

Spamerkennung

Eine mögliche Art der Spamerkennung ist es, Emails nach bestimmten Wörtern zu durchsuchen die üblicherweise in Spam-Emails vorkommen, und je nachdem wie groß der Anteil solcher Wörter ist, die Email als Spam zu markieren. Leider haben die Autoren der Spam-Emails darauf reagiert und ändern die Schreibweise solcher Wörter ab, z. B. steht dann in der Email "V!4.G.R4" statt "Viagra".

Um weiterhin solche Wörter in den Emails zu erkennen, soll approximatives Stringmatching eingesetzt werden. Dabei gilt ein Teilwort $T[i..j]$ des Texts als Match für das Pattern P , wenn die minimalen Kosten, um das Teilwort $T[i..j]$ in das Pattern P zu ändern, kleiner oder gleich einer Konstanten D sind. Für unseren Fall betrachten wir folgende Kosten für die Änderungen:

1. Einfügen eines Zeichens: 4
2. Änderung Groß-/Kleinschreibung: 0
3. Ersetzen eines Nicht-Buchstaben durch einen Buchstaben: 2
4. Beliebiges Ersetzen eines Zeichens: 4
5. Löschen eines Zeichens: 1

So sind zum Beispiel die Kosten, um "V!4.G.R4" in "Viagra" zu überführen, 8. Offensichtlich ist das gewählte Kostenmaß nicht besonders gut, für bessere Ergebnisse müssten die Kosten für Ersetzungen für jede Art von Ersetzung einzeln festgelegt werden. Zur Vereinfachung verzichten wir bei dieser Aufgabe darauf.

Eingabe

Die erste Zeile der Eingabe enthält das Wort, das gesucht werden soll. Die zweite Zeile der Eingabe enthält die Konstante D , die angibt, wie groß die Änderungskosten bei einem Match maximal sein dürfen. Die dritte Zeile enthält den Text einer Email. Sie können annehmen, dass das gesuchte Wort maximal Länge 12 hat, und dass der Text maximal Länge 100000 hat.

Ausgabe

Geben Sie "Spam" aus, wenn es ein Teilwort $T[i..j]$ des Textes gibt, das mit maximal Kosten D in das gesuchte Wort geändert werden kann. Falls es kein solches Teilwort gibt, geben Sie "Kein Spam" aus.

Beispiel

Eingabe:

Viagra

8

Mann muss ja keine Problem mit der Standkraft haben. Dieses natürliche V!4.G.R4 ist für jedermann!

Ausgabe:

Spam