Day - 6

**Reeborgs World**

Programming game

**6.4 Reeborg's World**

Reeborg's World is a free "Karel the robot" type of environment used to teach programming, using either Python or Javascript.

* **Exercise 6.1 Hurdles with constant-height**

def turn\_right():

turn\_left()

turn\_left()

turn\_left()

while not at\_goal():

move()

turn\_left()

move()

turn\_right()

move()

turn\_right()

move()

turn\_left()

* **Exercise 6.2 Hurdles with constant-height and variable-distance**

def turn\_right():

turn\_left()

turn\_left()

turn\_left()

def jump():

turn\_left()

move()

turn\_right()

move()

turn\_right()

move()

turn\_left()

while not at\_goal():

if wall\_in\_front():

jump()

else:

move()

* **Exercise 6.3 Hurdles with height**

#<mycode>

def turn\_right():

turn\_left()

turn\_left()

turn\_left()

def jump():

turn\_right()

move()

turn\_right()

move()

while not at\_goal():

while wall\_on\_right() and wall\_in\_front:

if wall\_in\_front():

turn\_left()

elif wall\_on\_right:

if at\_goal():

done()

else:

move()

jump()

#<angela's code>

|  |  |
| --- | --- |
| def turn\_right():  turn\_left()  turn\_left()  turn\_left()  def jump():  turn\_left()  while wall\_on\_right():  move()  turn\_right()  move()  turn\_right()  while front\_is\_clear():  move()  turn\_left()    while not at\_goal():  if wall\_in\_front():  jump()  else:  move() | def turn\_right():  turn\_left()  turn\_left()  turn\_left()  def up():  while wall\_on\_right():  move()  def down():  while front\_is\_clear():  move()    def jump():  turn\_left()  up()  turn\_right()  move()  turn\_right()  down()  turn\_left()  while not at\_goal():  if wall\_in\_front():  jump()  else:  move() |

* **Final Project : Escaping the Maze**

Algorithm: following the right edge of the wall.

def turn\_right():

turn\_left()

turn\_left()

turn\_left()

#<mycode>

while not at\_goal():

if wall\_on\_right():

if wall\_in\_front():

turn\_left()

elif front\_is\_clear():

move()

else:

turn\_right()

move()

#<angela's code>

def turn\_right():

turn\_left()

turn\_left()

turn\_left()

while not at\_goal():

if right\_is\_clear():

turn\_right()

move()

elif front\_is\_clear():

move()

else:

turn\_left()

Special case: if frint and right is clear. Middle of space.

def turn\_right():

turn\_left()

turn\_left()

turn\_left()

while front\_is\_clear():

move()

turn\_left()

while not at\_goal():

if wall\_on\_right():

if wall\_in\_front():

turn\_left()

else:

move()

else:

turn\_right()

move()