

# Aufgabe 1: Störung

Team-ID: 00944

Team: Dominik

Bearbeiter/-innen dieser Aufgabe:

Dominik Korolko

18. November 2022

## Inhaltsverzeichnis

<b>Lösungsidee</b>	<b>2</b>
Umsetzung	2
<b>Beispiele</b>	<b>3</b>
Beispiel 1 - stoerung0.txt	3
Beispiel 2 - stoerung1.txt	3
Beispiel 3 - stoerung2.txt	3
Beispiel 4 - stoerung3.txt	3
Beispiel 5 - stoerung4.txt	4
Beispiel 6 - stoerung5.txt	4
Eigene Beispiele	4
Eigenes Beispiel 1 - extra_stoerung0.txt	4
Eigenes Beispiel 2 - extra_stoerung1.txt	4
Eigenes Beispiel 3 - extra_stoerung2.txt	5
<b>Quellcode</b>	<b>5</b>

## Lösungsidee

Für das Finden der Textstelle mit dem angegebenen Lückentext muss der Text auf Stellen durchsucht werden, die mit den angegebenen Wörtern des Lückentextes übereinstimmen und in denen für jeden Unterstrich im Lückentext ein Wort vorhanden ist. Wörter sind als Textteile definiert, welche von mindestens einem Leerzeichen oder einem Zeilenumbruch umgeben sind.

## Umsetzung

Um das Arbeiten mit dem Text zu vereinfachen, wird dieser zuerst in eine eindimensionale Liste umgewandelt, in der jedes Wort ein Element darstellt. Dieser Schritt erlaubt es, Textpassagen zu finden, die sich über Zeilenumbrüche erstrecken.

Als erstes iteriert das Programm über alle Wörter im ganzen Text. Dabei wird stets das jeweilige Wort mit dem ersten Wort des Lückentextes verglichen. Wenn sie übereinstimmen, oder das Wort des Lückentextes ein Unterstrich ist, iteriert das Programm über alle Wörter des Lückentextes. Dabei werden Unterstriche mit der continue-Anweisung sofort übersprungen. Wenn nun jedes der Wörter des Lückentextes in Reihenfolge mit den Wörtern des Textes übereinstimmt, wird die Textpassage gespeichert und am Ende ausgegeben. Wenn ein Wort während der Iteration nicht mit dem Lückentext übereinstimmt, wird die innere Iteration abgebrochen und das Programm macht in der äußeren Iteration mit dem nächsten Wort des Textes weiter.

Für den Vergleich werden sowohl die Wörter aus dem Text als auch aus dem Lückentext in eine kleingeschriebene Version umgewandelt, die nur aus den alphanumerischen Zeichen (also Zahlen und Buchstaben) des Originalwortes bestehen. Dies lässt einen Vergleich zu, selbst wenn Sonderzeichen wie Fragezeichen oder Anführungszeichen am Wort stehen. In der Ausgabe wird aber trotzdem die Originalversion der Wörter verwendet, um die Suche nach der Passage in Texteditoren zu ermöglichen.

## Beispiele

### Beispiel 1 - stoerung0.txt

**Eingabe:**

*das \_ mir \_ \_ \_ vor*

**Ausgabe:**

*Alle gefundenen Passagen:*

*»Das kommt mir gar nicht richtig vor,«*

### Beispiel 2 - stoerung1.txt

Das Programm findet nicht nur die erste passende Textstelle, sondern gibt alle passenden Passagen im Text aus.

**Eingabe:**

*ich muß \_ clara \_*

**Ausgabe:**

*Alle gefundenen Passagen:*

*Ich muß in Clara verwandelt*

*»Ich muß doch Clara sein,*

### Beispiel 3 - stoerung2.txt

**Eingabe:**

*fressen \_ gern \_*

**Ausgabe:**

*Alle gefundenen Passagen:*

*»Fressen Katzen gern Spatzen?*

*Fressen Katzen gern Spatzen?*

*Fressen Spatzen gern Katzen?«*

### Beispiel 4 - stoerung3.txt

**Eingabe:**

*das \_ fing \_*

**Ausgabe:**

*Alle gefundenen Passagen:  
das Spiel fing an.  
'Das Publikum fing an,*

Beispiel 5 - stoerung4.txt

**Eingabe:**

*ein \_ \_ tag*

**Ausgabe:**

*Alle gefundenen Passagen:  
ein sehr schöner Tag!«*

Beispiel 6 - stoerung5.txt

**Eingabe:**

*wollen \_ so \_ sein*

**Ausgabe:**

*Alle gefundenen Passagen:  
»Wollen Sie so gut sein,*

## **Eigene Beispiele**

Eigenes Beispiel 1 - extra\_stoerung0.txt

Bei leerer Eingabe weist das Programm auf einen Fehler hin.

**Eingabe:**

*[zwei leere Zeilen]*

**Ausgabe:**

*Die Eingabe ist leer. Bitte überprüfen Sie die Eingabedatei.*

Eigenes Beispiel 2 - extra\_stoerung1.txt

Das Programm findet die Textstelle, obwohl sie über zwei Zeilen geht.  
Diese Passage befindet sich in den Zeilen 32/33.

**Eingabe:**

*goldner \_ \_ \_ und \_ lacht*

**Ausgabe:**

*Alle gefundenen Passagen:*

*goldner Nachmittag, Wo Flut und Himmel lacht!*

Eigenes Beispiel 3 - extra\_stoerung2.txt

Das Programm findet Textstellen auch, wenn der Lückentext mit einem Unterstrich beginnt.

Diese Passage befindet sich in der Zeile 34.

**Eingabe:**

*\_ schwacher \_ bewegt*

**Ausgabe:**

*Alle gefundenen Passagen:*

*Von schwacher Kindeshand bewegt,*

## Quellcode

```
def find_all(search_term: str, text: str):
    text = text.split()
    terms = search_term.split()
    result = []
    for idx, word in enumerate(text):
        if get_lower_letters_only(word) == get_lower_letters_only(terms[0]) or
terms[0] == "_":
            found_passage = True
            for idx2, term in enumerate(terms):
                if term == "_":
                    continue
                try:
                    is_match = get_lower_letters_only(term) ==
get_lower_letters_only(text[idx + idx2])
                except IndexError:
                    found_passage = False
                    break
                if not is_match:
                    found_passage = False
                    break
            if found_passage:
```

```
        result.append(" ".join(text[idx:idx + (len(terms))]))  
    return result
```

```
def get_lower_letters_only(word: str):  
    return "".join([letter for letter in word if letter.isalpha()]).lower()
```