

Aufgabe 1. Schreiben Sie ein Programm, das Zahlen aus einer Datei einliest und das kleinste gemeinsame Vielfache aller eingelesenen Zahlen ausgibt. Sie können annehmen, dass in der Datei in jeder Zeile genau eine Zahl steht und dass die Datei nur positive ganze Zahlen enthält.

Sie können als Unterfunktion natürlich Ihre Lösung von Aufgabe 4 des 2. Zettels benutzen, mit der Sie den größten gemeinsamen Teiler von zwei positiven ganzen Zahlen bestimmen können.

Testen Sie ihr Programm auf selbstgeschriebenen Instanzen.

Aufgabe 2. Schreiben Sie ein Programm, das zu einer gegebenen Menge von achsenparallelen Rechtecken mit ganzzahligen Koordinaten entscheidet, ob diese Rechtecke paarweise überlappungsfrei sind.

Die Eingabe ihres Programm sei eine Datei, die in jeder Zeile genau ein Rechteck kodiert. Dazu enthält jede Zeile genau vier ganze Zahlen (getrennt durch Leerzeichen) a , b , c und d , mit der das Rechteck

$$[a, b] \times [c, d]$$

kodiert wird. Wir betrachten dabei zwei Rechtecke $[a_1, b_1] \times [c_1, d_1]$ und $[a_2, b_2] \times [c_2, d_2]$ als überlappend, wenn $([a_1, b_1] \times [c_1, d_1]) \cap ([a_2, b_2] \times [c_2, d_2])$ nicht leer ist (auch wenn dieser Schnitt z.B. nur aus einem Punkt besteht).

Wenn Ihr Programm feststellt, dass es zwei überlappende Rechtecke gibt, soll es zwei solche Rechtecke und einen Punkt, der in beiden liegt, ausgeben. Wenn die Rechtecke paarweise überlappungsfrei sind, soll Ihr Programm eine entsprechende Meldung ausgeben.

Führen Sie auch für dieses Programm Korrektheitstests mit selbstgeschriebenen Instanzen durch.

Aufgabe 3. Auf der Veranstaltungshomepage finden Sie Instanzen für die vorige Aufgabe. Diese sind immer nach dem selben Prinzip aufgebaut (und insbesondere überlappungsfrei), unterscheiden sich aber stark in der Größe. Lesen Sie diese Instanzen ein und messen Sie (wie in der Vorlesung erläutert) die Laufzeit, die Sie für die Überprüfung auf Überlappungsfreiheit benötigen.

Aufgabe 4. Implementieren Sie ein Programm, das Folgendes leistet: Zunächst soll das Programm eine Datei einlesen, deren Einträge ausschließlich nichtnegative ganze Zahlen sind. In der ersten Zeile der Datei steht genau eine Zahl $n > 1$, jede weitere Zeile enthält genau zwei Zahlen aus der Menge $\{0, \dots, n - 1\}$ (wobei beide Zahlen identisch sein können und auch Zeilen doppelt auftreten können). Ihr Programm soll dann entscheiden, ob es eine endliche Folge nichtnegativer ganzer Zahlen v_0, \dots, v_k gibt, so dass Folgendes gilt:

- $v_0 = 0$,
- $v_k = 1$ und
- für jedes $i \in \{0, \dots, k - 1\}$ gibt es eine Zeile aus der Datei, welche als ersten Eintrag v_i und als zweiten Eintrag v_{i+1} hat.

Legen Sie sich dazu eine Datenstruktur an, die für jedes $i \in \{0, \dots, n - 1\}$ alle $j \in \{0, \dots, n - 1\}$ abspeichert, für die es in der Datei eine Zeile der Form

i j

gibt. Wenn es eine solche Folge gibt, soll Ihr Programm natürlich eine solche ausgeben. Testen Sie Ihr Programm auf eigenen Instanzen, es gibt aber auch Instanzen auf der Veranstaltungshomepage.

Beispiel: Bei einer Eingabe von

```
4
3 1
0 3
0 2
```

wäre die Antwort "ja", weil es die Folge 0 3 1 gibt.
