Primzahlsuche: Zahlensieb

Vom griechischen Philosoph und Mathematiker Eratosthenes von Kyrene (3. Jahrhundert v. Chr.) ist ein Verfahren überliefert, Primzahlen bis zu einer beliebigen Grenze schnell zu finden. Das Verfahren ist bekannt als das "Sieb des Erathostenes" oder "Zahlensieb".

Idee

1. Man stellt zunächst eine Liste mit allen Zahlen von 2 bis r gewünschten Obergrenze zusammen.
2. Jetzt streicht man alle Vielfachen von 2, denn das sind ja keine Primzahlen (durch 2 teilbar) und "behält" die 2 als erste Primzahl.
3. Die nächste nicht durchgestrichene Zahl ist die nächste Primzahl - die 3.
4. Jetzt streicht man alle Vielfachen der 3.
5. Jetzt wiederholt man die Schritte ab 3. bis man am Ende des Zahlenbereichs angekommen ist.

Die Zahlen, die dann noch übrig sind, sind die gesuchten Primzahlen.

Modellierung des Problems

Man verwendet ein Array aus booschen-Werten (wahr/falsch). "wahr" soll dabei für Primzahl stehen, "falsch" für "keine Primzahl". Wenn also das Arrayelement `sieb[9]` den Wert `true` hat, würde das bedeuten, dass wir 9 für eine Primzahl halten.

Aufgabe

Das Grundgerüst eines BlueJ Projekts kannst du hier herunterladen oder mit dem Befehl `git clone https://gitea.schule.social/QGM-Unterricht/bluej-eratosthenes.git` aus dem Repo klonen.

- Überlege dir zunächst, was die Grenzen für den Index deines Sieb-Arrays sein sollten, damit die den Zahlenbereich von 1 bis zur `grenze` abdecken kannst.
- Vervollständige den Konstruktor: Erzeuge und initialisiere das Sieb-Array.[1)]

[1)]
0 und 1 sind keine Primzahlen, die kleinste Primzahl ist 2

From:
https://www.info-bw.de/ -

Permanent link:
https://www.info-bw.de/faecher:informatik:oberstufe:java:algorithmen:arrays:eratosthenes:start?rev=1616666922

Last update: **25.03.2021 10:08**