

# Leiteraufgabe

April 1, 2023

```
[1]: # print out each line in the input cell not only the last one
from IPython.core.interactiveshell import InteractiveShell
InteractiveShell.ast_node_interactivity = "all"

# expand the cell width to 100% of t
from IPython.core.display import display, HTML
display(HTML("<style>.container { width:70% !important; }</style>"))
```

<IPython.core.display.HTML object>

Gruppe: Tobias Blümlhuber, Silvan Kron, Simon Gärtner

## 1 Aufgabe 1: Leiteraufgabe

### 1.1 Renovierung eines Gutshofes

Ein alter Gutshof wird renoviert. Dabei soll der Buntsandstein in der Scheune frei gelegt werden, indem der Putz von den 7,80m hohen Wänden abgeschlagen wird. Auf dem Boden der Scheune liegen allerdings noch andere Materialien im Weg, die nicht verschoben werden können. So steht eine würfelförmige Kiste direkt an der Wand mit der Seitenlänge von 1,60m. Reicht eine 6,50m lange Leiter, die an der Wand anlehnt und die Kiste in einem Punkt berührt aus, um den Putz an der kompletten Wand abzuschlagen?

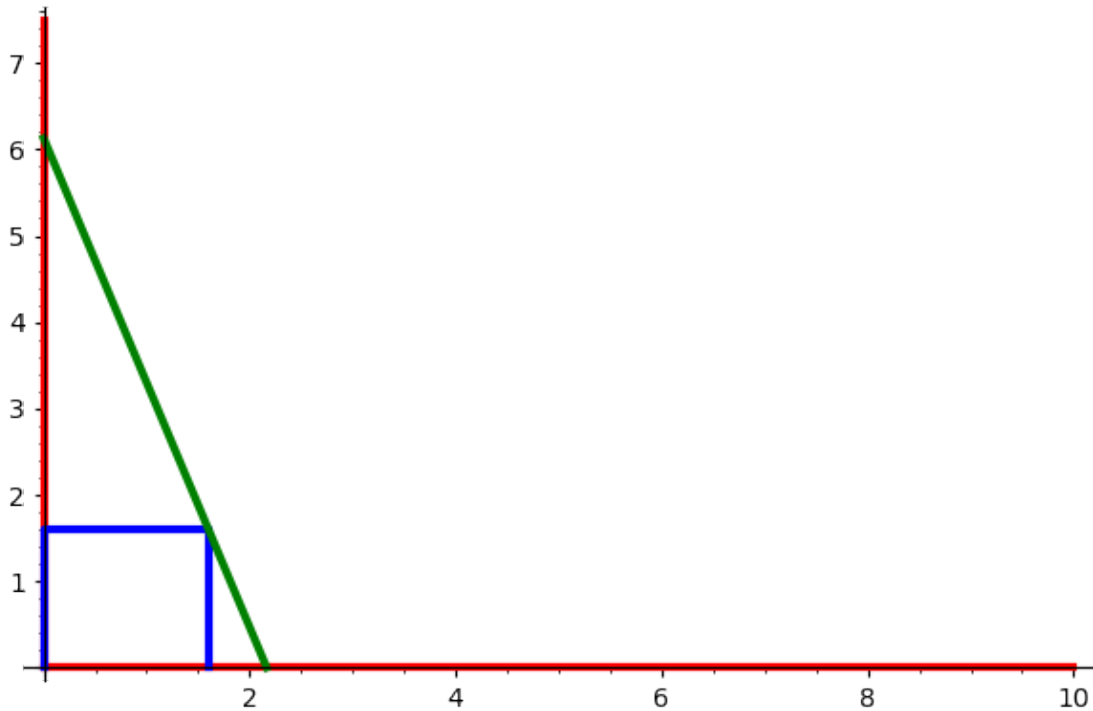
#### 1.1.1 Skizze für besseres Verständnis des Problems

```
[2]: wall = line([(0,0),(0,7.5)], color = 'red', thickness = 3)
floor = line([(0,0),(10,0)], color = 'red', thickness = 3)

chest = line([(0,0), (0,1.6), (1.6, 1.6), (1.6,0)], color = 'blue', thickness = 3)

ladder = line([(2.165,0), (0,6.1287)], color = 'green', thickness = 3)

show(wall + floor + chest + ladder)
```



### Gegebene Maße

Kiste: Würfelförmig 1.60m x 1.60m (blauer Würfel)

Leiter: 6.50m lang, liegt an der Wand an und an einer Ecke des Würfels (grüne Linie)

Wand: 7.80m hoch (rote y-Achse)

[3]: *# define all parameters*

```

wall = 7.80
ladder = 6.50
chest = 1.60
security_distance = 0.30
person_height = 2.10

```

### 1.1.2 Ziel der Berechnung

*Ist die volle Höhe der Wand durch die Leiter und einen darauf stehenden Menschen erreichbar?*

→ Um diese Frage zu beantworten berechnen wir den höchsten Punkt an dem die Leiter anliegen könnte.

### 1.1.3 Vorgehen

1. Mittels Strahlensatz die Maße des großen Dreiecks berechnen. Hierfür setzen wir das Verhältnis des mittleren oberen Dreiecks (auf der Kiste) mit dem großen Dreieck in Verhältnis. Ziel ist es dabei den Anlagepunkt der Leiter zu berechnen

Strahlensatz:

```
[4]: # define variables
```

```
x, y = var('x', 'y')
```

```
[5]: # create intercept theorem
```

```
x / chest == y / (y - chest)
```

```
x == y / (y - chest) * chest
```

```
[5]: 0.6250000000000000*x == y/(y - 1.6000000000000000)
```

```
[5]: x == 1.6000000000000000*y/(y - 1.6000000000000000)
```

Nun erhält man eine Gleichung mit zwei Unbekannten:

1.  $x \rightarrow$  Abstand von der Wand zum Standpunkt der Leiter (Schnittpunkt mit x-Achse)
2.  $y \rightarrow$  Anlagepunkt der Leiter an der Wand (Schnittpunkt mit y-Achse)

Um die Gleichung zu lösen, benötigen wir noch eine zweite Gleichung: Pythagoras des großen Dreiecks

```
[6]: # create pythagoras
```

```
ladder^2 == y^2 - x^2
```

```
y == sqrt(ladder^2 - x^2)
```

```
[6]: 42.25000000000000 == -x^2 + y^2
```

```
[6]: y == sqrt(-x^2 + 42.25000000000000)
```

Nun können wir die Gleichungen ineinander einsetzen und erhalten ein Ergebnis.

```
[7]: # combine both formulas and solve it
```

```
sol = find_root(sqrt(ladder^2 - (y / (y - chest) * chest)^2) == y, 4, 7)
```

```
y = sol
```

```
print('Anlagepunkt der Leiter: ' + str(y))
```

Anlagepunkt der Leiter: 6.12874976467387

Dies bedeutet, dass unser Anlagepunkt bei einer 6.50m langen Leiter bei  $\approx 6.1287m$  liegt.

### 1.1.4 Kontrollrechnung

Um das Ergebnis zu überprüfen berechnen wir uns des Weiteren noch den Standpunkt der Leiter, um dann die Leiterlänge auszurechnen, wenn diese bei 6.50m liegt, dann ist das Ergebnis richtig. Um den Standpunkt auszurechnen verwenden wir die Formel aus dem Strahlensatz von oben.

```
[8]: # use intercept theorem

x = y / (y - chest) * chest
print('Standpunkt der Leiter am Boden: ' + str(x))
```

Standpunkt der Leiter am Boden: 2.16527742379786

Der Standpunkt der Leiter liegt bei  $\approx 2.165m$ . Mit dieser Angabe können wir nun über den Pythagoras die Länge der Leiter ausrechnen. Um genau zu prüfen Berechnen wir auch die Abweichung von der gegebenen Leiterlänge.

```
[9]: # calculate ladder length and check the difference to the origin value

r, a = var('r', 'a')

r = sqrt(x^2 + y^2)
a = ladder - r

print('Abweichung: ' + str(a))
```

Abweichung: 9.94759830064140e-14

Die Abweichung von 9.94759830064140e-14 lässt sich durch das Runden der Maße erklären.

### 1.1.5 Fazit

Um nun die Erreichbarkeit der Decke zu prüfen. Müssen eine Reihe von Annahmen getroffen werden:

1. Größe der Person: Wir haben festgelegt, dass die Person durch ihre Körpergröße und das ausstrecken der Arme nach oben eine Höhe von 2.10m abdecken kann
2. Sicherheit: Ebenso haben wir festgelegt, dass die Person nicht auf der Spitze der Leiter steht, sondern 0.30m unter der Spitze.

Mittels dieser Annahmen können wir nun berechnen, ob die Person die Decke erreicht.

```
[10]: # check if the person reaches the top of the wall

h = var('h')

h = y - security_distance + person_height

print('Maximale Höhe: ' + str(h))
print('Wird die komplette Wandhöhe vollständig abgedeckt? ' + str(h > wall))
```

Maximale Höhe: 7.92874976467387

Wird die komplette Wandhöhe vollständig abgedeckt? True

Da die maximal erreichbare Höhe bei  $\approx 8m$  liegt, erreicht die Person die Decke!