

Junioraufgabe 1: Reimerei

J1.1 Lösungsidee

Für jedes Paar von Wörtern (w_1, w_2) muss geprüft werden, ob die drei gegebenen Voraussetzungen erfüllt werden.

Dafür muss zunächst für jedes Wort die jeweils maßgebliche Vokalgruppe bestimmt werden. Hierzu kann das gesamte Wort durchlaufen und nach allen möglichen Vokalgruppen durchsucht werden. Wichtig dabei ist, dass es um *Vokalgruppen* geht. Aufeinanderfolgende Vokale (z. B. *Wiese*) müssen also auch als Gruppe betrachtet werden. Von allen gefundenen Vokalgruppen kann dann die vorletzte – sofern es mindestens zwei Vokalgruppen gibt – oder die einzige – falls es nur eine Vokalgruppe gibt – als die maßgebliche Vokalgruppe gespeichert werden. Wurde überhaupt keine Vokalgruppe gefunden, kann das Wort aus der Liste der zu betrachtenden Wörter gestrichen werden. Es genügt, den Index, also die Position im Wort, der maßgeblichen Vokalgruppe im Wort zu speichern. Natürlich kann zusätzlich aber auch die Vokalgruppe an sich gespeichert werden.

Nun können die einzelnen Regeln mithilfe der beiden berechneten maßgeblichen Vokalgruppen überprüft werden. Dazu können beispielsweise beide Wörter rückwärts, beginnend beim letzten Buchstaben, Buchstabe für Buchstabe verglichen werden, sodass bekannt ist, ab welchem Index sich beide Wörter gleichen. Sei $i_{1 \rightarrow 2}$ der kleinste Index, ab welchem w_1 dem Ende des Wortes w_2 gleicht, und entsprechend $i_{2 \rightarrow 1}$ der kleinste Index, ab welchem w_2 dem Wort w_1 gleicht.¹ Mit dieser Information und den Indizes vok_1, vok_2 , an denen die maßgeblichen Vokalgruppen in w_1 und w_2 starten, ist es möglich, alle Regeln schnell zu überprüfen.

1. Regel: Beide Wörter enden gleich.

Dazu wird für jedes der beiden Wörter geprüft, ob der Index des Starts der maßgeblichen Vokalgruppe größer oder gleich dem Index ist, ab dem das Wort dem anderen gleicht. Damit gleichen sie sich mindestens ab der maßgeblichen Vokalgruppe (und ggf. auch noch davor):

$$i_{1 \rightarrow 2} \leq vok_1 \text{ und } i_{2 \rightarrow 1} \leq vok_2$$

2. Regel: Die maßgebliche Vokalgruppe und was ihr folgt enthält mindestens die Hälfte der Buchstaben.

Dazu wird für jedes der beiden Wörter geprüft, ob der (*nullbasierte*) Index des Starts der maßgeblichen Vokalgruppe kleiner oder gleich der Hälfte der Länge des dazugehörigen Wortes ist:

$$vok_1 \leq \frac{l_1}{2} \text{ und } vok_2 \leq \frac{l_2}{2}$$

3. Regel: Keines der beiden Wörter darf mit dem kompletten anderen Wort enden.

Dazu wird für jedes der beiden Wörter geprüft, ob der Index, ab dem das Wort dem anderen gleicht, größer als 0 ist. Ist das bei einem der beiden Wörter nicht der Fall, bedeutet das, dass das andere Wort mit diesem endet. Wir prüfen also:

$$i_{1 \rightarrow 2} > 0 \text{ und } i_{2 \rightarrow 1} > 0$$

¹Sind l_1 und l_2 die Längen der Wörter, so gilt also $l_1 - i_{1 \rightarrow 2} = l_2 - i_{2 \rightarrow 1}$.

Werden alle drei Regeln erfüllt, wird das aktuell betrachtete Wortpaar als passend im Sinne der Aufgabenstellung ausgegeben.

J1.2 Optimierung

Es ist sinnvoll, die maßgebliche Vokalgruppe eines Wortes nicht für jeden Vergleich des Wortes mit einem anderen Wort neu zu berechnen. Effizienter ist es, zu Beginn des Programms ein Mal die maßgebliche Vokalgruppe jedes Wortes vorzuberechnen und zu speichern, sodass später bei den Vergleichen von zwei Wörtern darauf zurückgegriffen werden kann.

J1.3 Sonderfälle

Die Aufgabenstellung legt nicht fest, welche Buchstaben als Vokale gelten und daher Vokalgruppen bilden können. Wir betrachten selbstverständlich die Buchstaben a, e, i, o, u, aber auch ä, ö und ü als Vokale. Sonderfälle wie qu oder y als Vokale zu zählen wäre ebenfalls denkbar, wir entscheiden uns aber dagegen.

Abkürzungen wie LKW (Lastkraftwagen) stellen einen weiteren Sonderfall dar. In diesen Wörtern sind keine Vokale und dementsprechend sind die Regeln der Reimerei nicht auf sie anwendbar. Wir sortieren diese Wörter daher einfach aus.

Bei Wörtern mit Bindestrich, wie zum Beispiel U-Bahn, sind ebenfalls mehrere Vorgehen denkbar. Wie entscheiden uns dafür, dieses Wort nach den ganz normalen Regeln zu behandeln, sodass das „U“ in diesem Fall die maßgebliche Vokalgruppe ist.

J1.4 Laufzeit

Sei n die Anzahl der Wörter und m die Länge des längsten Wortes.

Da alle Paare von Wörtern geprüft werden müssen, benötigt das eine Laufzeit von $\mathcal{O}(n^2)$. Das Vergleichen der Wortenden der zwei aktuell betrachteten Wörtern hat eine Laufzeit von $\mathcal{O}(m)$, da spätestens dann der Vergleich endet. Um die maßgebliche Vokalgruppe eines Wortes zu berechnen, wird das gesamte Wort durchlaufen, was wieder eine Laufzeit von $\mathcal{O}(m)$ hat.

Für das Vorberechnen der maßgeblichen Vokalgruppen und erst darauf folgende Vergleichen der Wörter wird damit eine Laufzeit von $\mathcal{O}(n^2 \cdot m)$ erreicht, denn das Vorberechnen hat eine Laufzeit von $\mathcal{O}(n \cdot m)$ und das folgende Vergleichen eine Laufzeit von $\mathcal{O}(n^2 \cdot m)$.

Ohne die Vorberechnung würde die asymptotische Laufzeit $\mathcal{O}(n^2 \cdot (2 \cdot m + m)) = \mathcal{O}(n^2 \cdot m)$ betragen, denn in jedem Vergleich müsste für zwei Wörter die maßgebliche Vokalgruppe berechnet und diese Vokalgruppen dann verglichen werden. Die asymptotische Laufzeit wäre also nicht schlechter, das Programm jedoch tatsächlich langsamer, da mehr Instruktionen in jedem Schleifendurchlauf durchgeführt würden.

J1.5 Beispiele

reimerei0.txt

bemühen	glühen
biene	hygiene
biene	schiene
hygiene	schiene
knecht	recht

reimerei1.txt

bildnis	wildnis
brote	note

reimerei2.txt

epsilon	ypsilon
---------	---------

reimerei3.txt

absender	kalender	ansage	frage	ansage	garage
bahn	zahn	bank	dank	baum	raum
bein	wein	bier	tier	bild	schild
bitte	mitte	butter	großmutter	butter	mutter
dame	name	dezember	november	dezember	september
drucker	zucker	durst	wurst	ermäßigung	kündigung
ermäßigung	reinigung	fest	test	feuer	steuer
fisch	tisch	flasche	tasche	frage	garage
fuß	gruß	gas	glas	glück	stück
gleis	kreis	gleis	preis	gleis	reis
gruppe	suppe	hand	land	hand	strand
hose	rose	hund	mund	kündigung	reinigung
kanne	panne	kasse	klasse	kasse	tasse
kassette	kette	kassette	toilette	keller	teller
kette	toilette	kind	rind	kind	wind
klasse	tasse	kopf	topf	kreis	preis
kunde	stunde	land	strand	lohn	sohn
magen	wagen	nachmittag	vormittag	november	september
platz	satz	rind	wind	rock	stock
s-bahn	zahn	sache	sprache	sekunde	stunde
see	tee				

Anmerkung: reimerei3.txt war in dem hochgeladenen zip-Archiv leicht anders als bei den einzeln unter Junioraufgabe 1 gelisteten Dateien. Je nachdem, welche Version verwendet wurde, ist (see, tee) bei den Reimpaaren enthalten oder nicht.

J1.6 Bewertungskriterien

Die Bewertungskriterien vom Bewertungsbogen werden hier erläutert (Punktabzug in []).

- [−1] **Regeln werden nicht eingehalten**

Es müssen alle drei Regeln zur Überprüfung eines Reimpaars beachtet werden:

- Ab der maßgeblichen Vokalgruppe enthalten beide Wörter dieselben Buchstaben in derselben Reihenfolge.
- Der Teil ab der maßgeblichen Vokalgruppe umfasst jeweils mindestens die Hälfte der Buchstaben.
- Keines der beiden Wörter darf mit dem kompletten anderen Wort enden.

- [−1] **Lösungsverfahren anderweitig fehlerhaft**

Insbesondere muss darauf geachtet werden, dass die maßgebliche Vokalgruppe korrekt bestimmt wird. Sofern es mindestens zwei Vokalgruppen in einem Wort gibt, muss die vorletzte als maßgebliche verwendet werden.

Die Buchstaben a, e, i, o und u sollten auf jeden Fall als Vokale gezählt werden. Umlaute nicht als Vokale zu zählen, führt nicht zu Punktabzug. Sonderfälle wie qu oder y als Vokale zu zählen, ist erlaubt und richtig, sofern dies in der Dokumentation entsprechend begründet wurde.

Wie Abkürzungen wie LKW behandelt werden, ist gänzlich freigestellt, solange die anderen Wortpaare korrekt geprüft werden.

- [−1] **Ergebnisse schlecht nachvollziehbar**

Alle Reimpaare sollten erkennbar und voneinander getrennt ausgegeben werden. Es ist dabei in Ordnung, wenn dasselbe Paar zwei Mal, aber in unterschiedliche Richtungen ausgegeben wird (z. B. recht – knecht, knecht – recht). Die Liste muss allerdings angemessen sortiert sein (z. B. lexikographisch). Ist dies nicht der Fall, darf hier ein Punkt abgezogen werden.

- [−1] **Beispiele fehlerhaft bzw. zu wenige oder ungeeignete Beispiele**

Die Dokumentation soll Ergebnisse zu allen vorgegebenen Beispielen (reimerei0.txt bis reimerei3.txt) enthalten.