

5.2 main Namespace Reference

Classes

- class **Plant**
Deze klasse bevat alle parameters van de plant: zijn positie, of hij is opgepakt of niet,...
- class **Robot**
Deze klasse bevat alle parameters van de robot: zijn positie, welke plant hij vast heeft, zijn geheugen,...

Functions

- def **draw_situation** (robot, Plant1, Plant2, Plant3, Plant4)
Tekent de huidige situatie op het scherm.
- def **loopRandomRond** ()
Door het oproepen van deze functie kan je de robot voor een bepaalde tijd.
- def **checkVoorPlanten** (ignorePlant)
De robot checkt of er plantenpotten in de buurt staan, en kiest vervolgens de dichtste plant.
- def **placePlant** ()
Dit laat de robot de plant die hij vast heeft neerzetten.
- def **pickUpPlant** (plant)
Dit laat de robot de meegegeven plant oppakken.
- def **movePlantToSafePlace** ()
Dit laat de robot de plant die hij vast heeft naar een veilige plaats brengen om neer te zetten.
- def **checkPlantenOpstelling** ()
Deze functie zal nagaan of de huidige opstelling van de planten in orde is.
- def **rijdRobotNaarPlant** (plant)
Deze functie laat de robot naar een bepaalde plant toe rijden.
- def **rijCirkelEnMap** (straal)
Deze functie laat de robot een cirkel rijden.
- def **draai90Graden** ()
Deze functie laat de robot 90 graden draaien.
- def **rijRechtdoor** (meter)
Deze functie laat de robot rechtdoor rijden voor een aantal meter.
- def **beeindigSpel** ()
Deze functie wordt enkel opgeroepen als de opstelling in orde is.

Variables

- int **height** = 950
- int **width** = 850
- **win** = pygame.display.set_mode((**height** , **width**))
- **font** = pygame.font.SysFont('arial', 30)
- **label** = font.render('Druk op enter om de simulatie te starten', 1, (0, 0, 0))
- **robotImg** = pygame.image.load('HA.png')
- list **operationList** = [-1, 0, 1]
- **df** = pd.read_excel("Scenario3.xlsx")
- **df_nr** = **df**["Nummer"]
- **df_x** = **df**["X"]
- **df_y** = **df**["Y"]
- **Plant1** = **Plant**(**df_x**[0], **df_y**[0], **df_nr**[0])

- **Plant2** = **Plant**(**df_x**[1], **df_y**[1], **df_nr**[1])
- **Plant3** = **Plant**(**df_x**[2], **df_y**[2], **df_nr**[2])
- **Plant4** = **Plant**(**df_x**[3], **df_y**[3], **df_nr**[3])
- list **bloempottenLijst** = [**Plant1**, **Plant2**, **Plant3**, **Plant4**]
- **bloempotlmg** = pygame.image.load('bloempot.png')
- int **x** = **height***0.5
- int **y** = **width***0.5
- **robot** = **Robot**(**x**, **y**)
- int **snelheid** = 10
- **ignorePlant** = None
- bool **start** = False
- bool **done** = False
- **gevondenPlant** = robot.gevondenPlanten.pop()
- **gevondenPlanten**

5.2.1 Detailed Description

Dit package bevat de lege while loop waar de leerlingen de code in dienen te schrijven. Verder bevat dit package ook documentatie voor alle functies.

5.2.2 Function Documentation

5.2.2.1 beeindigSpel()

```
def main.beeindigSpel ( )
```

Deze functie wordt enkel opgeroepen als de opstelling in orde is.

Ze zorgt voor een correcte afsluiting van de simulatie.

5.2.2.2 checkPlantenOpstelling()

```
def main.checkPlantenOpstelling ( )
```

Deze functie zal nagaan of de huidige opstelling van de planten in orde is.

Dit door te kijken hoever alle planten van elkaar staan. Als ze allemaal ver genoeg van elkaar staan zal de functie TRUE terugsturen. Anders stuurt het FALSE terug.

Parameters

out	<i>goed</i>	Goed is TRUE als de opstelling in orde is, en FALSE als dit niet het geval is.
-----	-------------	--

5.2.2.3 checkVoorPlanten()

```
def main.checkVoorPlanten (
    ignorePlant )
```

De robot checkt of er plantenpotten in de buurt staan, en kiest vervolgens de dichtste plant.

Parameters

in	<i>ignorePlant</i>	Dit is het NUMMER van de plant die genegeerd zal worden. Dit is wellicht de plant die de robot net verzet heeft. Door deze nu te negeren zal de robot niet 2 keer op rij dezelfde plant detecteren.
out	<i>goToPlant</i>	De robot geeft de dichtstbijzijnde plant terug als argument.

5.2.2.4 draai90Graden()

```
def main.draai90Graden ( )
```

Deze functie laat de robot 90 graden draaien.

5.2.2.5 draw_situation()

```
def main.draw_situation (
    robot,
    Plant1,
    Plant2,
    Plant3,
    Plant4 )
```

Tekent de huidige situatie op het scherm.

Parameters

in	<i>robot</i>	De robot
in	<i>Plant1</i>	De eerste plantenpot
in	<i>Plant2</i>	De tweede plantenpot
in	<i>Plant3</i>	De derde plantenpot
in	<i>Plant4</i>	De vierde plantenpot

5.2.2.6 loopRandomRond()

```
def main.loopRandomRond ( )
```

Door het oproepen van deze functie kan je de robot voor een bepaalde tijd.

De duur dat de robot zal rondlopen hangt af van de parameter 'snelheid'

5.2.2.7 movePlantToSafePlace()

```
def main.movePlantToSafePlace ( )
```

Dit laat de robot de plant die hij vast heeft naar een veilige plaats brengen om neer te zetten.

De robot blijft hiervoor willekeurig rond lopen tot de kust veilig is.

5.2.2.8 pickUpPlant()

```
def main.pickUpPlant (
    plant )
```

Dit laat de robot de meegegeven plant oppakken.

Parameters

in	<i>plant</i>	De plant die opgepakt moet worden.
----	--------------	------------------------------------

5.2.2.9 placePlant()

```
def main.placePlant ( )
```

Dit laat de robot de plant die hij vast heeft neerzetten.

5.2.2.10 rijCirkelEnMap()

```
def main.rijCirkelEnMap (
    straal )
```

Deze functie laat de robot een cirkel rijden.

Tijdens het rijden van deze cirkel houdt de robot bij welke planten het allemaal gedecteerd heeft. De robot houdt deze planten bij in zijn lijst 'gevondenplanten'.

Parameters

in	<i>straal</i>	De straal van de cirkel die de robot moet rijden.
----	---------------	---

5.2.2.11 rijdRobotNaarPlant()

```
def main.rijdRobotNaarPlant (
    plant )
```

Deze functie laat de robot naar een bepaalde plant toe rijden.

Parameters

<i>in</i>	<i>plant</i>	De plant waar de robot naartoe moet rijden.
-----------	--------------	---

5.2.2.12 rijRechtdoor()

```
def main.rijRechtdoor (
    meter )
```

Deze functie laat de robot rechtdoor rijden voor een aantal meter.

Het aantal meter dat deze robot zal doorrijden hangt af van wat meegegeven is. Welke richting de robot precies vooruit rijdt hangt af van de richting waarin de robot is georiënteerd.

Parameters

<i>in</i>	<i>meter</i>	Het aantal meter dat de robot vooruit dient te rijden.
-----------	--------------	--

5.2.3 Variable Documentation

5.2.3.1 bloempotImg

```
main.bloempotImg = pygame.image.load('bloempot.png')
```

5.2.3.2 bloempottenLijst

```
list main.bloempottenLijst = [ Plant1, Plant2, Plant3, Plant4]
```

5.2.3.3 df

```
main.df = pd.read_excel("Scenario3.xlsx")
```