

Beispiel 1 (100 Punkte): Mustersuchalgorithmen.

Implementieren Sie drei Klassen `brute_search`, `bm_search` und `kmp_search`, die jeweils einen der in der Vorlesung besprochenen Mustersuchalgorithmen implementieren.

Jede Klasse besitzt eine statische und öffentliche Methode `search`. Dieser Methode werden der zu durchsuchende Text, das gesuchte Muster und die Startposition der Mustersuche im Text (Defaultwert 0) übergeben.

Instrumentieren Sie Ihre Mustersuchalgorithmen so, dass mitgezählt werden kann, wie viele Operationen in einer konkreten Suchsituation durchgeführt werden. Dazu übergeben Sie den `search`-Methoden ein Objekt der Klasse `data_collector`. Dieser „Datensammler“ zeichnet während einer Mustersuche die Anzahl der durchgeführten Zeichenvergleiche auf; außerdem zeichnet die Klasse `data_collector` für jedes Zeichen des zu durchsuchenden Textes auf, wie oft es erfolgreich und wie oft es erfolglos verglichen wurde. Ein Datensammler kann alle aufgezeichneten Daten in eine csv-Datei schreiben. Dadurch können in weiterer Folge Laufzeitanalysen und Laufzeitvergleiche durchgeführt werden.

Analysieren Sie das Verhalten Ihrer drei Mustersuchalgorithmen für

- das binäre Alphabet (zwei Symbole),
- das Alphabet der Basenpaare (vier Symbole),
- das Alphabet der Aminosäuren (23 Symbole) und
- den ASCII-Zeichensatz (256 Symbole).

Bestätigen oder widerlegen Sie experimentell die in der Vorlesung genannten Eigenschaften der einzelnen Mustersuchalgorithmen. Verwenden Sie dazu auch einen Zufallszahlengenerator.

Beschreiben Sie in einem Analysedokument, welche Versuche Sie durchgeführt haben und was die Versuche ergaben.

Stellen Sie Ihre Analyseergebnisse in geeigneter Form graphisch dar.