

Junioraufgabe 1: Zum Winde verweht

Team-ID: 00833

Team: Gewinner bwinf

Bearbeiter/-innen dieser Aufgabe:
Jonathan Rajewicz

21. November 2021

Inhaltsverzeichnis

| | |
|------------------|---|
| Lösungsidee..... | 1 |
| Umsetzung..... | 1 |
| Beispiele..... | 2 |
| Quellcode..... | 2 |

Lösungsidee

Um die maximale Höhe eines Windrades zu berechnen, muss die Distanz zu allen Häusern errechnet werden und die kleinste Distanz durch 10 dividiert werden. Die Distanz kann mithilfe des Satzes von Pythagoras ermittelt werden, indem die x-Koordinate eines Windrades mit der eines Hauses subtrahiert und dann quadriert wird, was zur Quadrierten Differenz der y-Koordinaten von Windrad und Haus addiert wird und dann eine Wurzel gezogen wird:

$$Distanz = \sqrt{(xKoordinate\ Windrad - xKoordinate\ Haus)^2 + (yKoordinate\ Windrad - yKoordinate\ Haus)^2}$$

Für ein Windrad müssen alle Häuser in diese Formel eingesetzt werden und die kleinste Distanz gespeichert werden. Die Höhe des Windrades lässt sich nun mit der umgestellten Gleichung berechnen:

$$maximale\ Höhe = \frac{kleinste\ Distanz}{10}$$

Dieser Vorgang muss für alle Windräder wiederholt werden und die maximalen Höhen der einzelnen Windräder können ausgegeben werden.

Umsetzung

Die Lösungsidee wird in ein Programm in der Sprache Python umgesetzt. Das Mathematical functions (math) Modul stellt das Wurzelziehen zur Verfügung, weshalb es importiert wird. Die

Funktion `hoehe` berechnet für ein Windrad alle möglichen Distanzen mithilfe einer `while`-Schleife, indem alle Häuser durchprobiert werden. Die kleinste Distanz wird mit einem `if`-Statement ermittelt und mit der oben gezeigten Gleichung wird die maximale Höhe berechnet und zurückgegeben. Diese Funktion kann in eine `for`-Schleife eingesetzt werden, die die Funktion für alle Windräder anwendet und die maximalen Höhen speichert.

Beispiele

Zunächst wird das Programm mit den Beispieldaten aus der Aufgabenstellung getestet. Die Eingabedatei `landreis1.txt` sieht so aus:

```
Textdatei mit .txt eingeben: landkreis1.txt
Maximale Höhen:
Windrad 1 = 48.52319033204638 m, Windrad 2 = 158.98003019247417 m, Windrad 3 = 72.41270606737467 m
```

Die Eingabedatei `landreis4.txt` enthält einen Sonderfall und hat diese Ausgabe:

```
Textdatei mit .txt eingeben: landkreis4.txt
Maximale Höhen:
Windrad 1 kann nicht gebaut werden, Windrad 2 = 383.8062271511498 m, Windrad 3 = 262.45491041
319843 m, Windrad 4 = 233.98739282277583 m, Windrad 5 = 296.1940242476205 m, Windrad 6 = 71.75
639344337199 m, Windrad 7 = 181.4135606838695 m, Windrad 8 = 235.40008496175187 m, Windrad 9 =
343.1133923355368 m, Windrad 10 = 177.89617758681607 m, Windrad 11 = 449.1574000280971 m, Win
drad 12 = 408.0275725977351 m, Windrad 13 = 317.9480460704233 m, Windrad 14 = 221.285562113753
8 m, Windrad 15 = 520.1203418440775 m, Windrad 16 = 394.71493511140415 m, Windrad 17 = 433.253
1015468902 m, Windrad 18 = 703.8262072415321 m, Windrad 19 = 168.41674501070253 m, Windrad 20
= 201.26889973366477 m, Windrad 21 = 139.1621356547822 m, Windrad 22 = 348.9877505013607 m, Wi
ndrad 23 = 297.9106073975883 m, Windrad 24 = 110.23107547329838 m, Windrad 25 = 813.6006145523
736 m, Windrad 26 = 236.18918264814755 m, Windrad 27 = 391.94164106407476 m, Windrad 28 = 125.
78346473205451 m, Windrad 29 = 241.00663891270716 m, Windrad 30 = 625.3953069859095 m
```

In diesem Fall ist die Distanz bei Windrad 1 0 Meter zu einem Haus, es würde also auf einem Haus gebaut werden. Das Programm gibt deshalb aus, dass dieses Windrad nicht gebaut werden kann.

Quellcode

```
import math as m # für Wurzel

eingabe = input("Textdatei mit .txt eingeben: ")
eingabe = open(eingabe, "r") # Datei im Read-Modus öffnen

anzahl = eingabe.readline() # erste Zeile einlesen
anzahl = anzahl.split(" ") # einzelne Zahlen zur Liste

anzahlH = int(anzahl[0]) # Anzahl der Häuser = erster str aus Liste und wird in int umgewandelt
anzahlW = int(anzahl[1]) # gleiches für Anzahl Windräder

posGesamt = [] # neue Liste = posGesamt
for zeile in eingabe: # zeilen der Datei einlesen
    posGesamt += zeile.rstrip().split() # eine Zeile wird an Leerzeichen getrennt und zur Liste hinzugefügt

posH = posGesamt[:] #[:] damit posGesamt nicht beeinflusst wird
del posH[anzahlH*2:(anzahlW*2 + anzahlH*2)] # Positionen Windräder werden gelöscht, damit nur Positionen Häuser

posW = posGesamt[:] # gleiches für Positionen von Windrädern
del posW[0:anzahlH*2]
```

```
def hoehe(anzahlH, posH, posW, nummerW): # neue Funktion
    nummerH = 0 # Position des ersten Hauses
    while nummerH < anzahlH*2: # solange nicht alle Häuser getestet
        dis = m.sqrt((int(posW[nummerW])-int(posH[nummerH]))**2+(int(posW[nummerW+1])-int(posH[nummerH+1]))**2)
        # Distanz mit Positionen berechnen
        if nummerH == 0:
            dismin = dis # erste Distanz als kleinste Distanz
        if dis < dismin:
            dismin = dis # kleinste Distanz gespeichert
        nummerH += 2 # nächstes Haus
    h = dismin / 10 # Höhe berechnen
    return h # Höhe ausgeben

nummerW = 0 # erstes Windrad
listeH = "" # "normale" Liste für Höhen
for nummerW in range(anzahlW): # alle Windräder testen
    h = hoehe(anzahlH, posH, posW, nummerW*2) # Funktion wird genutzt um Höhe zu berechnen
    if h <= 0:
        listeH += ("Windrad " + str(nummerW+1) + " kann nicht gebaut werden, ")
        # nicht baubares Windrad wird zur "normalen" Liste hinzugefügt
    else:
        listeH += ("Windrad " + str(nummerW+1) + " = " + str(h) + " m, ")
        # Höhe wird zur "normalen" Liste hinzugefügt
listeH = listeH[:len(listeH)-2] # letztes Komma wird gelöscht
print("Maximale Höhen:\n", listeH)
```