

Der Besuch von Karels Freund ist nun schon wieder einige Zeit her. Sie haben viel unternommen – gut, dass Sie dabei keine Aerosole ausstoßen. 🧑‍🚒 Karel und Karli haben sich auch inzwischen wieder vertragen und so ist Karel die Idee zu einem neuen Spiel gekommen.

Ich laufe und
wenn ich auf einen Beeper stehe,
dann *hebe ich ihn auf.*
Ansonsten *lege ich einen Beeper ab.*

Aufgabe 1: Karel erweitert seine Handlungsanweisung, indem er je nach Bedingungsprüfung *das eine* oder *das andere macht*.

Auch wir Menschen agieren so.

- Formulieren Sie eine Bedingung und was Sie tun würden, wenn diese erfüllt ist und was Sie machen würden, wenn diese nicht erfüllt ist.
- Wiederholen Sie a) noch zwei Mal für andere Beispiele.

Beispiel: **Wenn** ich an einer roten Ampel stehe, **dann** *warte ich*. **Ansonsten** *gehe ich über die Straße*

Schauen Sie sich den Quellcode an, mit dem es Karel schafft, bei jeder Position erst zu prüfen, ob er auf einem Beeper steht und dann entsprechend die Anweisungen zu befolgen.

Vorher



Nachher



```
void beeperWechselDich()
{
    repeat(9)
    {
        if(onBeeper())
        {
            pickBeeper();
            moveForward();
        }
        else
        {
            dropBeeper();
            moveForward();
        }
    }
}
```

Aufgabe 2: Kennzeichnen Sie im oben dargestellten Quelltext:

- ...die Anweisungen für den Dann-Fall in grün.
- ...die Anweisungen für den Sonst-Fall in rot.

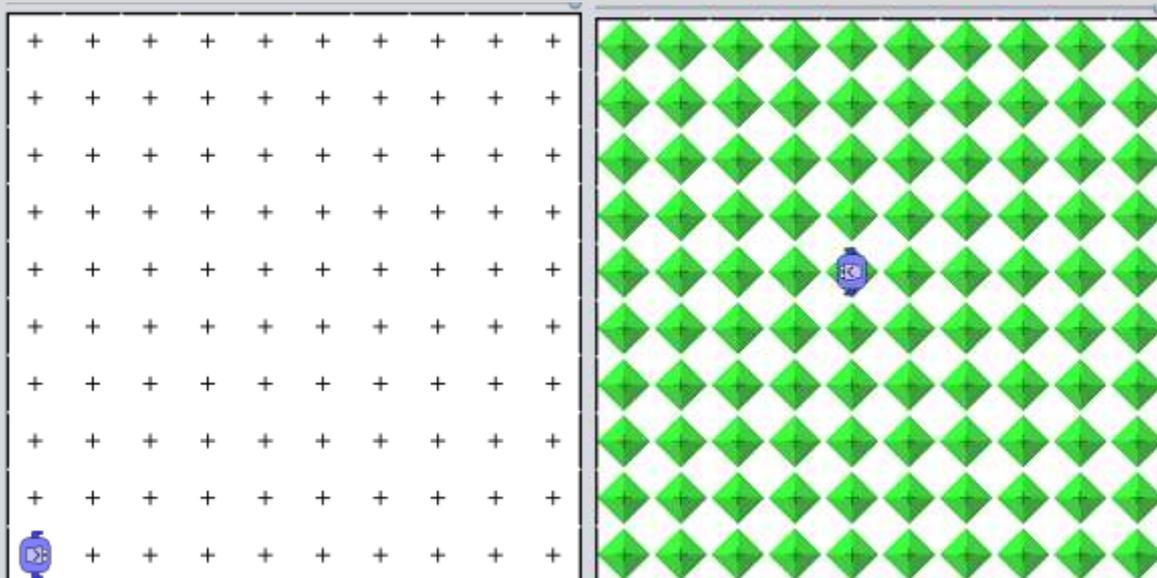
In Zukunft sagen wird dazu kurz: **zweiseitige if-Anweisung**.

Aufgabe 3:

- Angenommen, statt `repeat(9)` würde `repeat(10)` im dargestellten Quelltext stehen. Welche der Eigenschaften eines Algorithmus wäre dann nicht erfüllt? Nennen und erklären Sie diese Eigenschaft.

Aufgabe 4:

Karel hat bereits alle HomeOffice-Aufgaben erledigt, aber noch so viel Energie, dass er sich endlich an das Fliesen seines Bodens heranwagen will. Den ganzen Boden will er mit grünen Fliesen belegen. Helfen Sie ihm und bewegen Sie sich dazu in einer Schneckenlinie außen nach innen.



Lösungshinweis:

Karel kann auch darauf testen, ob zwei Bedingungen gleichzeitig gelten.

Die If-Anweisung beginnt dann so: **If (Bedingung1() && Bedingung2())**.

Natürlich müssen Sie statt Bedingung1() bzw. Bedingung2() die Methoden benutzen, die Karel kennt.

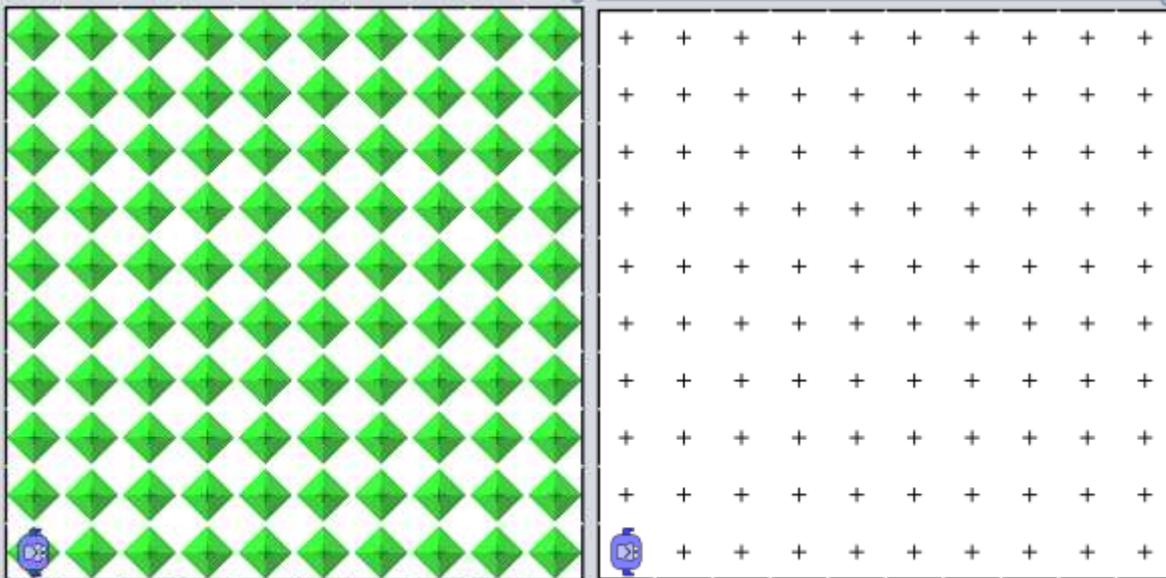
```
Void tileTheFloor()
{
```

Aufgabe 5:

So ein Mist! Karoline, Karels Frau, kam nach Hause und fand diese Überraschung mit den neuen Fliesen überhaupt nicht gut. Nun sammelt er alle grünen Fliesen wieder ein. Und Sie sind so hilfsbereit und helfen ihm wieder.

a) Nutzen Sie wieder die Schneckenlinie.

b) Bewegen Sie sich reihenweise vorwärts (der Code wird länger sein, sehen Sie es als Übung an).



a)

```
void removeTheTiles()    /*spiralförmig*/
{
```

b)

```
void removeTheTiles()    /*reihenweise*/
{
```