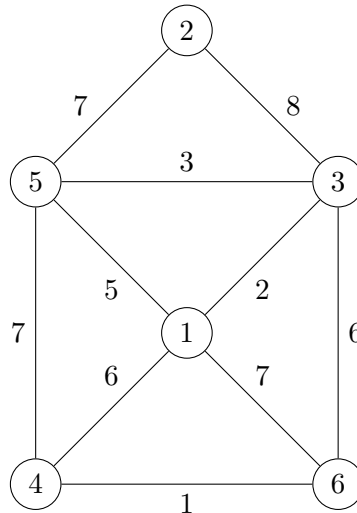


**Aufgabe 12.1** (4 Punkte) Gegeben sei der folgende ungerichtete Graph  $G$ :



- a) Führen Sie den Algorithmus von Jarník-Prim beginnend mit Knoten 1 durch und geben Sie die Werte für  $d$  und  $parent$  nach jedem Durchlauf der `while`-Schleife an. Wenn es während des Algorithmus mehrere Kanten mit gleichen minimalen Kosten gibt, wählen Sie die bzgl der topologischen Sortierung

$$(u, v) < (r, s) \iff: (u < r) \vee (u = r \wedge v < s)$$

kleinere.

- b) Geben Sie eine Reihenfolge der Kanten an, sodass der Algorithmus von Kruskal einen anderen minimalen Spannbaum erzeugt als der von Jarník-Prim.

**Aufgabe 12.2** (4 Punkte) Der Algorithmus von Dijkstra ähnelt dem Algorithmus von Jarník-Prim, erzeugt jedoch nicht immer einen minimalen Spannbaum. Zeigen Sie dies, indem sie für ein beliebiges  $n \in \mathbb{N}$  einen Graphen mit mindestens  $n$  Knoten angeben, für den der Algorithmus von Dijkstra im Gegensatz zum Algorithmus von Jarník-Prim einen nicht minimalen Spannbaum berechnet.

**Aufgabe 12.3** (4 Punkte) Gegeben sei ein Union-Find-Set mit 32 einelementigen Mengen. Geben Sie eine möglichst kurze Folge von Operationen an und beschreiben Sie deren Wirkung, sodass das resultierende Union-Find-Set aus nur einer Menge besteht und der Baum der diese Menge repräsentiert maximale Tiefe hat.