierung aus

der Vorlesung vorgenommen hast und ob es sonst noch Einsichten gab, die dir beim Lösen von dem Problem geholfen haben. Gib außerdem in der PDF-Abgabe für jede Instanz das kleinste  $D$  an, für das du eine Lösung gefunden hast.

Gib zusätzlich den Quellcode sowie deine gefundenen Lösungen (im unten beschriebenen Format) als eine ZIP-Datei ab. Für jede gelöste Instanz kannst du einen Punkt bekommen.

*Dateiformat Eingabe:* In der ersten Zeile steht  $k$ , die Anzahl der Strings. In den nächsten  $k$  Zeilen ist jeweils ein String gegeben. Alle Strings haben die selbe Länge und verwenden nur die kleinen Buchstaben a-z.

*Dateiformat Ausgabe:* Eine Zeile mit einem String  $s^*$ , der minimale Hamming-Distanz zu den gegebenen Strings hat.

*Hinweis:* Mithilfe der Datei `validator.py` kannst du die Hamming-Distanz deiner Lösung zu den gegebenen Strings bestimmen. Dabei wird das Maximum über alle Hamming-Distanzen ausgegeben.

*Hinweis:* Für diese Aufgabe benötigst du einen ILP-Solver. Die Anzahl der Programmiersprachen und Solver ist groß, aber du solltest in der Lage sein, die Aufgabe mit den meisten Kombinationen zu lösen. Einer der mit Abstand besten Solver ist Gurobi. Um ihn benutzen zu können, brauchst du allerdings eine Lizenz, die du kostenlos mit deiner Studenten-E-Mail erhältst.

*Wir wünschen euch schöne Weihnachtsferien!*