

4. Schreibe eine Klassenmethode, welche aus dem File `steps.txt` Instruktionen ausliest und mit Hilfe geeigneter Datenstrukturen herausfindet, in welcher Reihenfolge die verschiedenen Schritte durchzuführen sind. Falls zu einem Zeitpunkt mehrere Schritte möglich sind, so ist der alphabetisch niedrigste durchzuführen. \_\_\_\_/8

Beispiel:

```
Step C must be finished before step A can begin.
Step C must be finished before step F can begin.
Step A must be finished before step B can begin.
Step A must be finished before step D can begin.
Step B must be finished before step E can begin.
Step D must be finished before step E can begin.
Step F must be finished before step E can begin.
```

führt zur Struktur

```
  -->A---->B--
 /      \      \
C         -->D----->E
 \         /
  ---->F-----
```

Diese wird folgendermaßen abgearbeitet:

- (a) Zu Beginn ist nur Schritt **C** verfügbar und wird abgearbeitet.
- (b) Nun sind **A** und **F** verfügbar; **A** ist alphabetisch niedriger und wird abgearbeitet.
- (c) Verfügbar: {**B D F**}; **B** wird abgearbeitet.
- (d) Verfügbar: {**D F**}; **D** wird abgearbeitet.
- (e) **F** wird abgearbeitet.
- (f) **E** wird abgearbeitet.

Also ist für dieses Beispiel die Reihenfolge der Schritte **CABDFE**.

Gesamtpunkte: \_\_\_\_/20P

Prozent	Note
≥ 90	Sehr gut
≥ 80	Gut
≥ 65	Befriedigend
≥ 50	Genügend
< 50	Nicht genügend