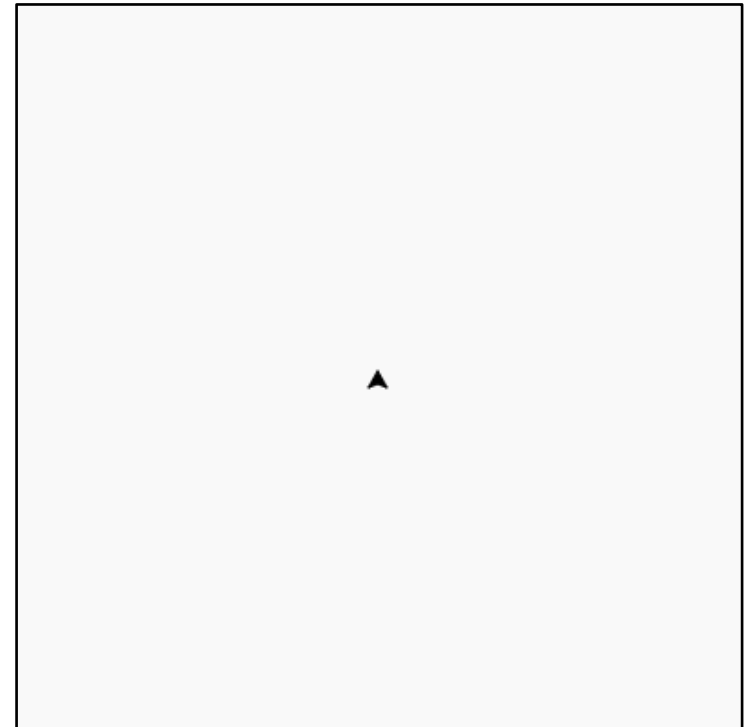


Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

```
trinket Run ? Modules  
main.py  
32  
33 trunk = 40  
34 level_limit = 2  
35  
36 # Recursive function  
37 def fwdtrn(d,l):  
38     if l <= level_limit:  
39         t.fd(d)  
40         t.lt(20)  
41         fwdtrn(d*0.9, l+1)  
42         t.rt(40)  
43         fwdtrn(d*0.9, l+1)  
44         t.lt(20)  
45         t.bk(d)  
46  
47 fwdtrn(trunk,1)
```

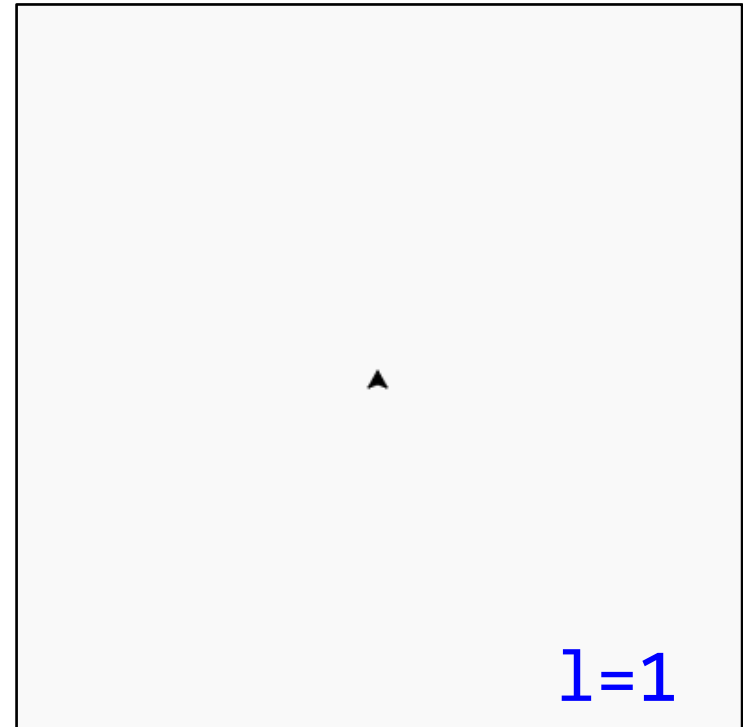


Call function with $l=1$

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 1 function
↓

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```

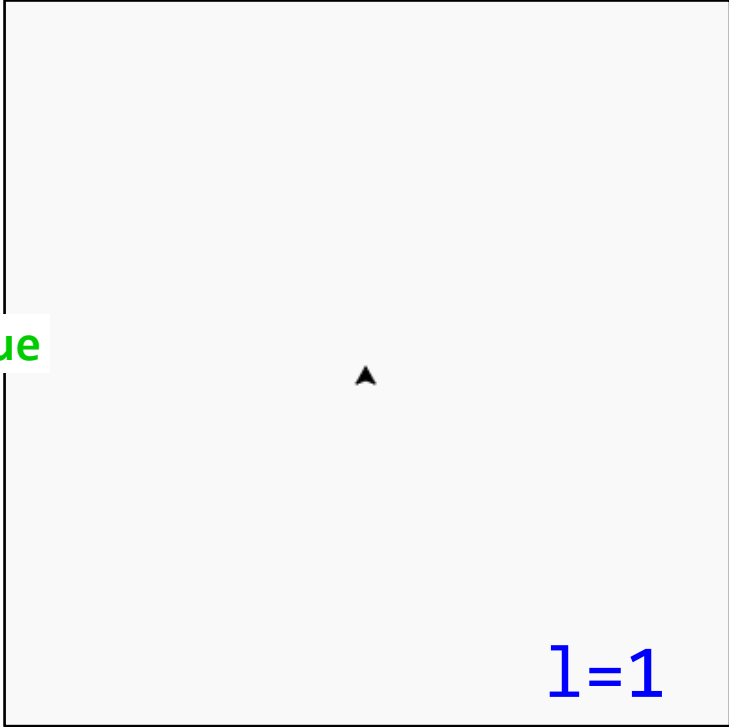


Call function with l=1

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 1 function
↓


```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit: True
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



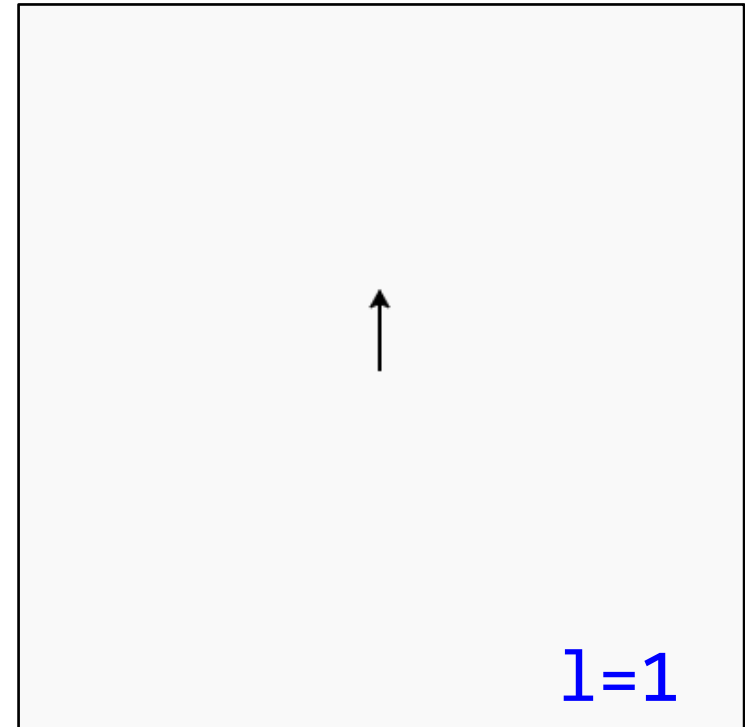
l=1

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 1 function




```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```

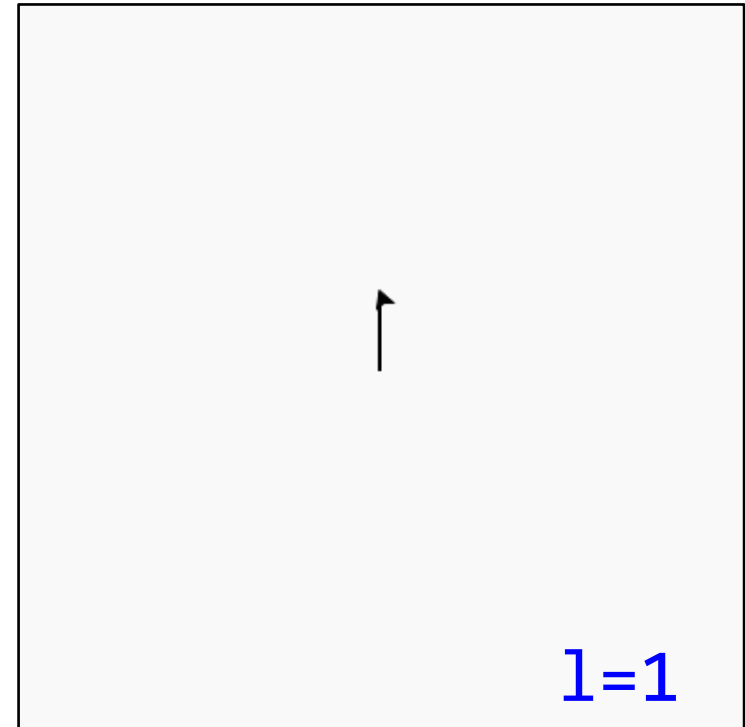


Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 1 function




```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```

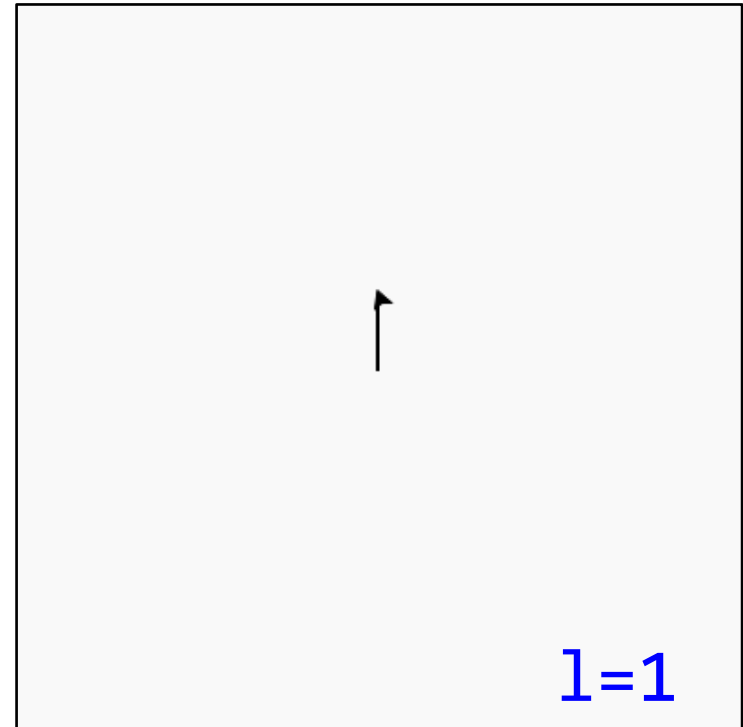


Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 1 function



```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```

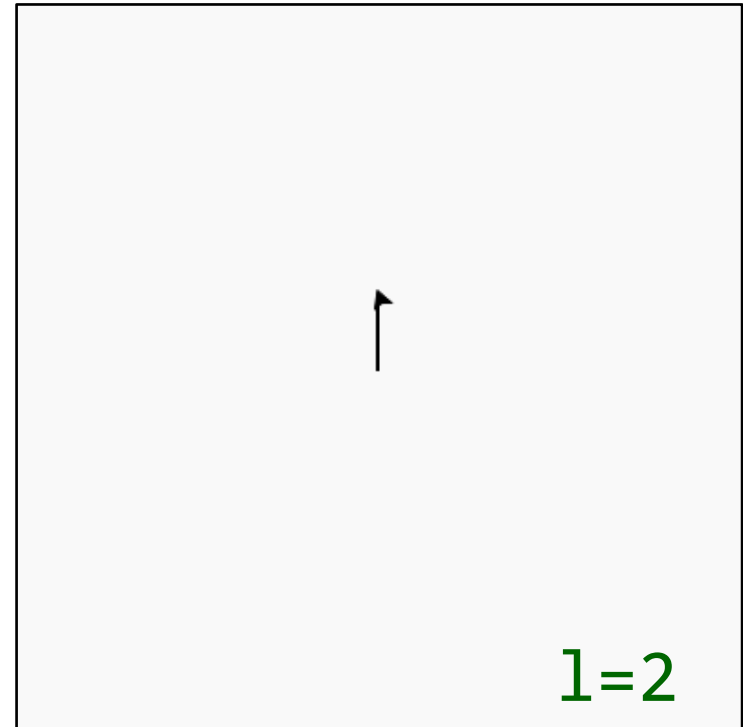


Call function with l=2

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

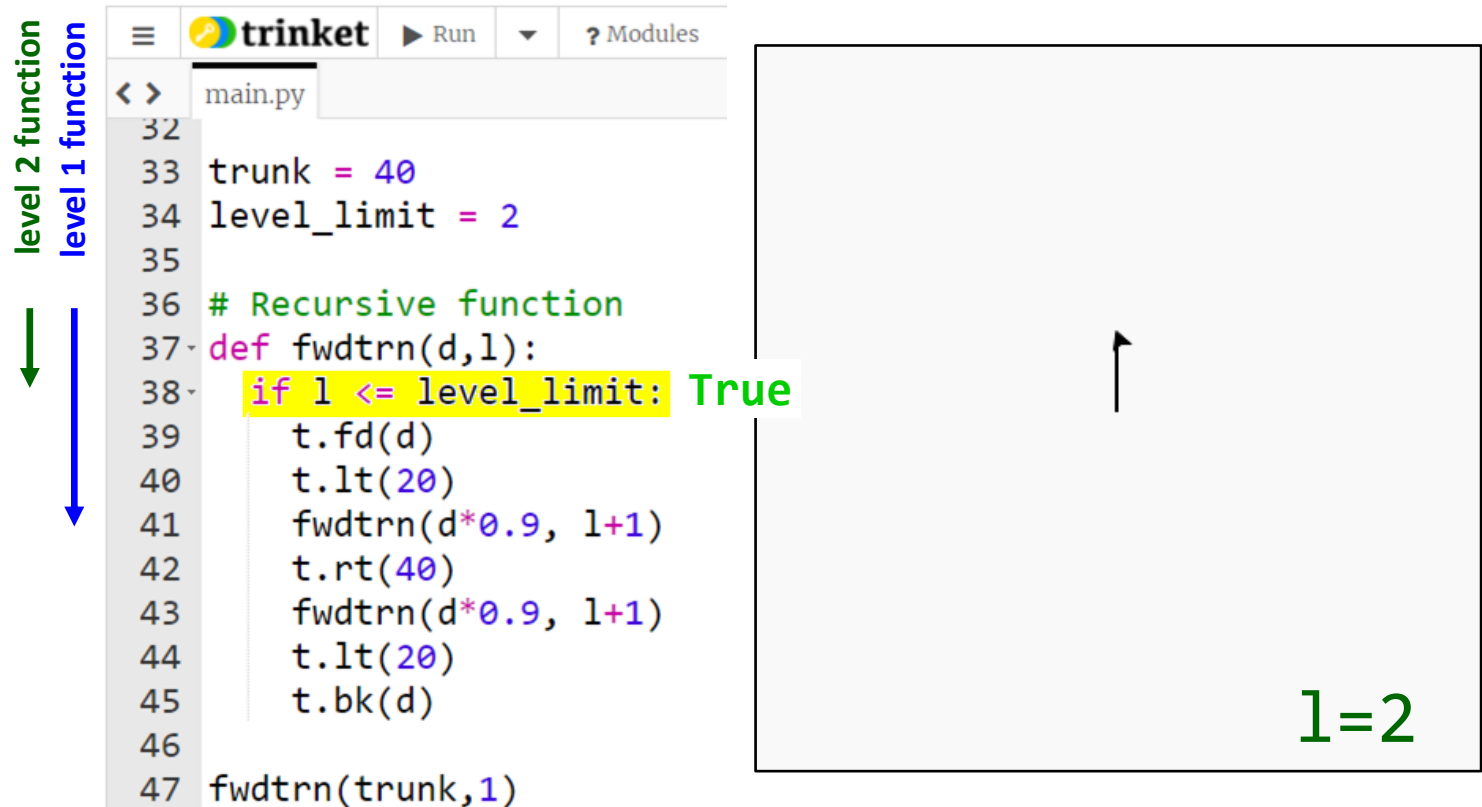
level 2 function
level 1 function

```
32  
33 trunk = 40  
34 level_limit = 2  
35  
36 # Recursive function  
37 def fwdtrn(d,l):  
38     if l <= level_limit:  
39         t.fd(d)  
40         t.lt(20)  
41         fwdtrn(d*0.9, l+1)  
42         t.rt(40)  
43         fwdtrn(d*0.9, l+1)  
44         t.lt(20)  
45         t.bk(d)  
46  
47 fwdtrn(trunk,1)
```



Call function with l=2

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



The image shows a screenshot of the Trinket code editor with a Python script for drawing a fractal tree. The code is as follows:

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit: True
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```

Annotations on the left side of the code:

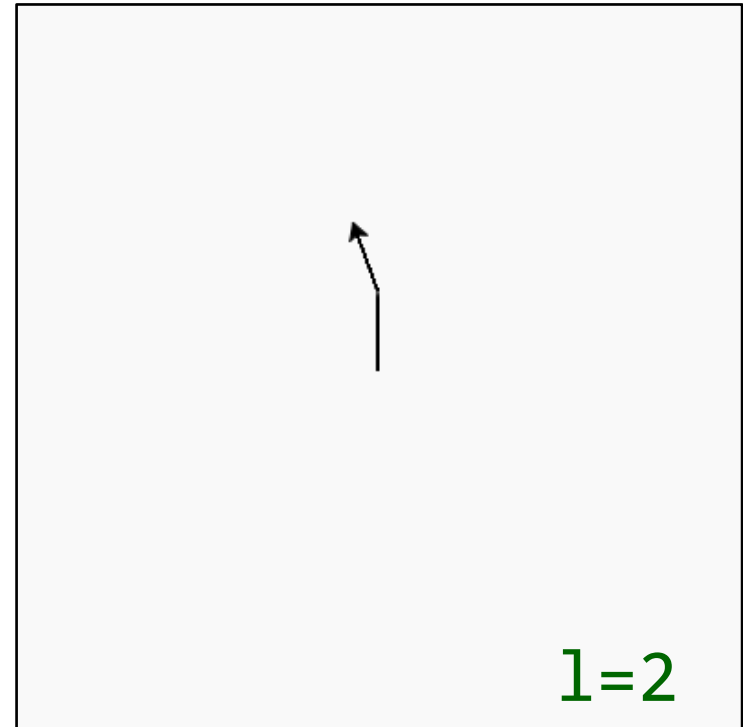
- A green arrow points down from line 36 to line 38, labeled "level 2 function".
- A blue arrow points down from line 38 to line 45, labeled "level 1 function".

On the right side, there is a large empty square box representing the canvas. Inside the box, there is a small black arrow pointing upwards and the text "l=2" in green.

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 2 function
level 1 function

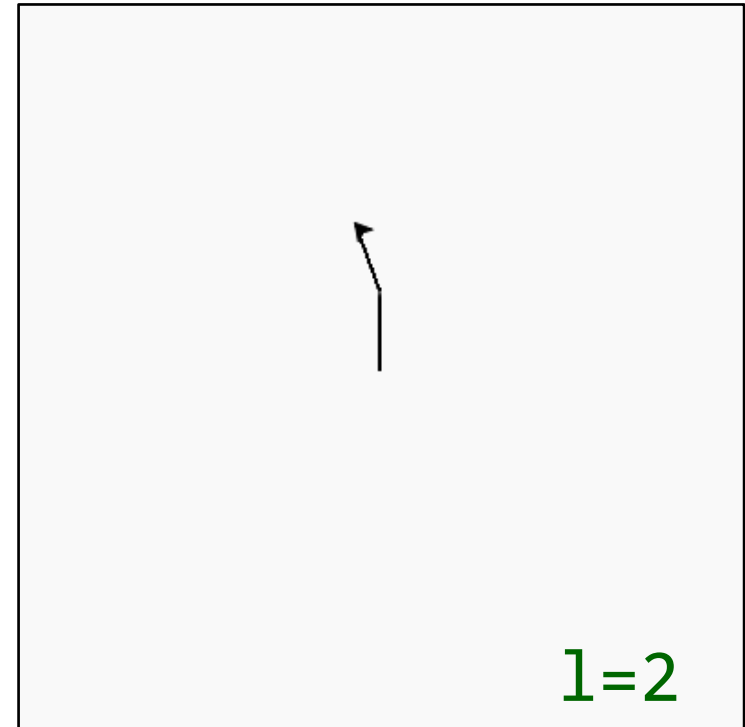
```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 2 function
level 1 function

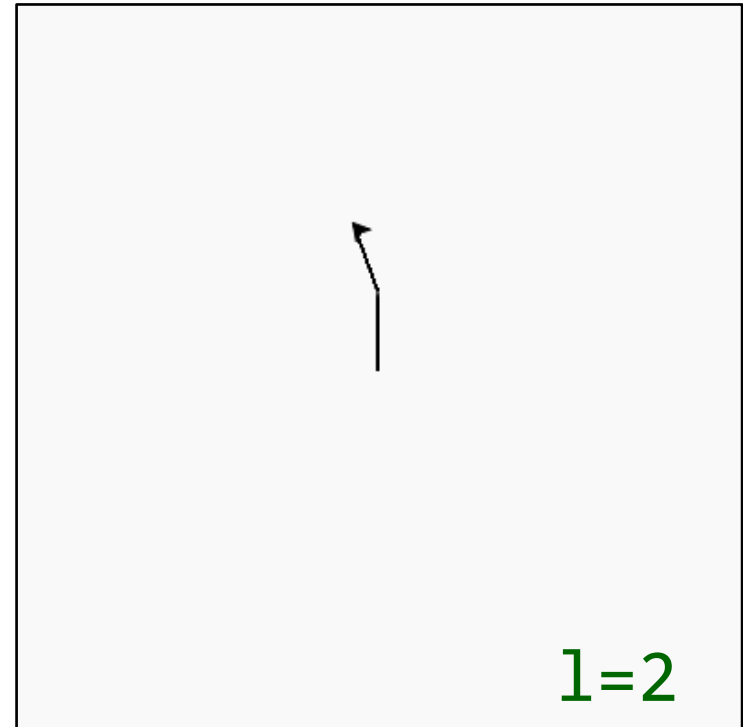
```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 2 function
level 1 function

```
32  
33 trunk = 40  
34 level_limit = 2  
35  
36 # Recursive function  
37 def fwdtrn(d,l):  
38     if l <= level_limit:  
39         t.fd(d)  
40         t.lt(20)  
41         fwdtrn(d*0.9, l+1)  
42         t.rt(40)  
43         fwdtrn(d*0.9, l+1)  
44         t.lt(20)  
45         t.bk(d)  
46  
47 fwdtrn(trunk,1)
```

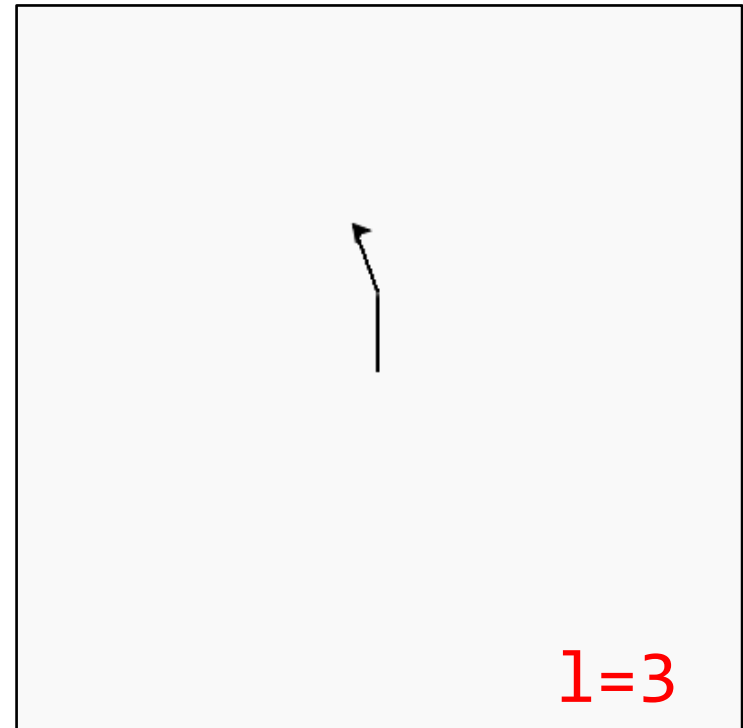


Call function with l=3

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function
level 2 function
level 1 function

```
32  
33 trunk = 40  
34 level_limit = 2  
35  
36 # Recursive function  
37 def fwdtrn(d,l):  
38     if l <= level_limit:  
39         t.fd(d)  
40         t.lt(20)  
41         fwdtrn(d*0.9, l+1)  
42         t.rt(40)  
43         fwdtrn(d*0.9, l+1)  
44         t.lt(20)  
45         t.bk(d)  
46  
47 fwdtrn(trunk,1)
```



Call function with l=3

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 2 function

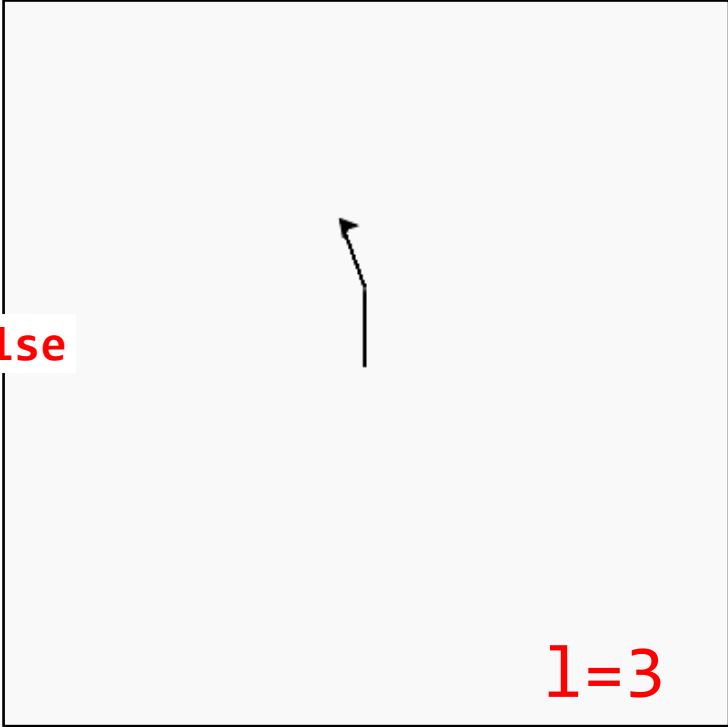
level 1 function

↓

↓

↓

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit: False
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



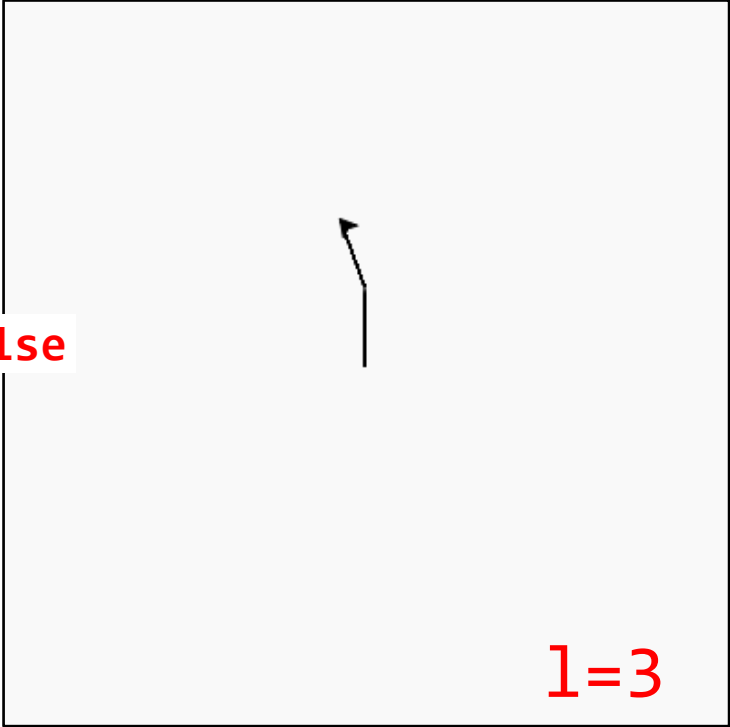
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 2 function

level 1 function

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit: False
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



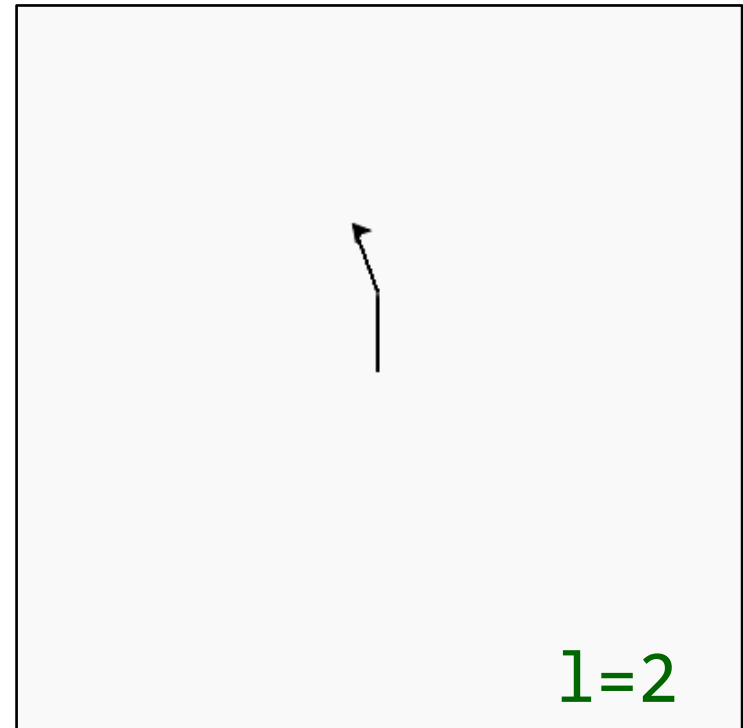
l=3

Complete level 3 fn

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function
level 2 function
level 1 function

```
32  
33 trunk = 40  
34 level_limit = 2  
35  
36 # Recursive function  
37 def fwdtrn(d,l):  
38     if l <= level_limit:  
39         t.fd(d)  
40         t.lt(20)  
41         fwdtrn(d*0.9, l+1)  
42         t.rt(40)  
43         fwdtrn(d*0.9, l+1)  
44         t.lt(20)  
45         t.bk(d)  
46  
47 fwdtrn(trunk,1)
```

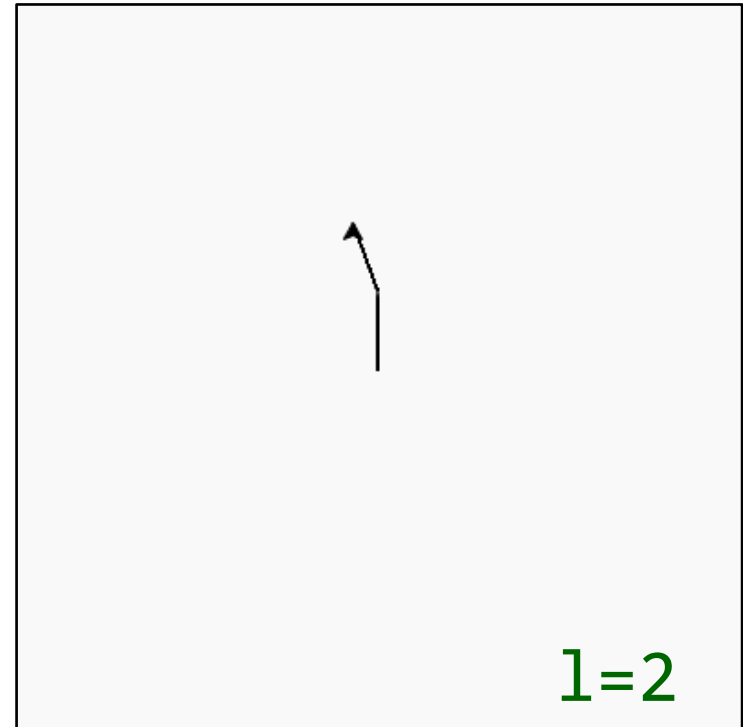


Continue unfinished level 2 fn

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function
level 2 function
level 1 function

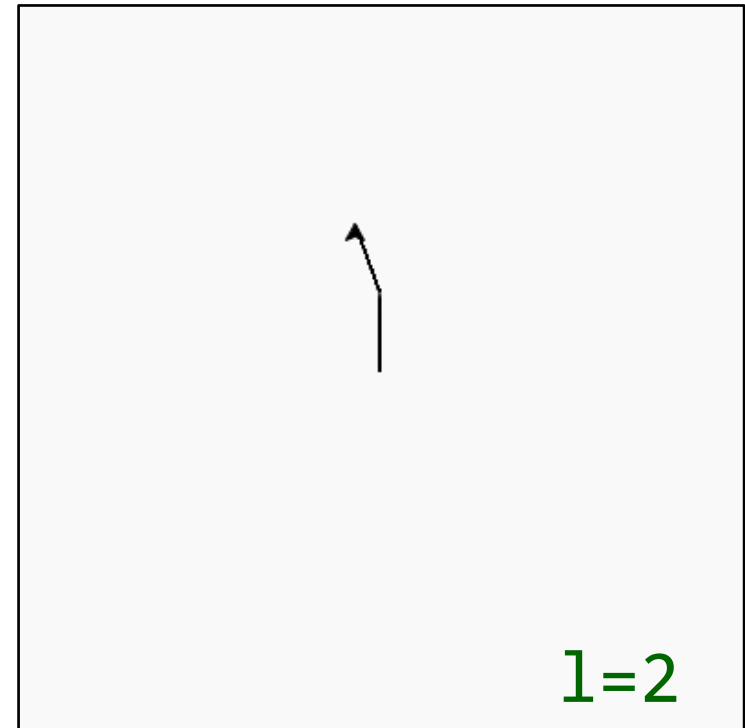
```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

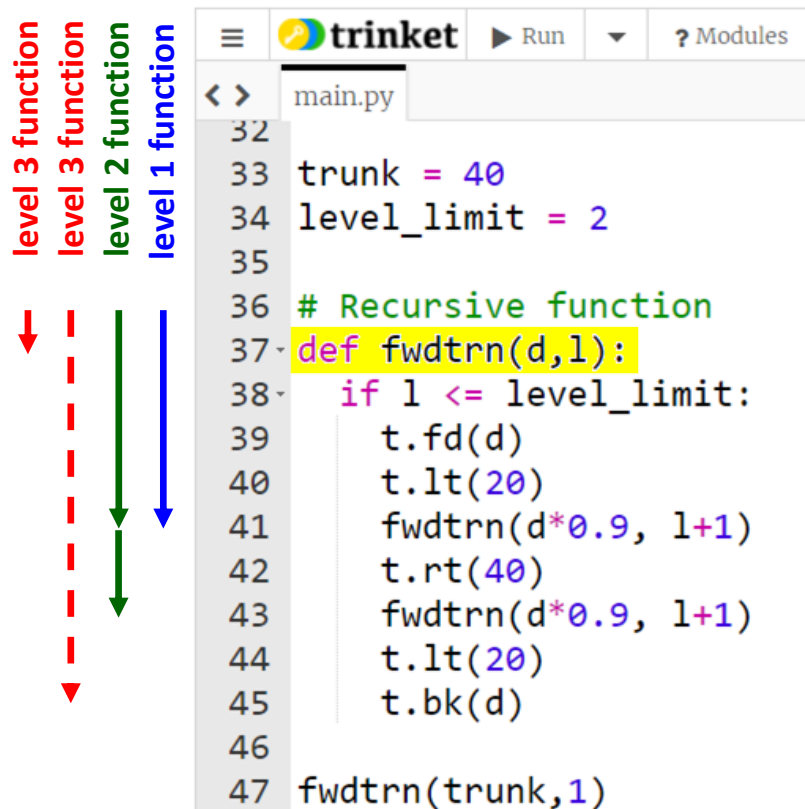
level 3 function
level 2 function
level 1 function

```
32  
33 trunk = 40  
34 level_limit = 2  
35  
36 # Recursive function  
37 def fwdtrn(d,l):  
38     if l <= level_limit:  
39         t.fd(d)  
40         t.lt(20)  
41         fwdtrn(d*0.9, l+1)  
42         t.rt(40)  
43         fwdtrn(d*0.9, l+1)  
44         t.lt(20)  
45         t.bk(d)  
46  
47 fwdtrn(trunk,1)
```

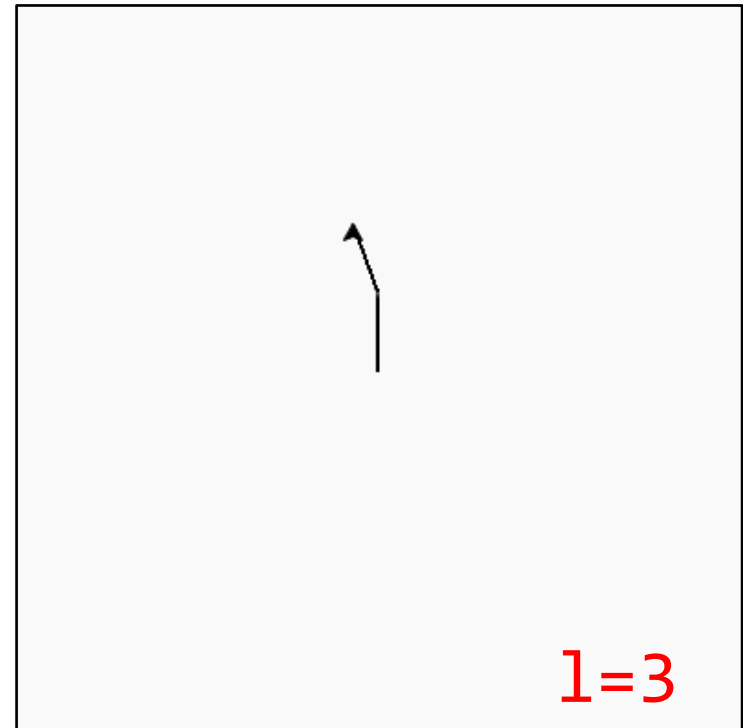


Call function with l=3

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Call function with **l=3**

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 3 function

level 2 function

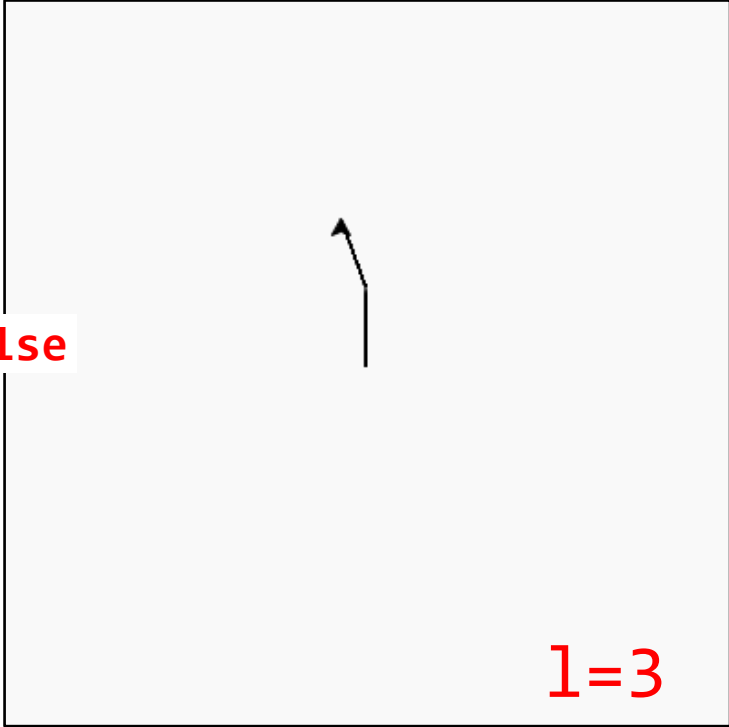
level 1 function

↓

↓

↓

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit: False
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 3 function

level 2 function

level 1 function

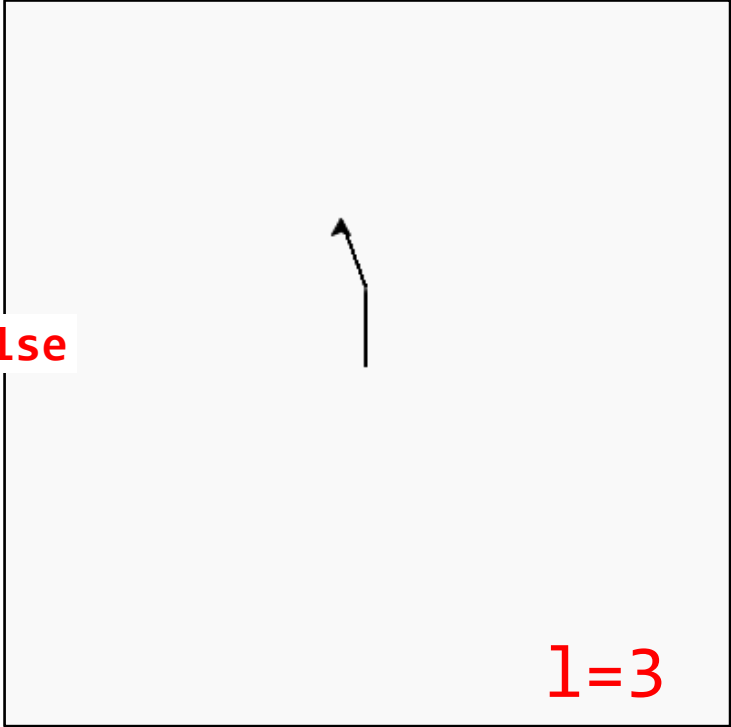
▼

▼

▼

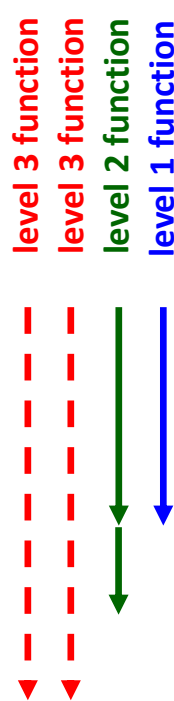
▼

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit: False
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```

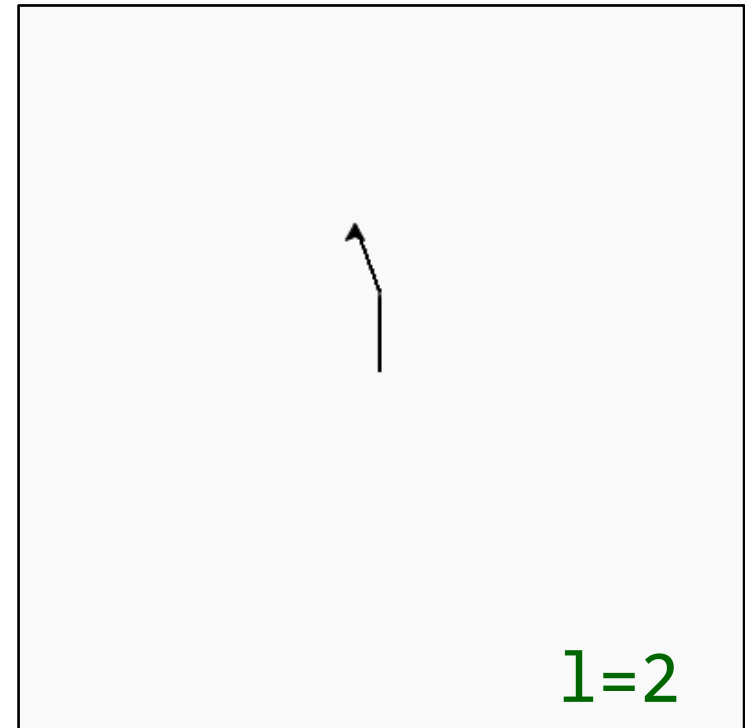


Complete level 3 fn

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

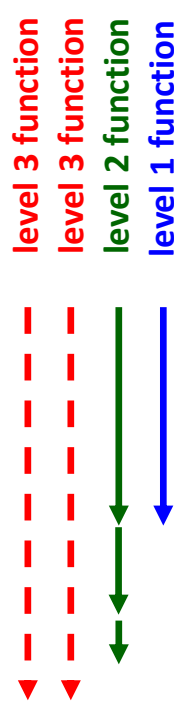


```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```

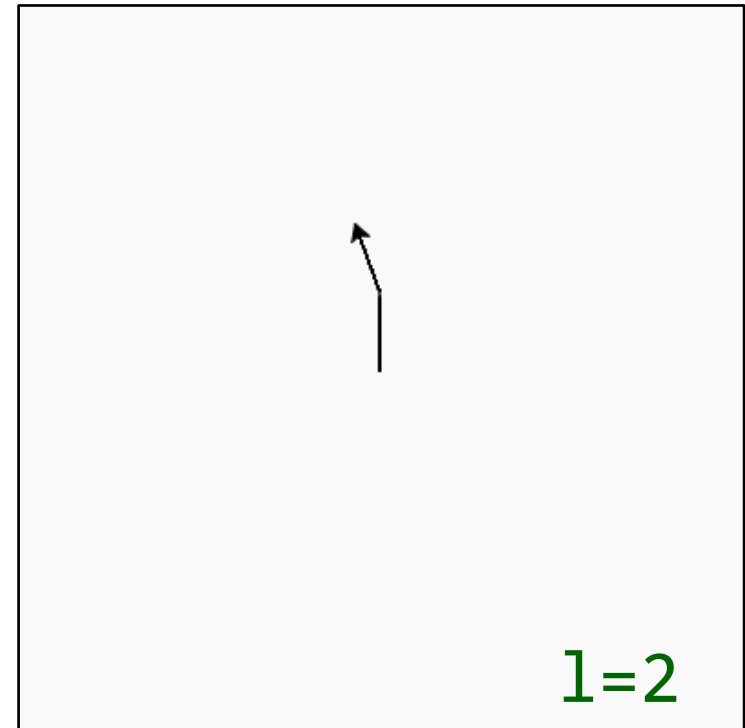


Continue unfinished level 2 fn

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Continue unfinished level 2 fn

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 3 function

level 2 function

level 1 function

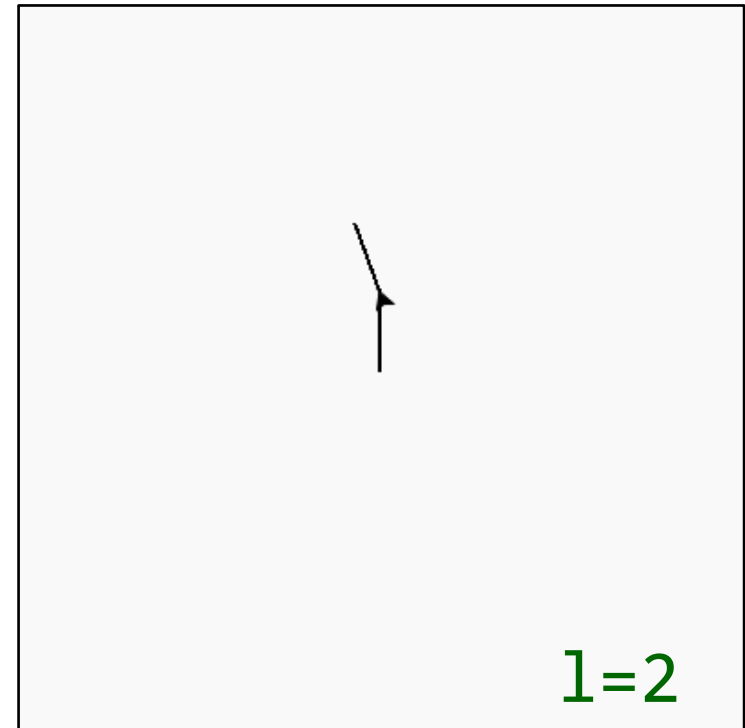
↓

↓

↓

↓

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Complete level 2 fn

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 3 function

level 2 function

level 1 function

--

--

--

--

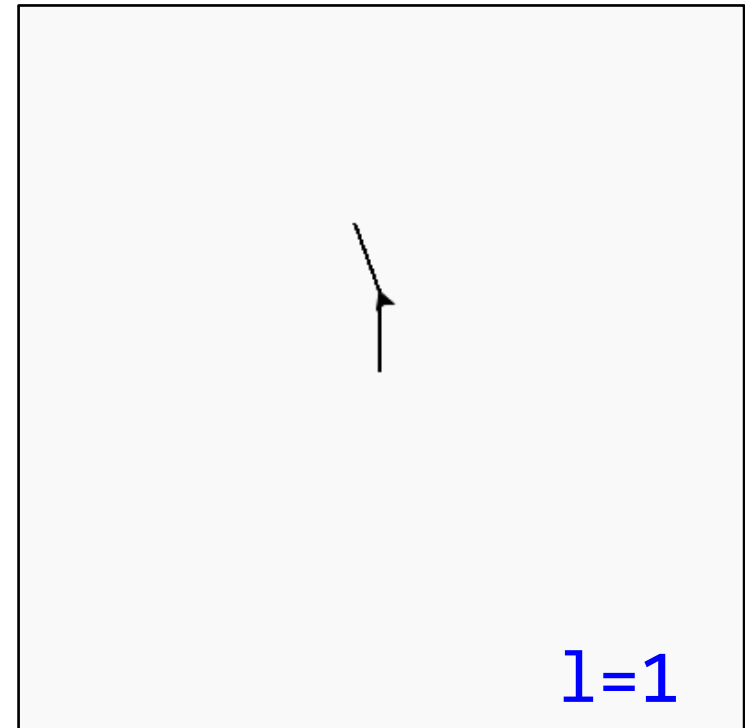
↓

↓

↓

↓

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Continue unfinished level 1 fn

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 3 function

level 2 function

level 1 function

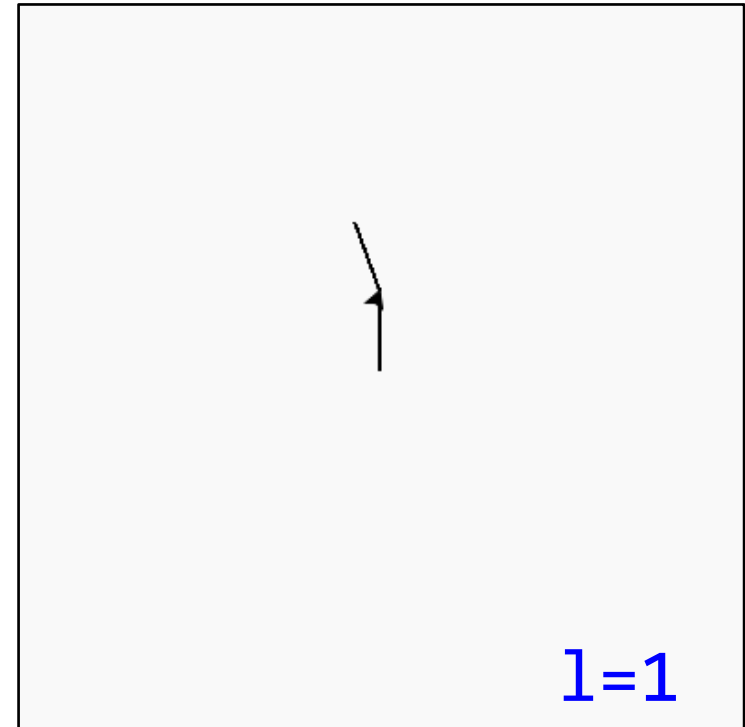
↓

↓

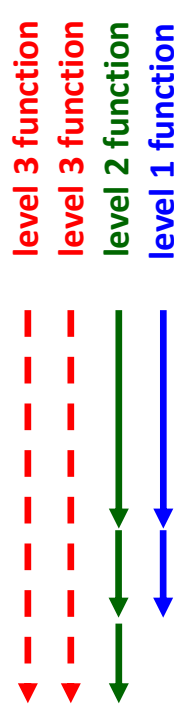
↓

↓

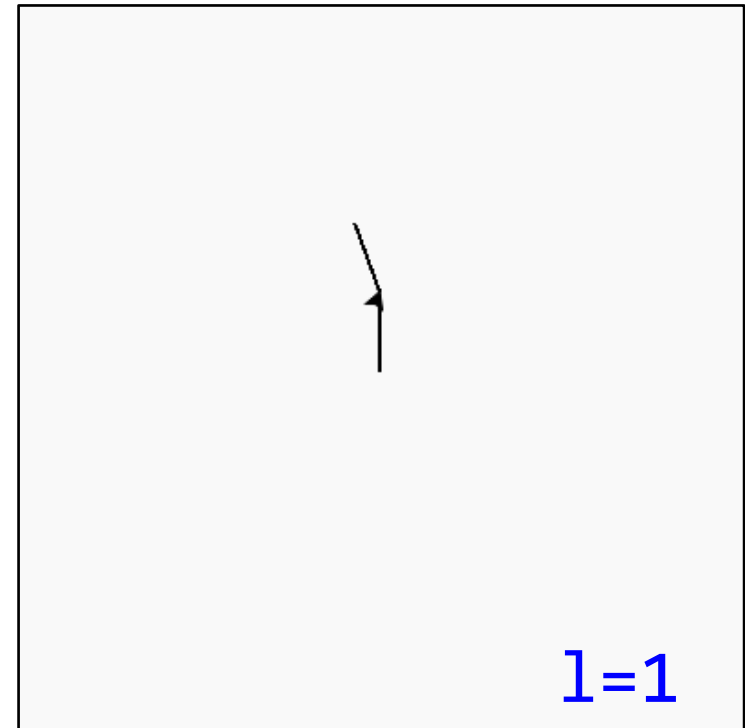
```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Call function with **l=2**

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 2 function

level 3 function

level 3 function

level 2 function

level 1 function

↓

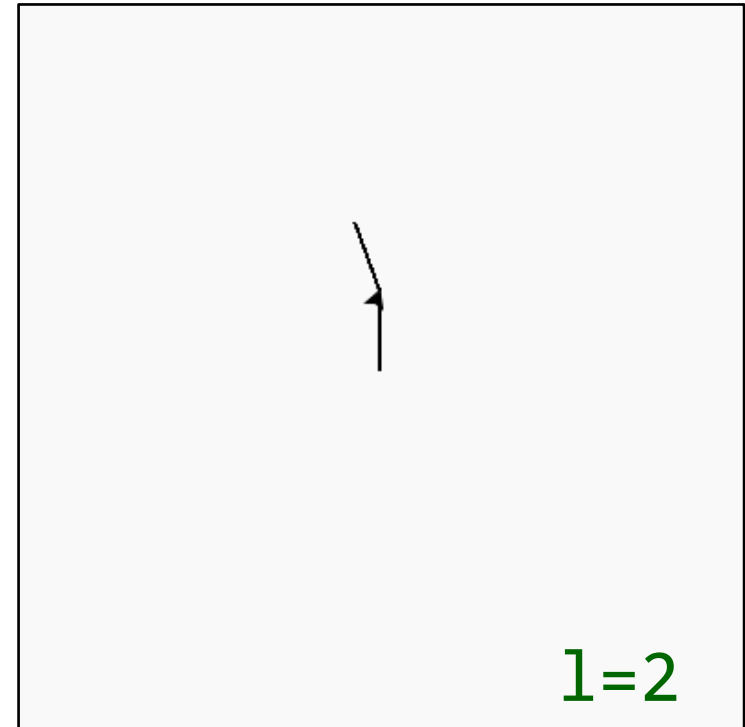
↓

↓

↓

↓

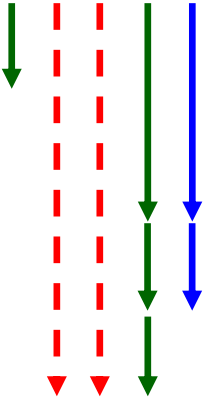
```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



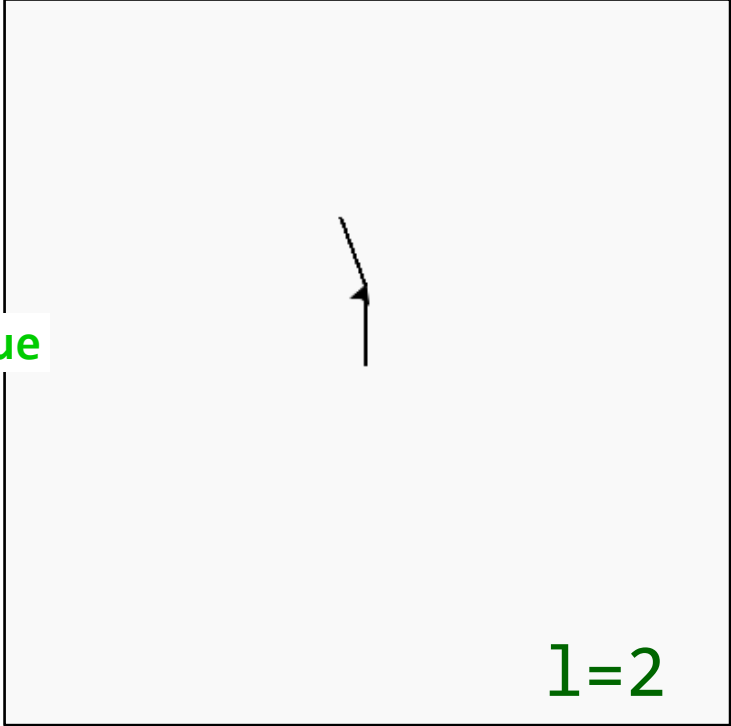
Call function with $l=2$

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 2 function
level 3 function
level 3 function
level 2 function
level 1 function



```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit: True
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



l=2

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

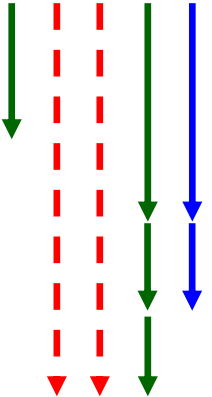
level 2 function

level 3 function

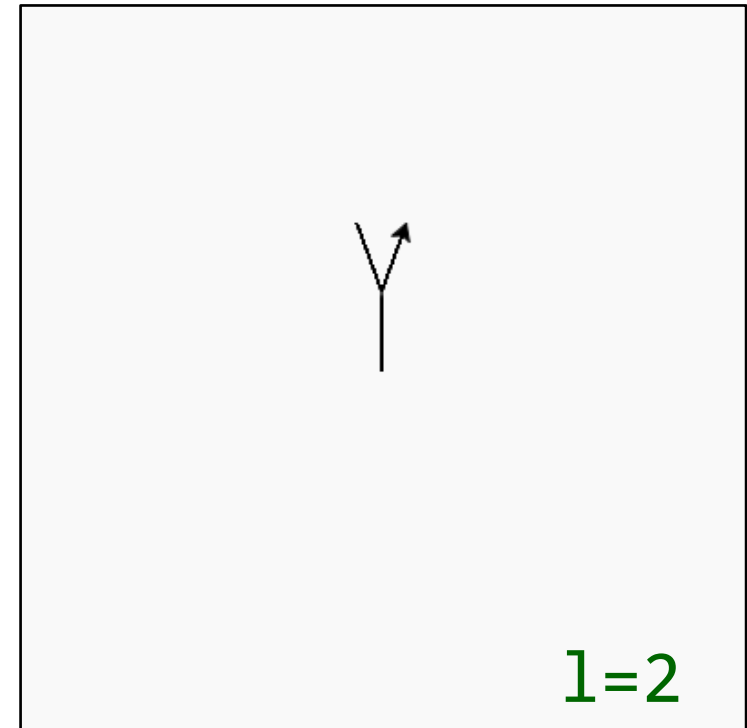
level 3 function

level 2 function

level 1 function

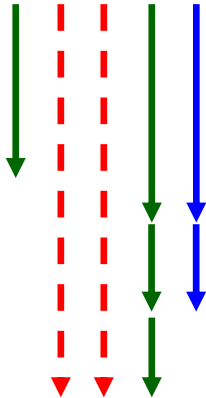


```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```

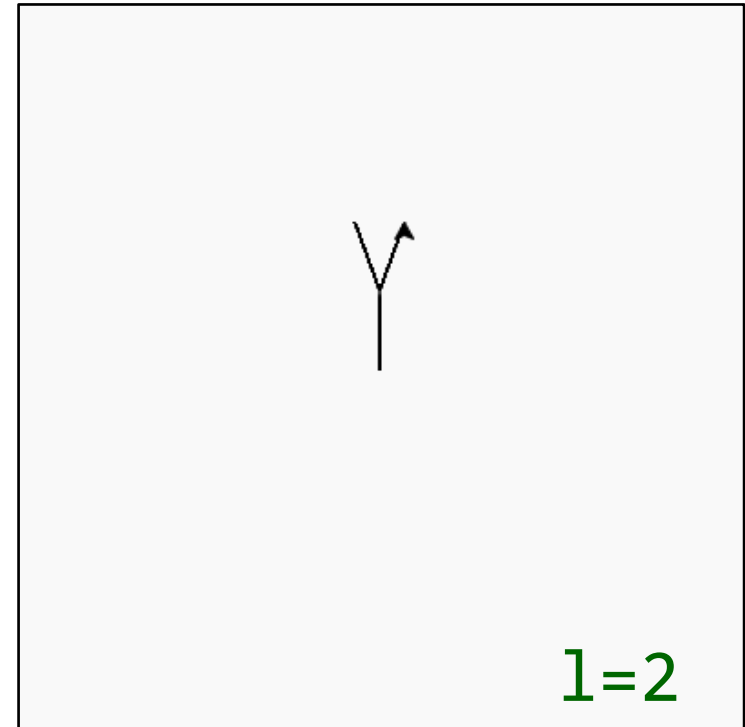


Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 2 function
level 3 function
level 3 function
level 2 function
level 1 function



```
trinket Run ? Modules
main.py
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

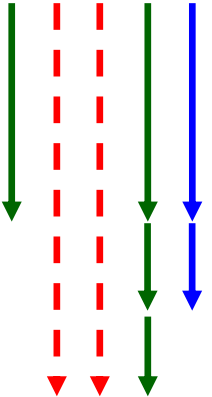
level 2 function

level 3 function

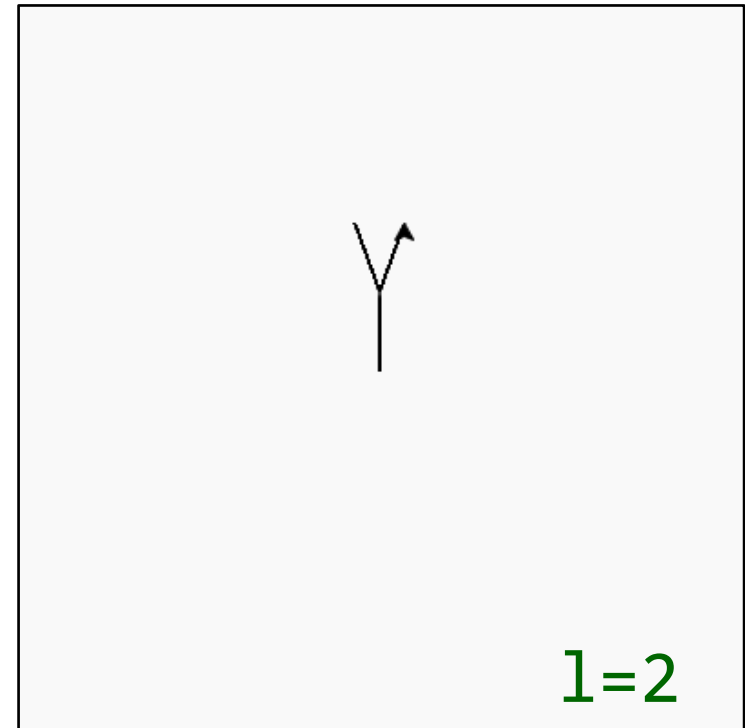
level 3 function

level 2 function

level 1 function



```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Call function with $l=3$

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 2 function

level 3 function

level 3 function

level 2 function

level 1 function

↓

↓

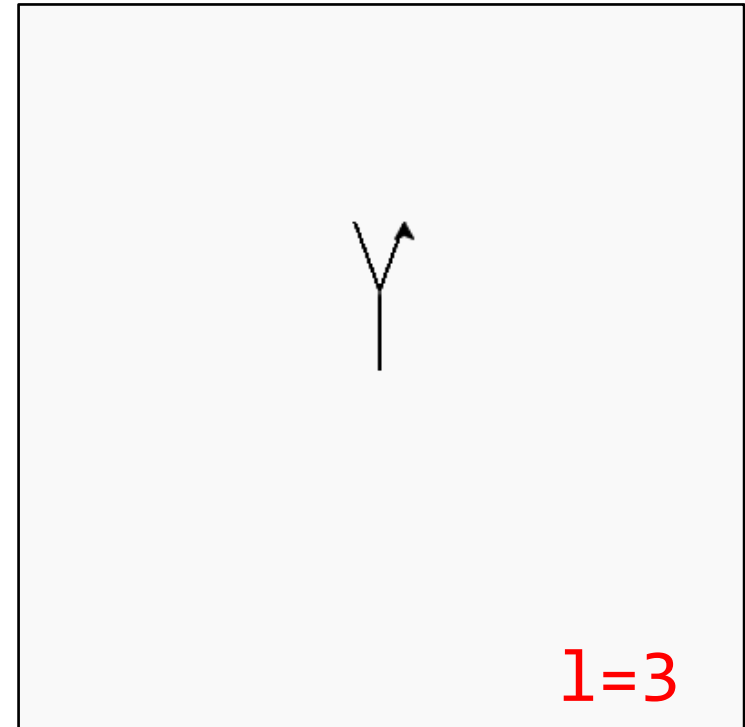
↓

↓

↓

↓

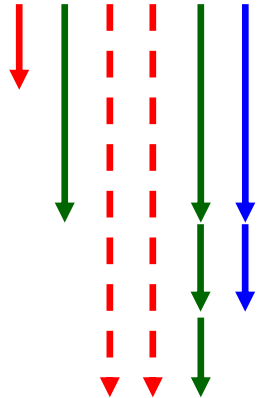
```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



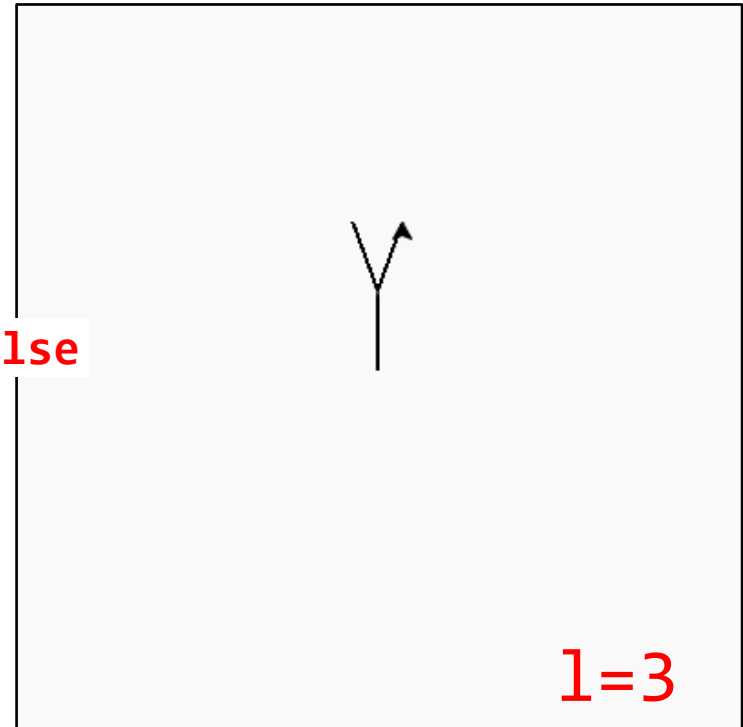
Call function with **l=3**

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function
level 2 function
level 3 function
level 3 function
level 2 function
level 1 function



```
trinket Run ? Modules
main.py
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit: False
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 2 function

level 3 function

level 3 function

level 2 function

level 1 function

↓

↓

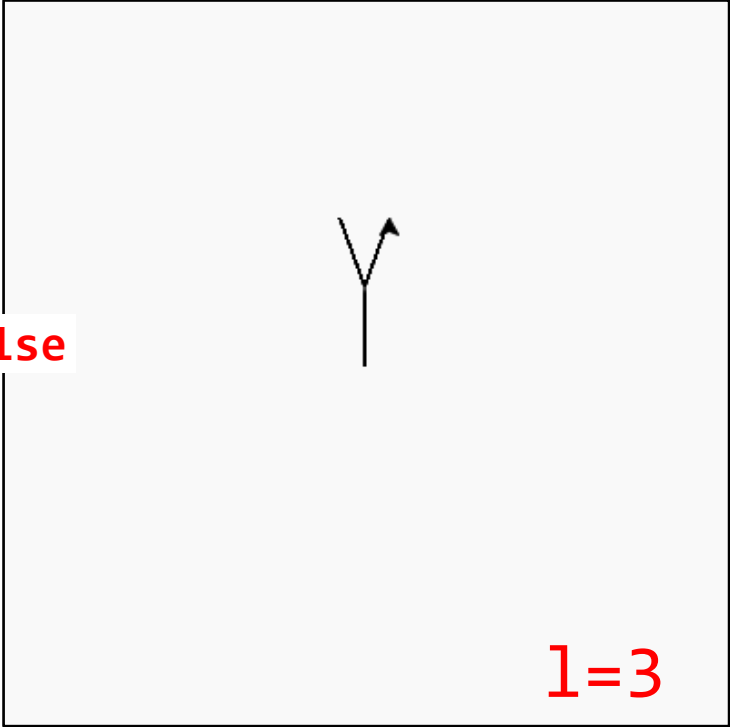
↓

↓

↓

↓

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit: False
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



l=3

Complete level 3 fn

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 2 function

level 3 function

level 3 function

level 2 function

level 1 function

↓

↓

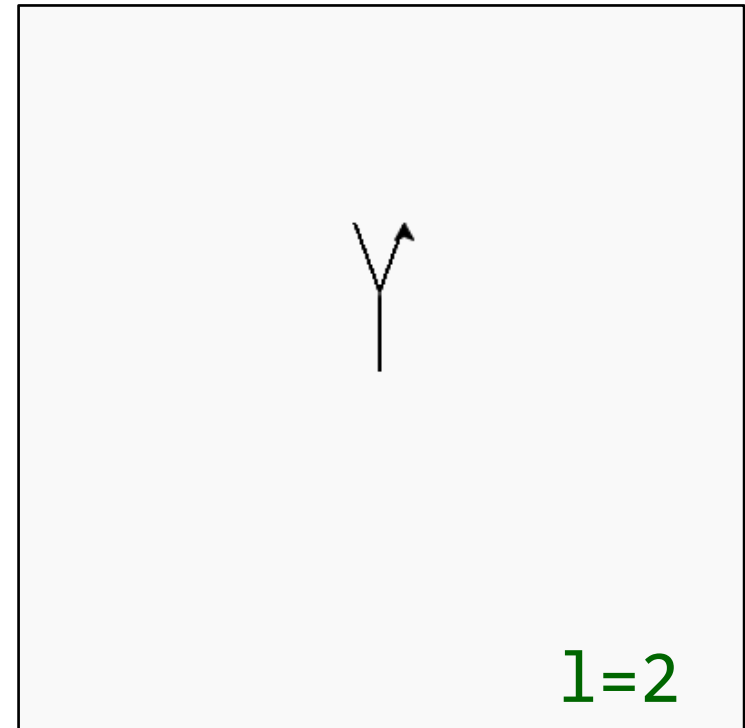
↓

↓

↓

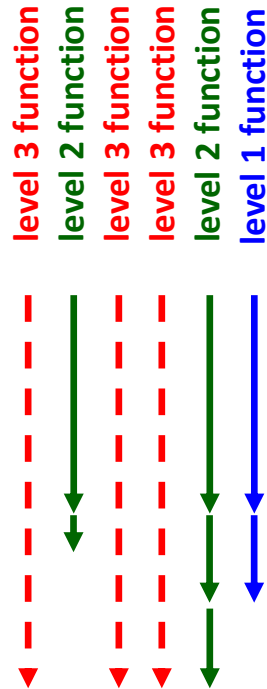
↓

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```

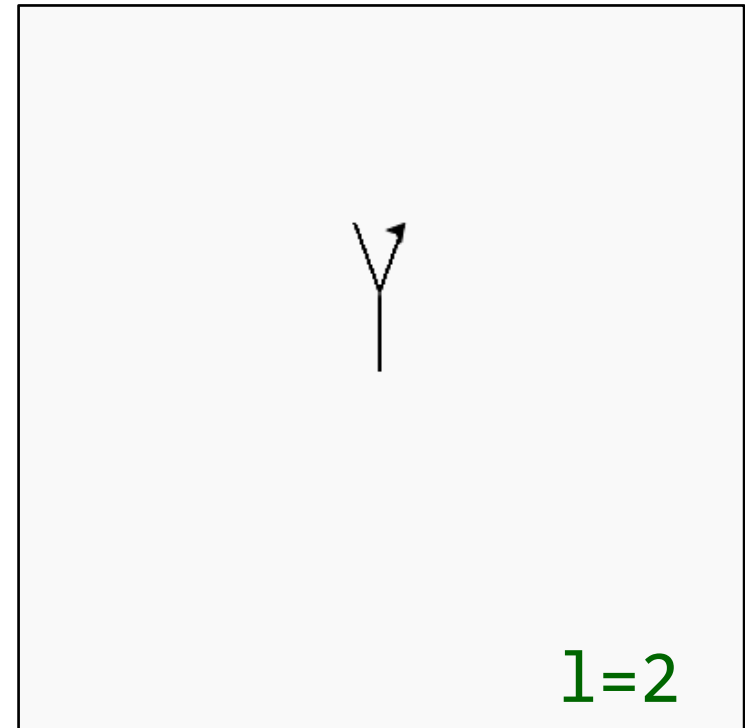


Continue unfinished level 2 fn

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



```
trinket
main.py
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 2 function

level 3 function

level 3 function

level 2 function

level 1 function

↓

↓

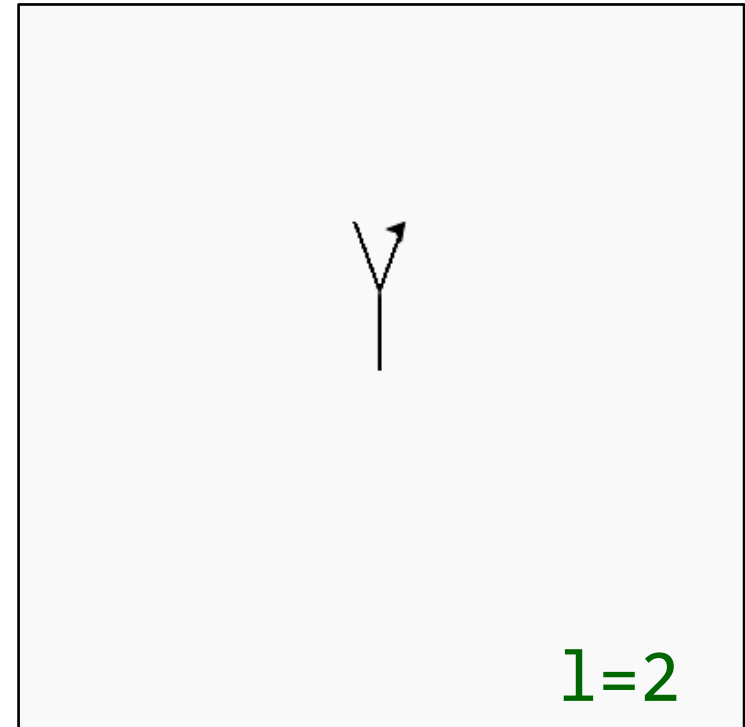
↓

↓

↓

↓

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Call function with **l=3**

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 3 function

level 2 function

level 3 function

level 3 function

level 2 function

level 1 function

↓

↓

↓

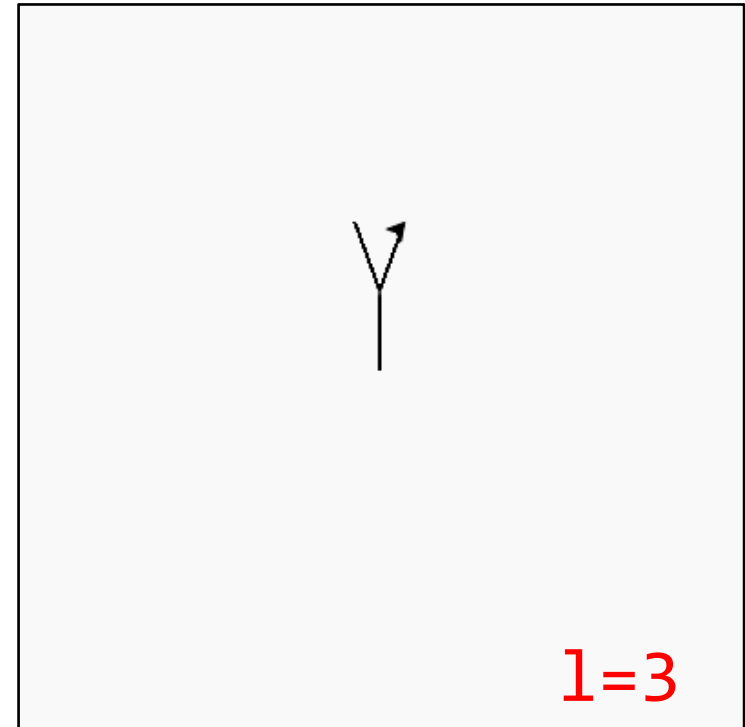
↓

↓

↓

↓

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Call function with **l=3**

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 3 function

level 2 function

level 3 function

level 3 function

level 2 function

level 1 function

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit: False
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```

Call function with $l=3$

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 3 function

level 2 function

level 3 function

level 3 function

level 2 function

level 1 function

↓

↓

↓

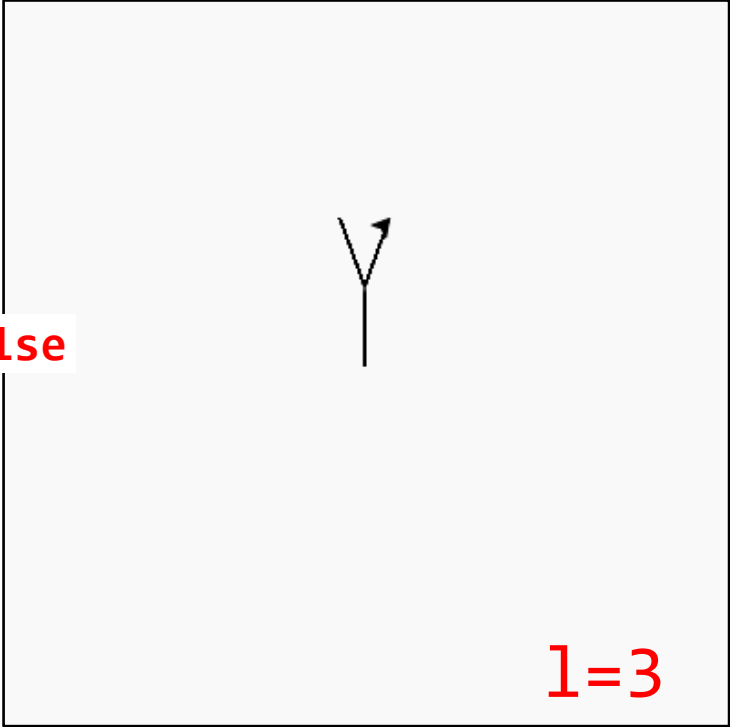
↓

↓

↓

↓

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit: False
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



l=3

Complete level 3 fn

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 3 function

level 2 function

level 3 function

level 3 function

level 2 function

level 1 function

↓

↓

↓

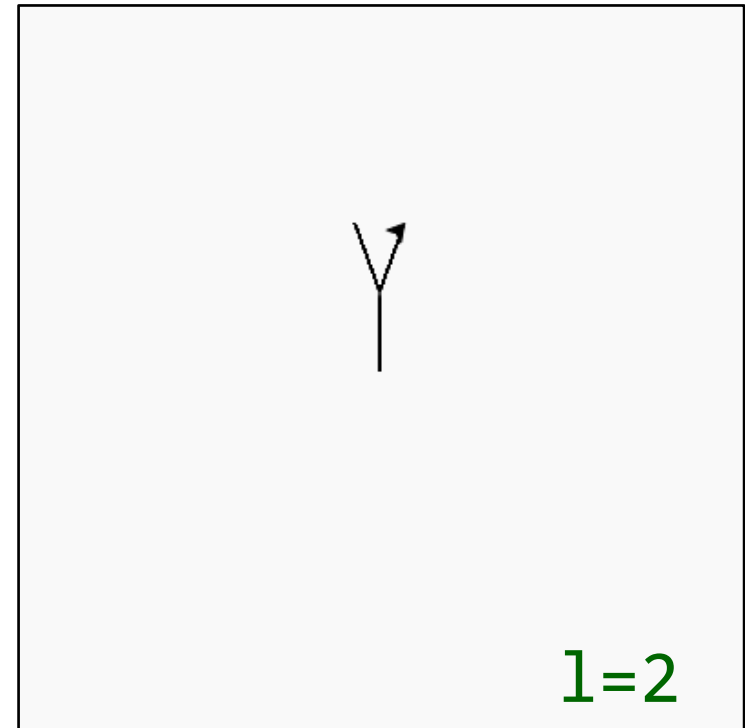
↓

↓

↓

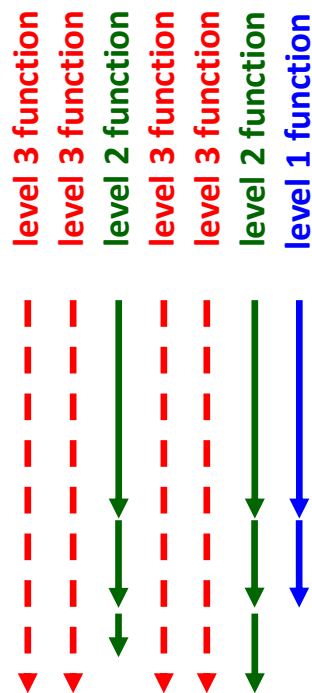
↓

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```

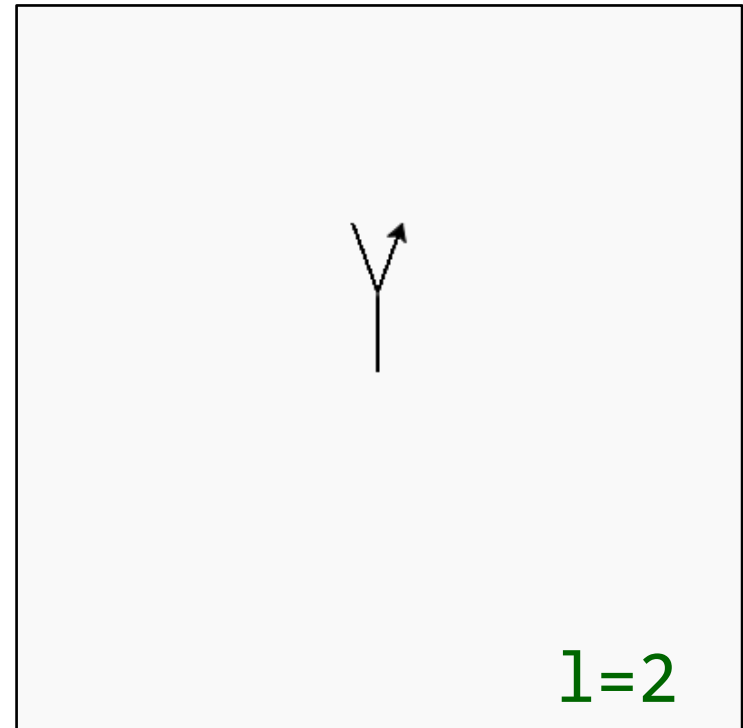


Continue unfinished level 2 fn

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



```
trinket Run Modules  
main.py  
32  
33 trunk = 40  
34 level_limit = 2  
35  
36 # Recursive function  
37 def fwdtrn(d,l):  
38     if l <= level_limit:  
39         t.fd(d)  
40         t.lt(20)  
41         fwdtrn(d*0.9, l+1)  
42         t.rt(40)  
43         fwdtrn(d*0.9, l+1)  
44         t.lt(20)  
45         t.bk(d)  
46  
47 fwdtrn(trunk,1)
```



Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 3 function

level 2 function

level 3 function

level 3 function

level 2 function

level 1 function

↓

↓

↓

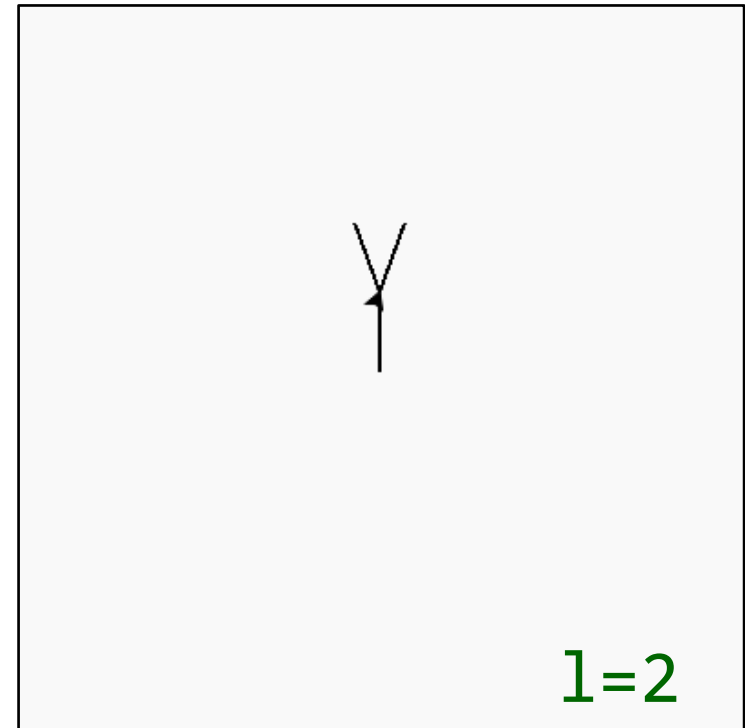
↓

↓

↓

↓

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Complete level 2 fn

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 3 function

level 2 function

level 3 function

level 3 function

level 2 function

level 1 function

↓

↓

↓

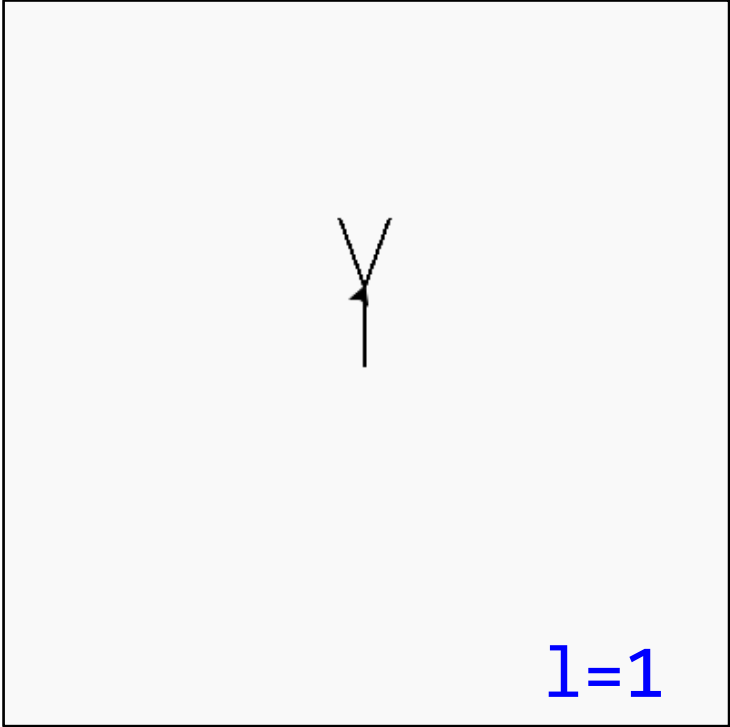
↓

↓

↓

↓

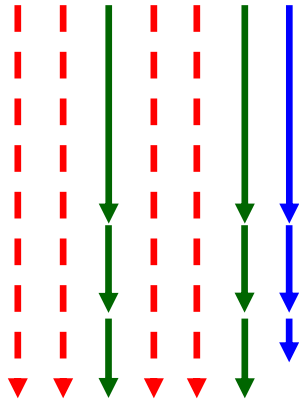
```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



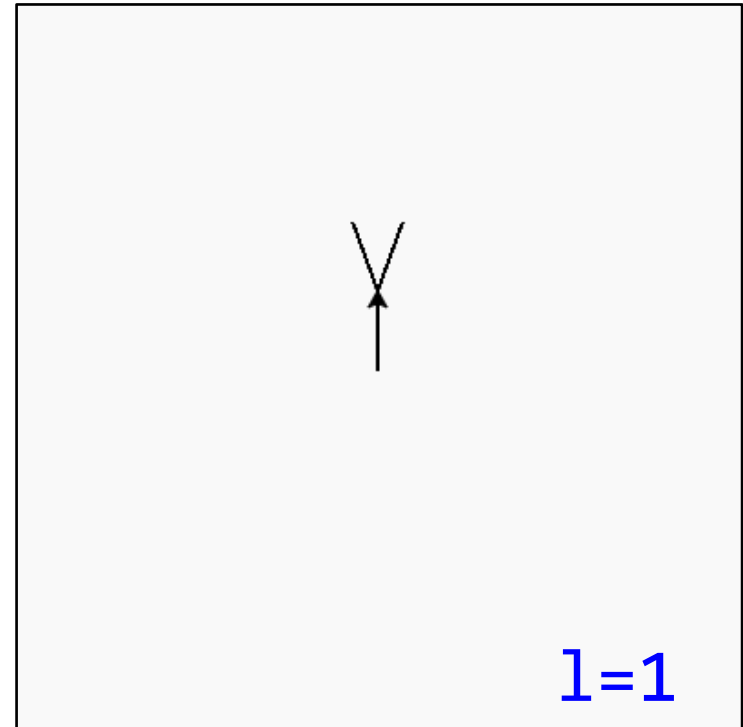
Continue unfinished level 1 fn

Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function
level 3 function
level 2 function
level 3 function
level 3 function
level 2 function
level 1 function



```
trinket Run ? Modules
main.py
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function

level 3 function

level 2 function

level 3 function

level 3 function

level 2 function

level 1 function

↓

↓

↓

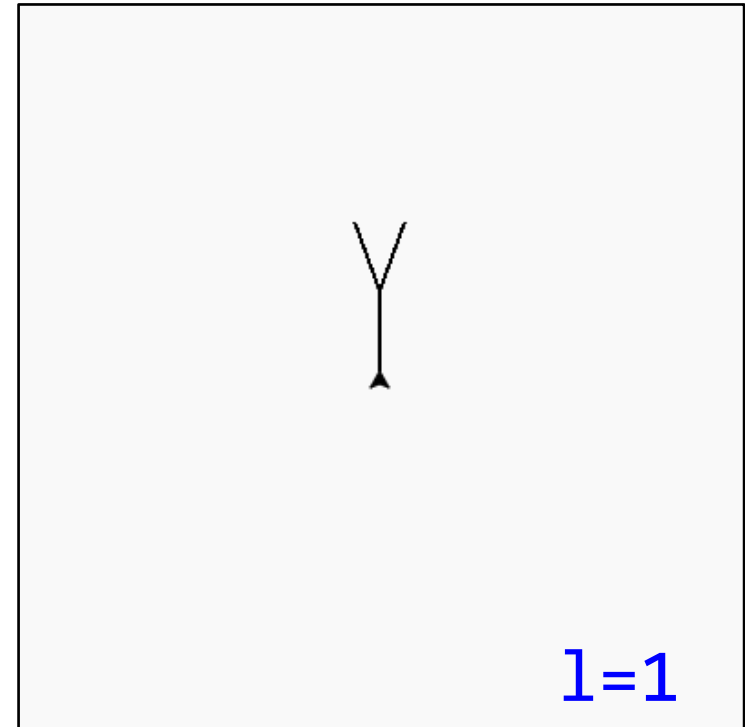
↓

↓

↓

↓

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38     if l <= level_limit:
39         t.fd(d)
40         t.lt(20)
41         fwdtrn(d*0.9, l+1)
42         t.rt(40)
43         fwdtrn(d*0.9, l+1)
44         t.lt(20)
45         t.bk(d)
46
47 fwdtrn(trunk,1)
```



Complete level 1 fn - END CODE

If you set the level limit up to 3 and went through the same sequence you would find that there were

- 1 level 1 function
- 2 level 2 functions
- 4 level 3 functions
- 8 failed level 4 functions

and the number of steps in the sequence would approximately double!

Each time you add one more level the number of steps roughly doubles.

(That's why it seemed a good idea to limit this document to a level limit of just 2.)

