**Junioraufgabe 1: Zum Winde verweht**

Team-ID: 00450 Team: Die Lamas

Bearbeiter/-innen dieser Aufgabe:

Philipp Tiede

7. September 2020

**Inhaltsverzeichnis**

[Lösungsidee 1](#_Toc87791651)

[Umsetzung 1](#_Toc87791652)

[Beispiele 1](#_Toc87791653)

[Quellcode 2](#_Toc87791654)

# Lösungsidee

In dieser Aufgabe muss man für einen Landkreis die Maximalhöhe von Windkraftwerken herausfinden, welche mit dem Abstand zu Häusern in diesem Landkreis zusammenhängt.

Die Maximalhöhe von einem Windrad lässt sich deshalb so herausfinden, dass man von diesem Windrad den Abstand zu jedem Haus berechnet und dann diesen Abstand durch 10 als Maximalhöhe zurückgibt.

# Umsetzung

Dies setzte ich in Python so um, dass ich erst alle Positionen (als Tuple), von Häusern und Windrädern aus der Datei in 2 Listen speichere.

# Beispiele

Beispiele

# Quellcode

**def** main():

*"""*

*main*

*"""*

dataFileLocation = sys.argv[1] *#*

dataFile = open(dataFileLocation, "r") *# Datei einlesen und in data als String*

data = dataFile.read() *# speichern.*

houses:List[Tuple] = []

winds:List[Tuple] = []

windHeight:List[int] = []

**for** idx, building **in** enumerate(data.splitlines()):

**if** idx == 0:

houseNum = int(building.split(" ")[0])

windNum = int(building.split(" ")[1])

**elif** idx <= houseNum:

houses.append((int(building.split(" ")[0]), int(building.split(" ")[1])))

**else**:

winds.append((int(building.split(" ")[0]), int(building.split(" ")[1])))

**for** idx, wind **in** enumerate(winds):

distances = []

**for** house **in** houses:

distance = (wind[0] - house[0], wind[1] - house[1])

distances.append(math.sqrt((distance[0]\*\*2)+(distance[1]\*\*2)))

windHeight.append(min(distances)/10)

**for** idx, height **in** enumerate(windHeight):

**print**("Die " + str(idx + 1) + ". Windturbine kann ca. " + str(round(height)) + " meter hoch sein.")