2025/11/24 07:59 1/3 LU09.A07 - Bremsweg

LU09.A07 - Bremsweg



Berechnen Sie die Anhaltestrecke eines Autos bei gegebener Geschwindigkeit [km/h] für trockene bzw. nasse Straße.

Benötigte Informationen

- 1. Die Geschwindigkeit [km/h] kann vom Benutzer eingegeben werden.
- 2. Geschwindigkeit [m/s] = Geschwindigkeit [km/h] / 3.6
- 3. Bremsbeschleunigung auf trockener Straße = $7 [m/s^2]$, auf nasser Straße = $4 [m/s^2]$. Wer das Programm benutzt, kann dabei «t» für die trockene bzw. «n» für die nasse Straße eingeben.
- 4. Reaktionszeit = 1.44 [s].
- 5. Reaktionsweg [m] = Geschwindigkeit [m/s] * Reaktionszeit.
- 6. Bremsweg [m] = Geschwindigkeit 2 [(m/s) 2] / (2 * Bremsbeschleunigung [m/s 2]).
- 7. Anhaltestrecke [m] = Reaktionsweg + Bremsweg.

Zusatzaufgabe Sicherheitsabstand

Berechnen Sie den notwendigen Sicherheitsabstand (in Metern und in Sekunden), um einen Auffahrunfall zu vermeiden, falls der vorausfahrende Wagen eine Vollbremsung einleitet.

1. Bremszeit [s] = Geschwindigkeit [m/s] / Bremsbeschleunigung [m/s^2].

Vorlage

main.py

```
def in_meters_per_second(speed_kmh):
    # Konvertiert Geschwindigkeit von km/h zu m/s
    pass

def reaction_distance(speed_mps):
    # Berechnet den Reaktionsweg
    pass

def braking_distance(speed_mps, dry_road=True):
    # Berechnet den Bremsweg
    pass

def stopping_distance(speed_kmh, dry_road=True):
```

```
# Berechnet die Anhaltestrecke
pass

def safety_distance_meter(speed_kmh, dry_road=True):
    # Berechnet den Sicherheitsabstand in Metern
pass

def safety_distance_seconds(speed_kmh, dry_road=True):
    # Berechnet den Sicherheitsabstand in Metern
pass

def main():
    # TODO: Berechne die Anhaltestrecke und den Sicherheitsabstand für
50, 80 und 120 km/h
    # TODO: Ruf die Funktionen auf und gib das Ergebnis aus
    ...

if __name__ == '__main__':
    main()
```

Beispiele

```
# Beispiel für die Berechnung der Anhaltestrecke bei 80 km/h auf trockener Straße print(stopping_distance(80))
```

Vorgehen

- 1. Akzeptieren Sie das GitHub Classroom Assignment im Moodlekurs.
- 2. Klone Sie das Repository in Ihre Entwicklungsumgebung.
- 3. Implementieren Sie die notwendigen Funktionen zur Berechnung der Anhaltestrecke.
- 4. Testen Sie jede Funktion mit verschiedenen Geschwindigkeiten und Straßenbedingungen.
- 5. überprüfen Sie alles mit den Unittests.
- 6. Rufen Sie die Funktionen in der main-Methode auf und geben Sie die Ergebnisse aus.

Abgabe

Die Abgabe erfolgt durch den Push in das GitHub-Repository. In Moodle ist keine Abgabe vorgesehen oder möglich.

https://wiki.bzz.ch/ Printed on 2025/11/24 07:59

2025/11/24 07:59 3/3 LU09.A07 - Bremsweg

⇒ GitHub Repo für externe Besucher

GitHub Repository https://github.com/templates-python/m319-lu09-a07-breaking

Lernende am BZZ müssen den Link zum GitHub Classroom Assignment verwenden

M319-LU09



BY NO SA © Marcel Suter, Kevin Maurizi

From:

https://wiki.bzz.ch/ - BZZ - Modulwiki

Permanent link:

https://wiki.bzz.ch/de/modul/m319/learningunits/lu09/aufgaben/bremsweg

Last update: 2025/06/23 07:45

