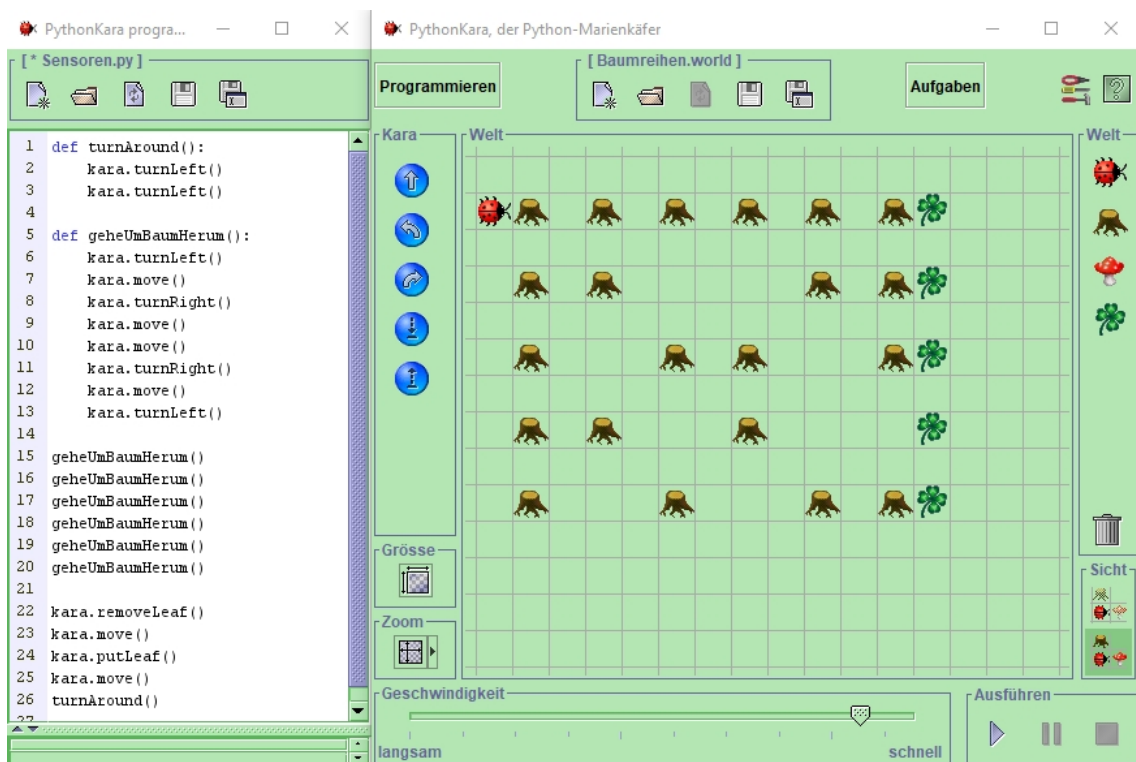


1. In der KARA-Welt sind fünf verschiedene Baumreihen mit jeweils einem Kleeblatt am Ende dargestellt. Das dargestellte Python-Programm arbeitet korrekt für alle fünf Baumreihen. Überzeuge Dich zuerst davon.



2. In den meisten Baumreihen macht KARA unnötige Bewegungen, wenn ein Baum fehlt. KARA besitzt aber Sensoren, die z.B. feststellen können, ob vor ihm, links neben ihm oder rechts neben ihm ein Baum steht. KARA soll nun einfach zwei Schritte geradeaus gehen, wenn vor ihm kein Baum steht. Den Sensor für die Abfrage, ob vorne ein Baum steht, fragt man mit `kara.treeFront()` ab. Damit kann man die Methode `geheUmBaumHerum()` umschreiben:

```

1 def geheUmBaumHerum():
2     if kara.treeFront():
3         kara.turnLeft()
4         kara.move()
5         kara.turnRight()
6         kara.move()
7         kara.move()
8         kara.turnRight()
9         kara.move()
10        kara.turnLeft()
11    else:
12        kara.move()
13        kara.move()

```

Schreibe Dein Programm entsprechend um und teste es in den verschiedenen Baumreihen.

### 3. Erläuterungen:

- (a) Die Syntax der einfachen Verzweigung mit `if` ist wie folgt:

```
if Bedingung :
    Auszufuehrende Methoden, wenn Bedingung wahr ist.
else :
    Auszufuehrende Methoden, wenn Bedingung falsch ist.
```

Die Bedingung ist in unserem Beispiel das Ergebnis der Sensorabfrage, die entweder *wahr* oder *falsch* ist. In Computersprachen benutzt man dafür die englischen Begriffe *true* und *false*.

Die Methoden, die jeweils ausgeführt werden sollen, stehen eingerückt hinter einem Doppelpunkt.

- (b) Der `else`-Zweig kann auch entfallen. Beispiel:

```
1 if kara.onLeaf():
2     kara.removeLeaf()
```

- (c) Eine verneinte Bedingung wird mit dem Schlüsselwort `not` formuliert:

```
1 if not kara.treeFront():
2     kara.move()
```

- (d) Mehrere Bedingungen können durch `or` bzw. `and` verknüpft werden.

```
1 if kara.treeFront() and kara.onLeaf():
2     kara.removeLeaf()
3     kara.turnLeft()
4     kara.turnLeft()
5
6 if kara.treeFront() or kara.treeLeft():
7     kara.turnRight()
```

- (e) Kompliziertere Bedingungen mit mehreren `and`, `or` und `not` kann man durch Klammerpaare `( )` schachteln.

- (f) Es gibt auch eine mehrfache Verzweigung:

```
1 if not kara.treeFront(): # Ist vorne frei?
2     kara.move()
3 elif not kara.treeLeft(): # Wenn nein, ist links frei?
4     kara.turnleft()
5     kara.move()
6 elif not kara.treeRight(): # Wenn nein, ist rechts frei?
7     kara.turnRight()
8     kara.move()
9 else: # Wenn ueberall nein, dann ...
10     kara.turnLeft()
11     kara.turnLeft()
12     kara.move()
```

Das Schlüsselwort `elif` ist eine Abkürzung für *else if*. Es können beliebig viele `elif`-Zweige genutzt werden. Der `else`-Zweig muss aber am Ende stehen.

4. Verändere dein Programm so, dass KARA auf seinem Weg zum Blatt bei einem fehlenden Baum ein Blatt ablegt (`kara.putLeaf()`).