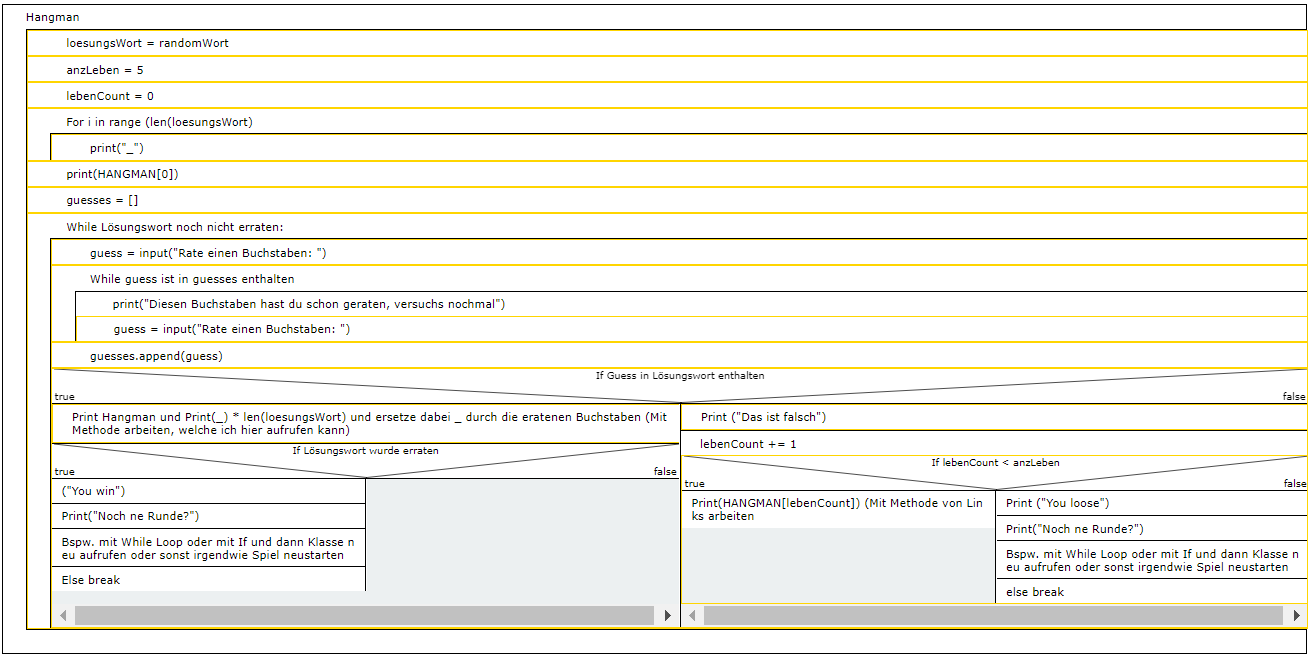
# Projekt Hangman Dokumentation

## Informieren

Ziel ist es, ein funktionierendes Hangman-Spiel im Sinne des vorgegebenen Flussdiagramms zu erstellen und allenfalls noch Erweiterungen umzusetzen.

## Planen

Zur Umsetzung habe ich nun ein Struktogramm erstellt. Ich werde vorerst sicherlich die ASCII-Hangmans verwenden und allenfalls später noch mit Turtle das ganze umsetzen.

## Realisieren

Grobe Form von Struktogramm umsetzen und bei non-Code von oben zuerst Recherche betreiben.

**Spiel wiederholen:** Alles in main einfügen und dann separate Methode im Sinne dieser Abbildung Ein Bild, das Text, Schrift, Screenshot enthält.

Automatisch generierte Beschreibung

**Ist Buchstabe in Lösungswort enthalten:**

if loesungsWort.find(guess) >= 0:

Code für wenn Buchstabe vorhanden

**Ist Lösungswort erraten:** Mit folgender Methode in einer if true Bedingung arbeiten

Bool IstErraten():

counter = 0

for i in range (len(loesungsWort)):

if loesungsWort.find(guesses[i]) >= 0:

counter += 1

if counter == len(loesungsWort):

return true

else:

return false

Zwischenstandsfazit: Planung bereits weit fortgeschritten. Nun geht es weiter damit, das Struktogramm zu analysieren und Pseudecode finden und dahinterliegendes Problem beheben (Bspw. Methode für Grafik-Output definieren)

#Methode für wie viele Buchstaben bereits erraten wurden

Int wieVieleBuchstabenErraten():

counter = 0

for i in range(len(loesungsWort)):

if loesungsWort.find(guesses[i]) >= 0:

counter += 1

return counter

String grafikOutput():

print(HANGMAN[lebenCount])

for i in range(len(loesungsWort)):

if loesungsWort.find(guesses[i]) >= 1:

print guesses[i]

else:

print(“\_”)