

Jugendwettbewerb Informatik 2019

Runde 3

Junioraufgabe 1: Parallelen

25. November 2019

Inhaltsverzeichnis

Lösungsidee.....	1
Umsetzung.....	2
Beispiel/Ausgabe.....	2
Quellcode.....	5

Lösungsidee

Martin hat behauptet, dass das Gedicht „Parallelen“ von Christian Morgenstern so aufgebaut ist, dass egal mit welchem Wort der ersten Gedichtshälfte man anfängt, man nach Wortsprüngen nach bestimmtem Muster immer beim gleichen Wort endet. Man soll immer genau so viele Wörter weiterspringen, wie das Wort, von dem aus gesprungen wird Buchstaben hat, bis Weiterspringen nicht mehr möglich ist.

Ein Programm, das seine Behauptung testet sollte:

- das Gedicht einlesen
- Satzzeichen entfernen (die gehören nicht zur Wortlänge und stören beim Zählen)
- Leerzeilen entfernen (sind keine Wörter und stören beim Zählen)
- jedes Wort der ersten Hälfte des Gedichts einmal als Startwort wählen
- die Buchstaben des Wortes zählen von dem aus gesprungen wird
- genau so viele Wörter im Text weiterspringen wie Buchstaben im Wort waren
- vom erreichten Wort wieder weiterspringen

Junioraufgabe 1: Parallelen

- die Sprungfolge beenden, wenn nicht mehr genug Wörter im Gedicht vorhanden sind, um nochmal zu springen
- die in jeder Sprungfolge erreichten Wörter zusammen mit ihren Buchstabennummern ausgeben
- die Endwörter miteinander vergleichen und dann feststellen, ob die Sprungfolgen immer mit dem selben Wort enden. Falls ja, hatte Martin Recht, sonst nicht. Das Programm soll dann eine entsprechende Meldung ausgeben.

Umsetzung

Das Programm habe ich mit Blockly geschrieben, da ich leider noch keine andere Programmiersprache kann.

Ich habe die zur Verfügung gestellte Eingabe-/Ausgabe Umgebung verwendet. Das Programm sind also die gespeicherten Blöcke als Datei (**Parallelen**), die in die Umgebung geladen werden müssen. Das Gedicht muss aus der Textdatei in das Eingabefenster kopiert werden.

Bitte nicht vergessen, die Eingabe mit der „enter“ Taste abzuschließen, da sonst die letzte Gedichtszeile nicht eingelesen wird.

Das Gedicht wird zeilenweise eingelesen. Die Funktion **satzzeichen weg** entfernt die Satzzeichen (',', ';', ':', ':', '?'). Auch die Leerzeilen ('\n') und ('-') werden entfernt und die Wörter anschließend in einer Liste **textliste** gespeichert.

Das Wort „hin“ (letztes Wort der ersten Texthälfte), wird in der **textliste** gesucht und seine Position in der Variable **index_hin** gemerkt.

Dann definiert das Programm jedes Wort der ersten Gedichtshälfte einmal als Anfangswort (**anfangswort**) und führt die Sprungfolgen durch. Die jeweiligen Sprungweiten (**sprung**) werden durch Buchstabennummern ermittelt und entsprechend viele Wörter in der **textliste** weiterspringen. Die erreichten Wörter werden in der Variable **ergebnis** als Liste gespeichert. Die am Ende der Sprungfolgen erreichten Wörter werden in der Liste **endwörter** gespeichert.

Die Funktion **ausgeben** gibt am Ende einer Sprungfolge die in der Folge erreichten Wörter zusammen mit ihren Buchstabennummern aus.

In der Funktion **hat er recht?**, wird am Ende des Programms überprüft alle Sprungfolgen immer mit dem selben Wort enden und eine entsprechende Nachricht ausgegeben.

Beispiel/Ausgabe

Das hat mein Programm ausgegeben:

Es(2), zwei(4), hinaus(6), solidem(7), bis(3), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

gingen(6), zwei(4), aus(3), Sie(3), nicht(5), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

zwei(4), hinaus(6), solidem(7), bis(3), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

Parallelen(10), Haus(4), nicht(5), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

ins(3), zwei(4), aus(3), Sie(3), nicht(5), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

Junioraufgabe 1: Parallelen

Endlose(7), solidem(7), bis(3), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

hinaus(6), solidem(7), bis(3), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

zwei(4), aus(3), Sie(3), nicht(5), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

kerzengerade(12), an(2), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

Seelen(6), wollten(7), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

und(3), Haus(4), nicht(5), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

aus(3), Sie(3), nicht(5), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

solidem(7), bis(3), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

Haus(4), nicht(5), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

Sie(3), nicht(5), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

wollten(7), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

sich(4), an(2), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

nicht(5), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

schneiden(9), einmal(6), Stab(4), zehn(4), sich(4), dem(3), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

bis(3), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

an(2), seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

ihr(3), Das(3), einmal(6), Stab(4), zehn(4), sich(4), dem(3), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

seliges(7), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

Grab(4), einmal(6), Stab(4), zehn(4), sich(4), dem(3), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

Das(3), einmal(6), Stab(4), zehn(4), sich(4), dem(3), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

Junioraufgabe 1: Parallelen

war(3), der(3), Stolz(5), sie(3), gewandert(9), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

nun(3), beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

einmal(6), Stab(4), zehn(4), sich(4), dem(3), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

der(3), Stolz(5), sie(3), gewandert(9), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

beiden(6), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

geheimer(8), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

Stolz(5), sie(3), gewandert(9), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

und(3), als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

Stab(4), zehn(4), sich(4), dem(3), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

Doch(4), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

als(3), Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

sie(3), gewandert(9), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

zehn(4), sich(4), dem(3), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

Lichtjahre(10), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

gewandert(9), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

neben(5), dem(3), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

sich(4), dem(3), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

hin(3), dem(3), nicht(5), Warn(4), Sie(3), nicht(5), zwei(4), ewiges(6), sie(3), sie(3), ihm(3), verschlang(10)

Martin hat recht, man landet immer wieder auf dem selben Wort, egal wo man anfängt.

Das Endwort ist immer "verschlang"

Quellcode

Hier die Blöcke von meinem Blockly – Programm. Sie sind in der Datei [Parallelen](#) gespeichert.

```
graph TD
    Start[hat er recht?] --> SetzeRichtig[setze richtig auf wahr]
    SetzeRichtig --> SetzeEndwort[setze endwort auf in der Liste endwortliste nimm #tes 1]
    SetzeEndwort --> Loop[für jeden Wert o aus der Liste endwortliste]
    Loop --> LoopBody[mache falls o ≠ endwort]
    LoopBody --> SetzeUnwahr[mache setze richtig auf unwahr]
    LoopBody --> Loop
    Loop --> LoopEnd[falls richtig]
    LoopEnd --> LoopEndBody[mache schreibe "Martin hat recht, man landet immer wieder auf de..."]
    LoopEndBody --> LoopEndBody2[schreibe erstelle Text aus "Das Endwort ist immer" " " endwort " " ]
    LoopEndBody2 --> LoopEndBody3[sonst schreibe "Martin hat unrecht"]
```

```
graph TD
    Start[ausgeben] --> SetzeAusgabertext[setze ausgabertext auf " "]
    SetzeAusgabertext --> Zaehle[zähle n von 1 bis Länge von ergebnis in Schritten von 1]
    Zaehle --> Loop[mache an ausgabertext Text anhängen]
    Loop --> LoopBody[erstelle Text aus in der Liste ergebnis nimm #tes n]
    LoopBody --> LoopBody2[Länge von in der Liste ergebnis nimm #tes n]
    LoopBody2 --> LoopBody3[schreibe ausgabertext]
```

```
graph TD
    Start[sätzezeichen weg] --> Loop1[falls in der Liste zeilenliste nimm letztes = " - "]
    Loop1 --> Loop1Body[mache in der Liste zeilenliste entferne letztes]
    Loop1Body --> Loop1
    Loop1 --> Loop2[zähle m von 1 bis Länge von zeilenliste in Schritten von 1]
    Loop2 --> Loop2Body[mache setze wort auf Liste aus Text erstellen in der Liste zeilenliste nimm #tes m mit Trennzeichen " " ]
    Loop2Body --> Loop2Body2[setze buchstabe auf in der Liste wort nimm letztes]
    Loop2Body2 --> Loop2Body3[für jeden Wert i aus der Liste zeichenliste]
    Loop2Body3 --> Loop2Body3Body[mache falls buchstabe = i]
    Loop2Body3Body --> Loop2Body3Body2[mache in der Liste wort entferne letztes]
    Loop2Body3Body2 --> Loop2Body3Body
    Loop2Body3Body --> Loop2Body3
    Loop2Body3 --> Loop2Body3End[in der Liste zeilenliste setze für #tes m ein Text aus Liste erstellen wort mit Trennzeichen " " ]
```

Junioraufgabe 1: Parallelen

```
setze textliste auf erzeuge eine leere Liste
wiederhole bis Ende der Eingabe
mache
  setze zeile auf lies Zeile als Text
  setze zeilenliste auf Liste aus Text erstellen zeile mit Trennzeichen " "
  setze zeichenliste auf erzeuge Liste mit " "
  satzzeichen weg
  falls Länge von zeilenliste > 1
  mache
    für jeden Wert k aus der Liste zeilenliste
    mache
      in der Liste textliste füge als letztes ein k
  setze index_hin auf in der Liste textliste suche erstes Auftreten von "hin"
  setze endwortliste auf erzeuge eine leere Liste
  zähle j von 1 bis index_hin in Schritten von 1
  mache
    setze ergebnis auf erzeuge eine leere Liste
    setze anfangswort auf in der Liste textliste nimm #tes j
    in der Liste ergebnis füge als letztes ein anfangswort
    setze sprung auf Länge von anfangswort
    setze zähler auf j + sprung
    wiederhole solange zähler < Länge von textliste
    mache
      setze wort1 auf in der Liste textliste nimm #tes zähler
      in der Liste ergebnis füge als letztes ein wort1
      erhöhe zähler um Länge von wort1
    in der Liste endwortliste füge als letztes ein in der Liste ergebnis nimm letztes
  ausgeben
  schreibe " "
hat er recht?
```