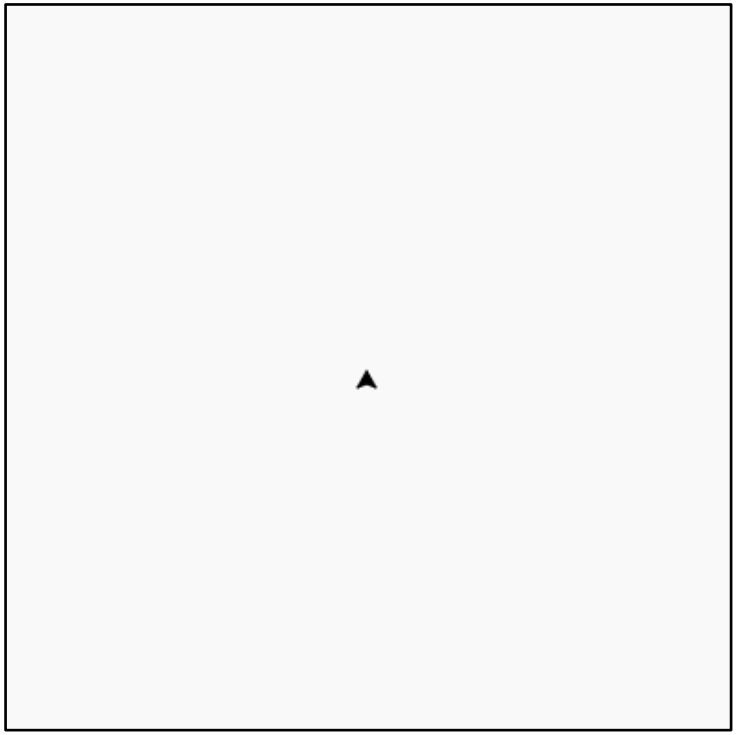
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

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

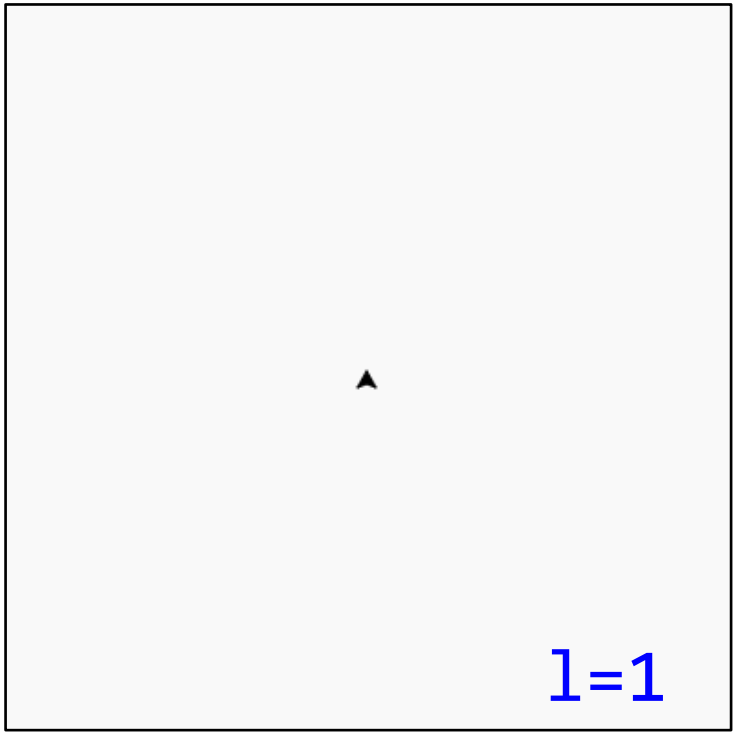
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 1 function

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

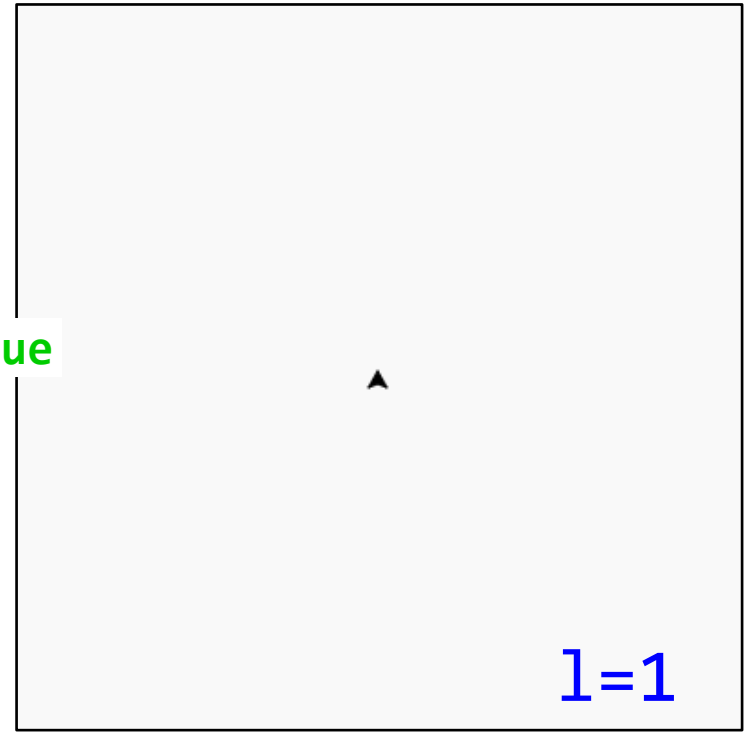
level 1 function

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)
```

True



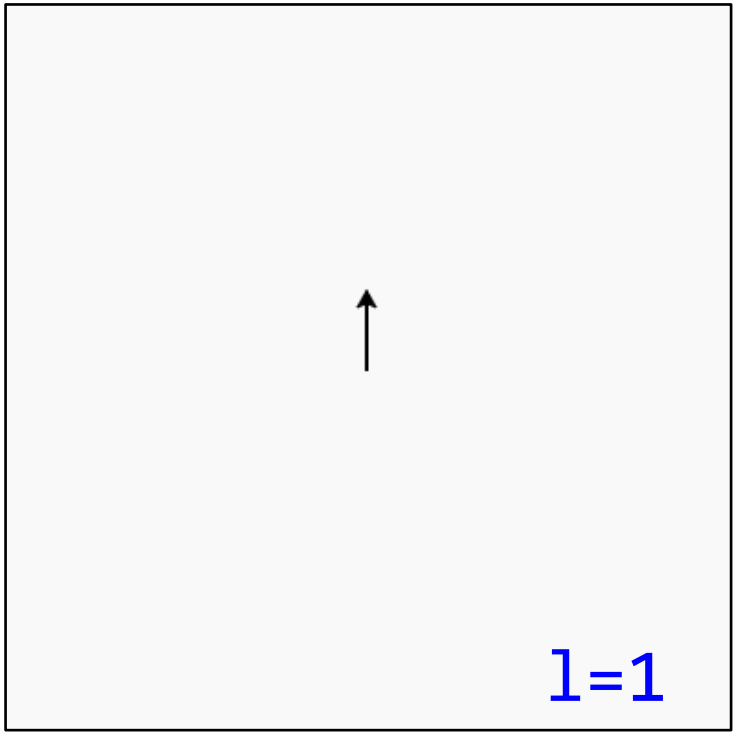
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

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)
```



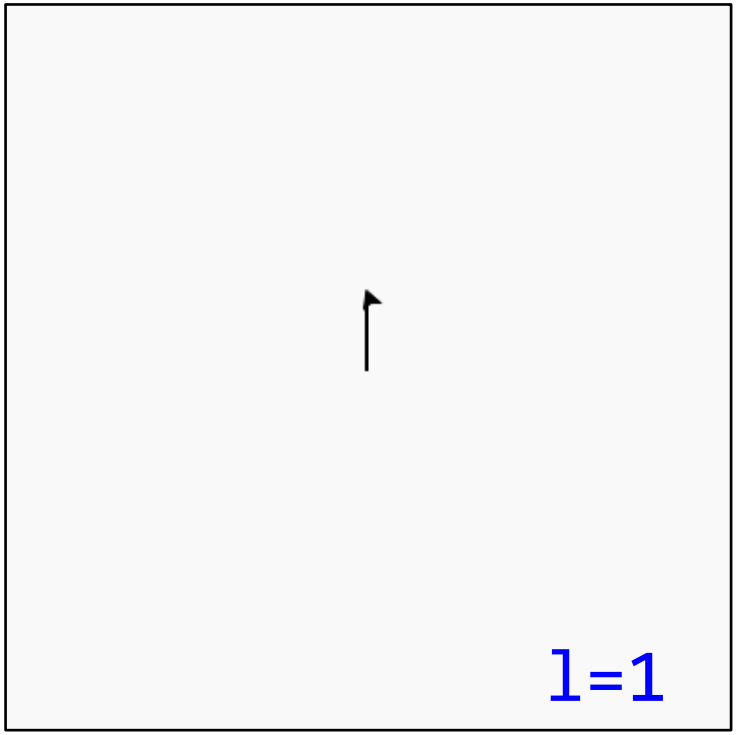
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 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)
```



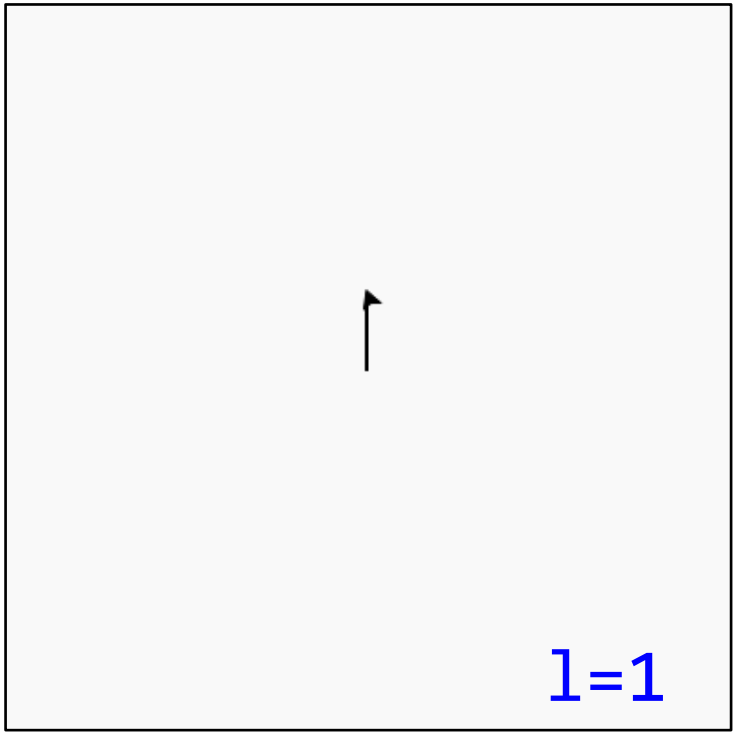
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 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)
```



Call function with l=2

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

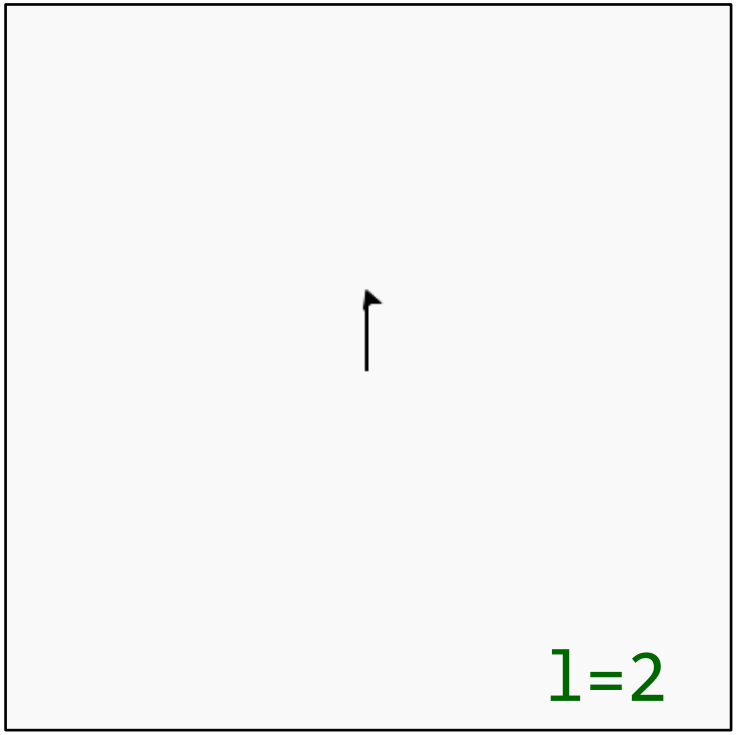
level 2 function

level 1 function

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=2

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

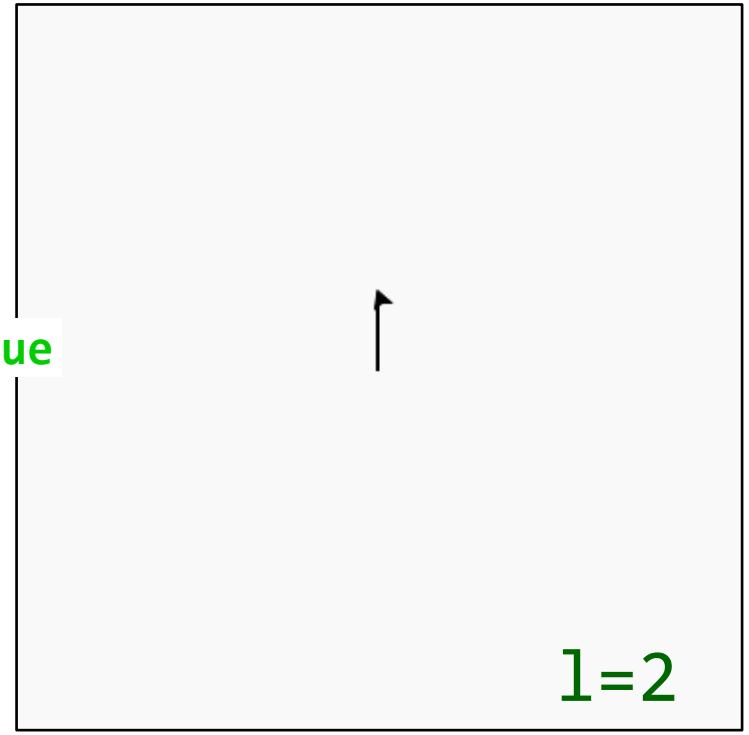
level 2 function

level 1 function

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: 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)
```



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

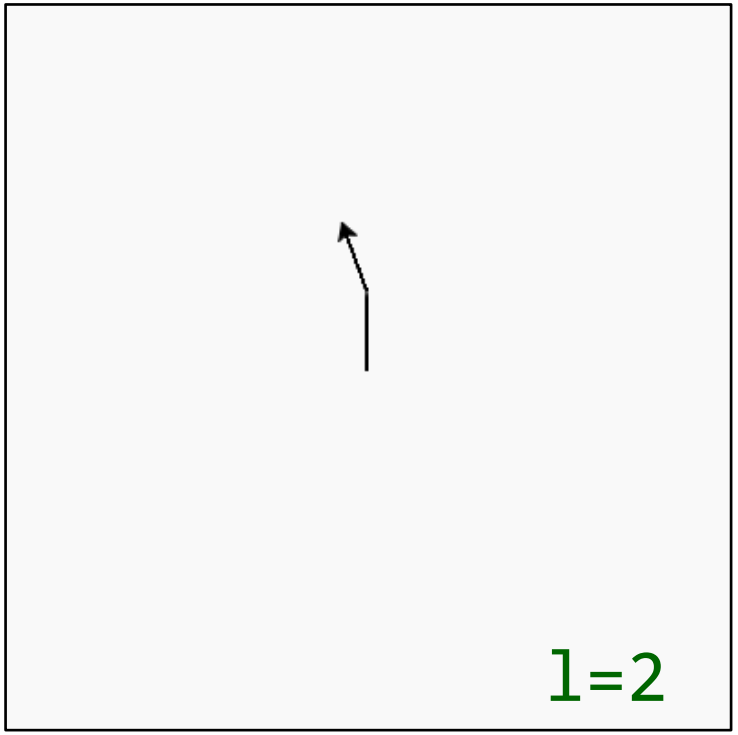
level 2 function

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

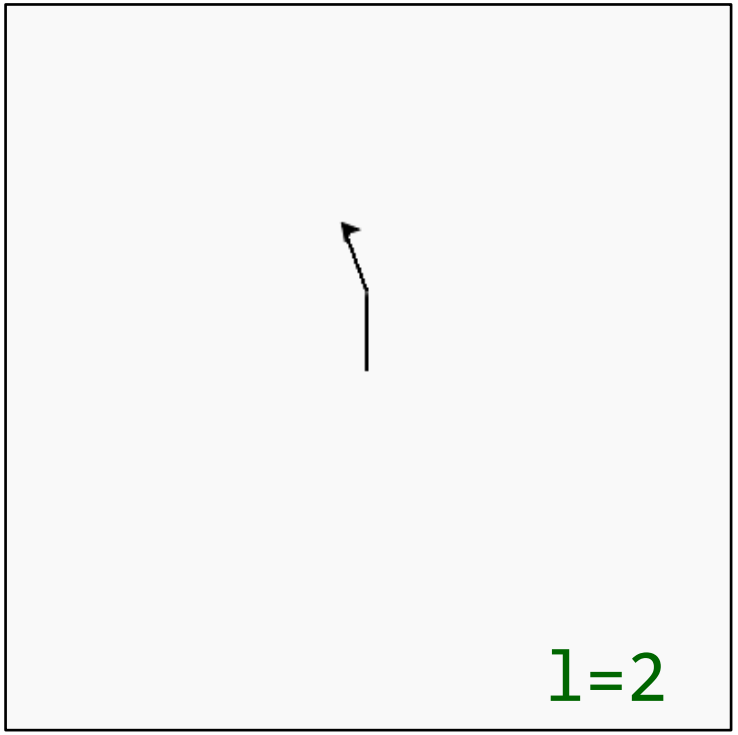
level 2 function

level 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

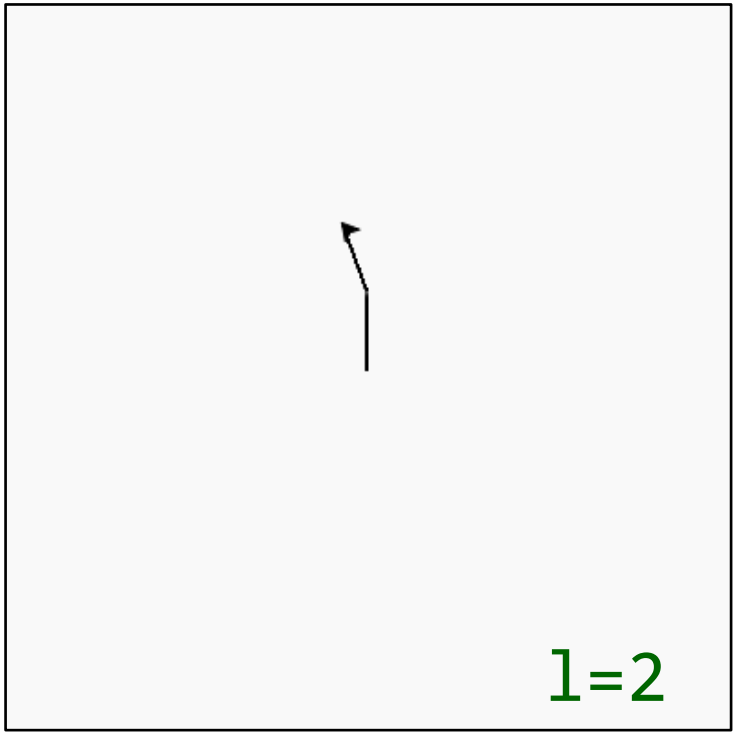
level 2 function

level 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)
```



Call function with l=3

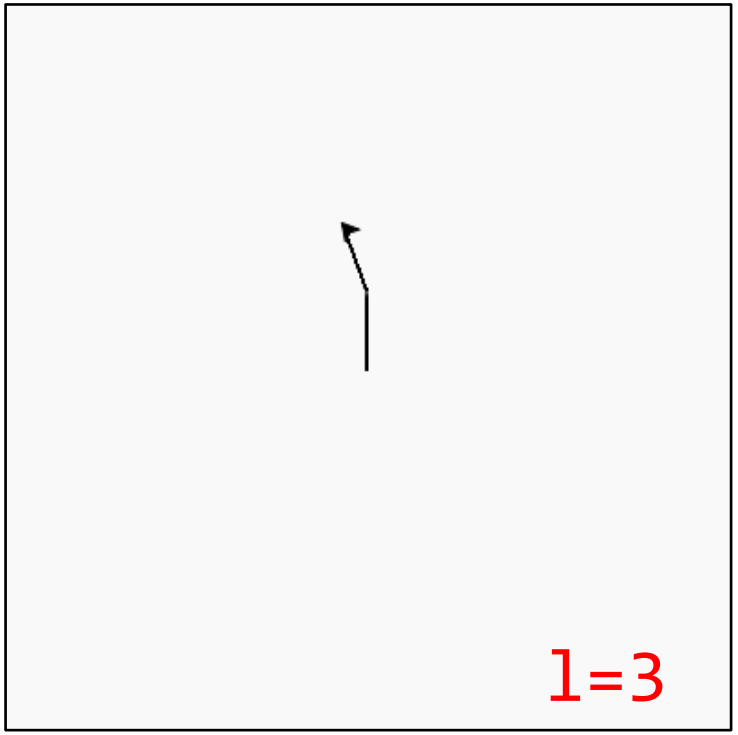
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function
level 2 function
level 1 function

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=3

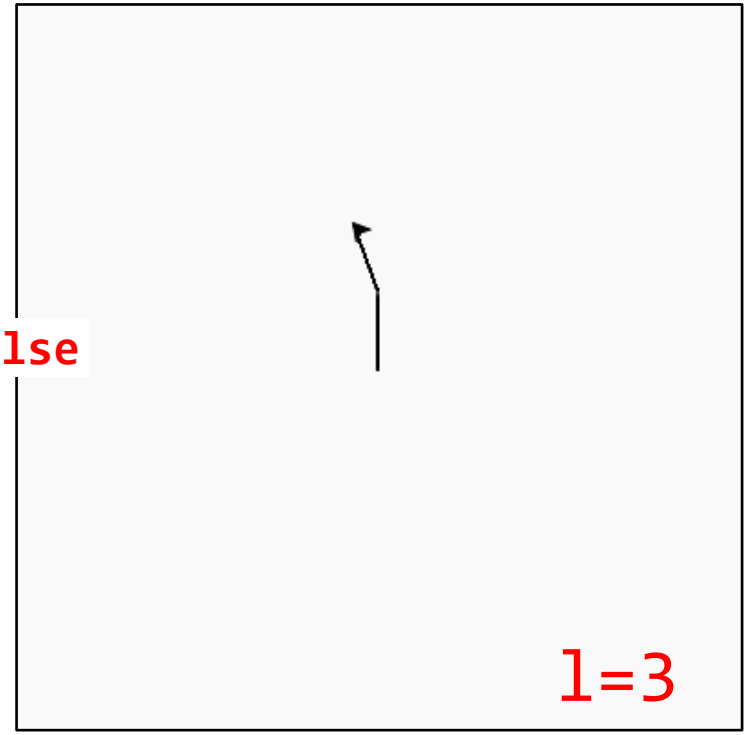
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

level 3 function
level 2 function
level 1 function

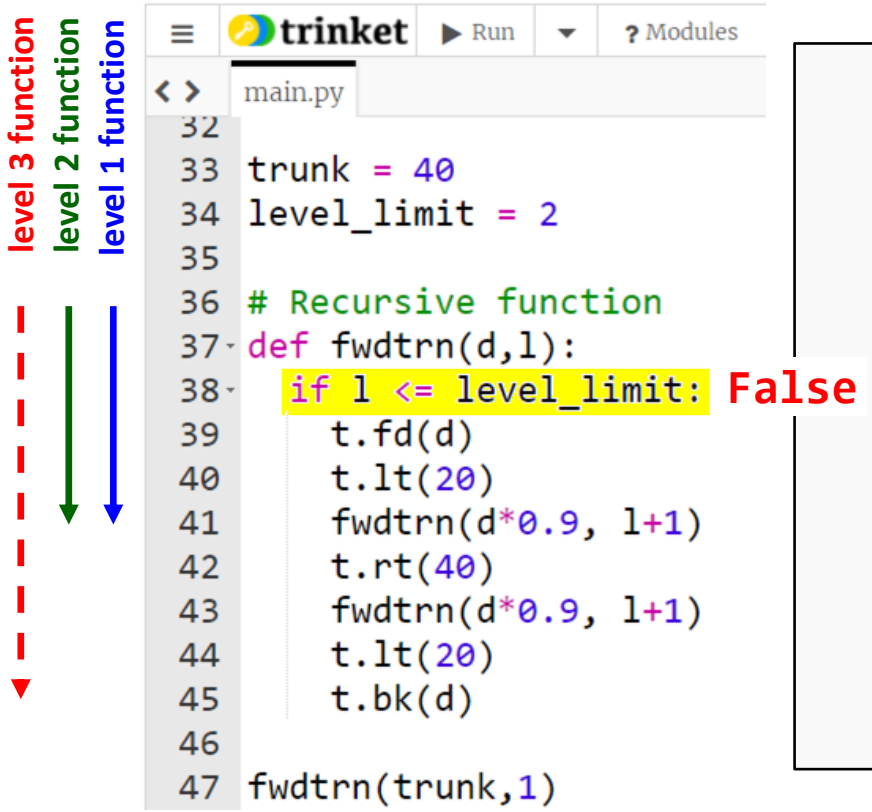
trinket ▶ Run ? Modules

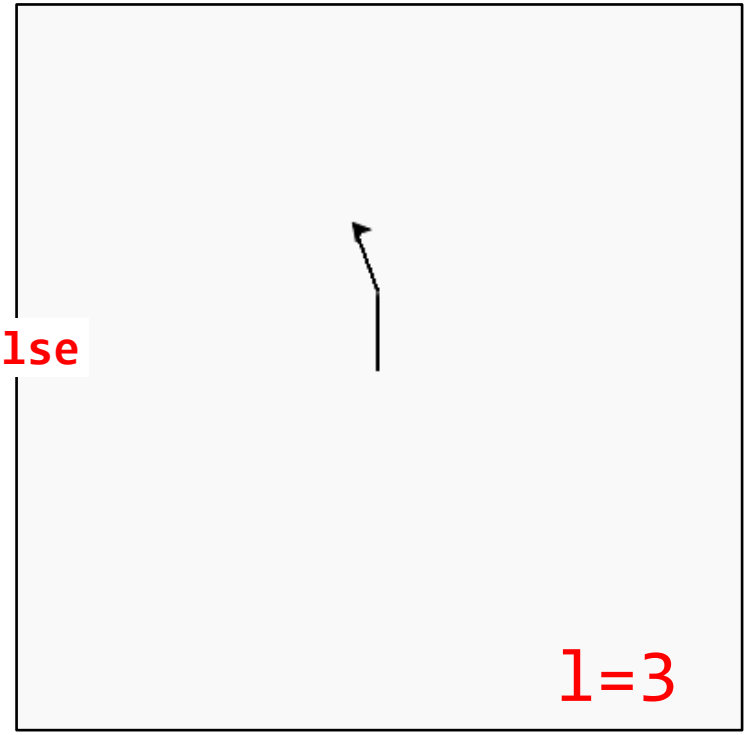
main.py

```
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)
```



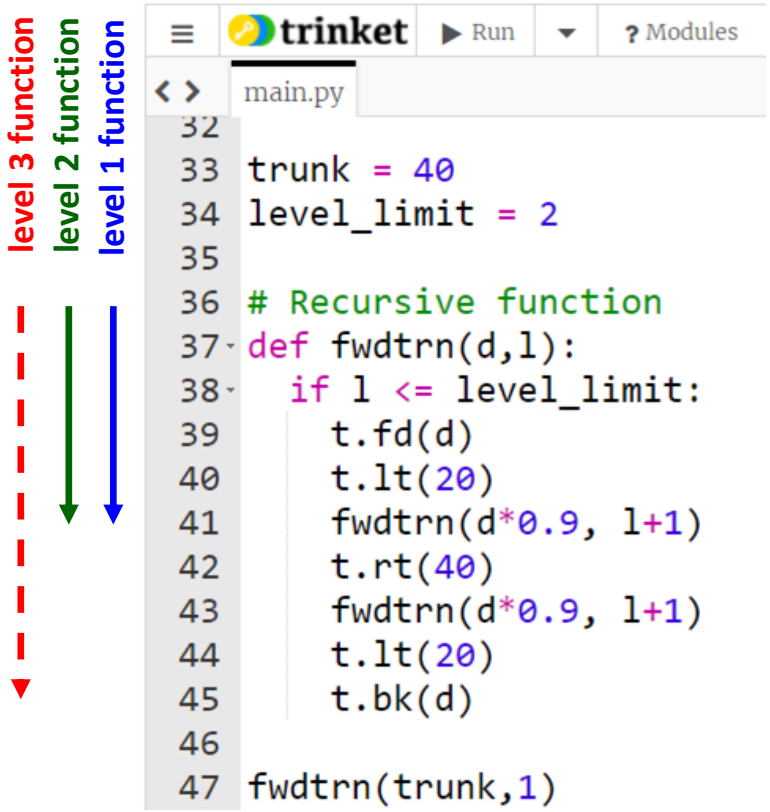
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

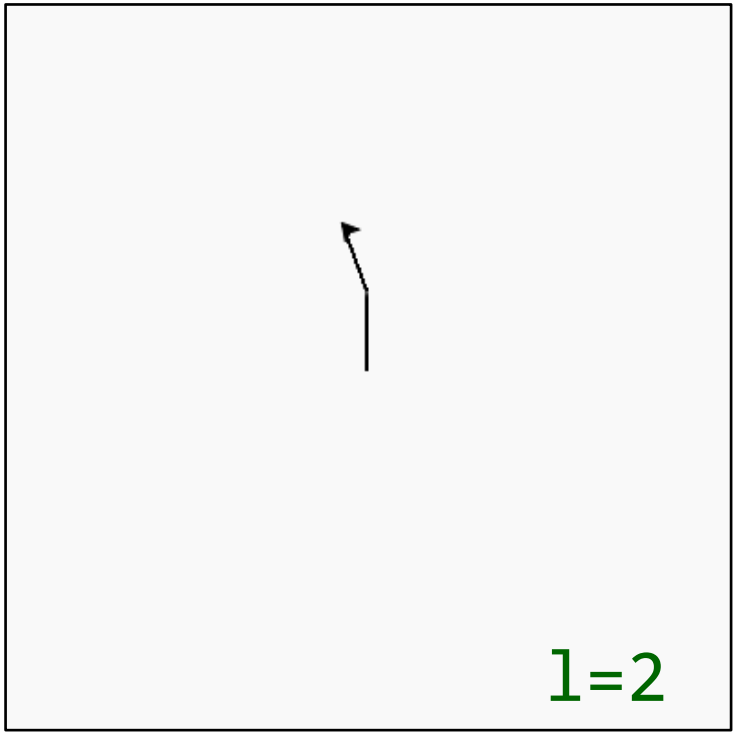




Complete level 3 fn

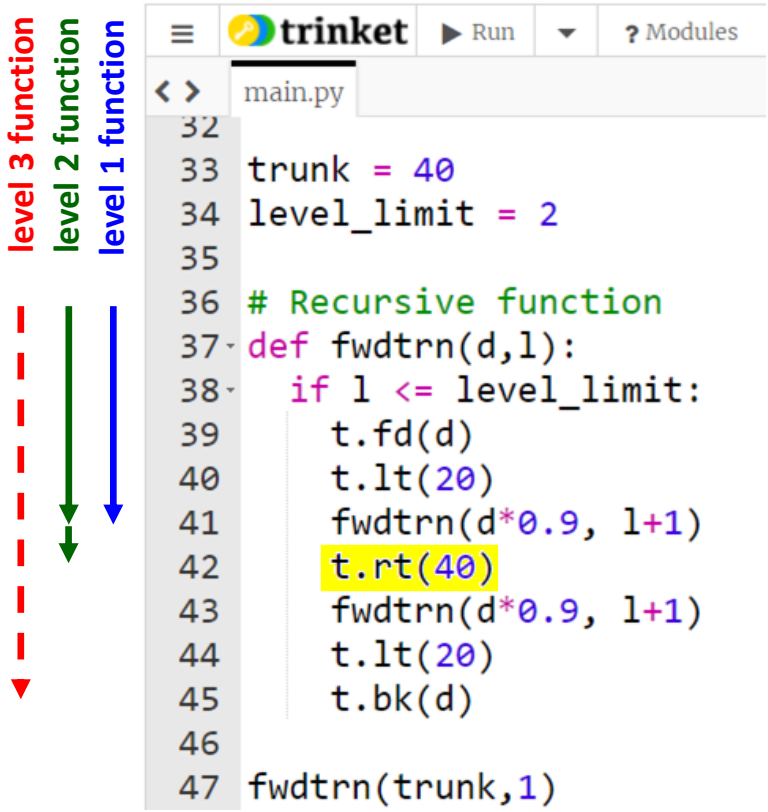
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

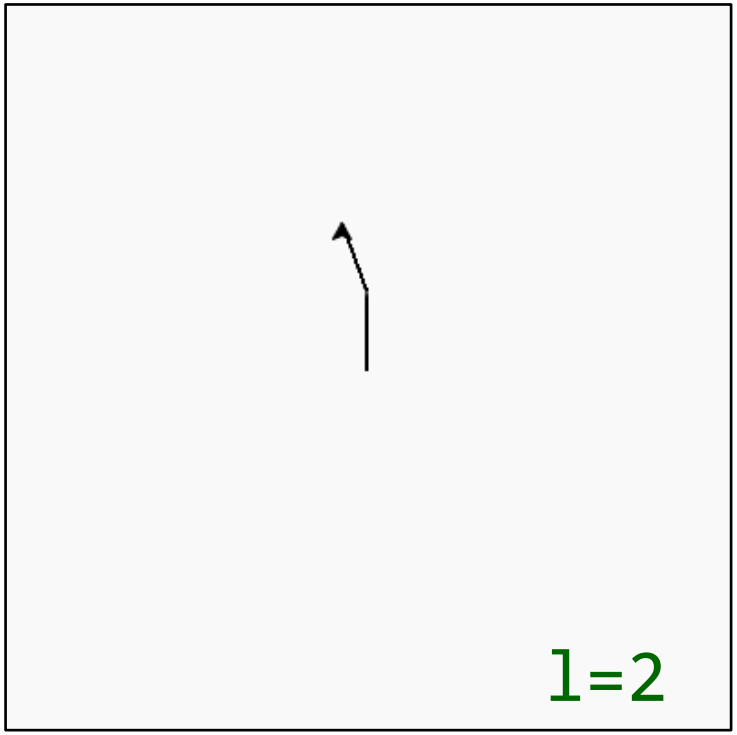




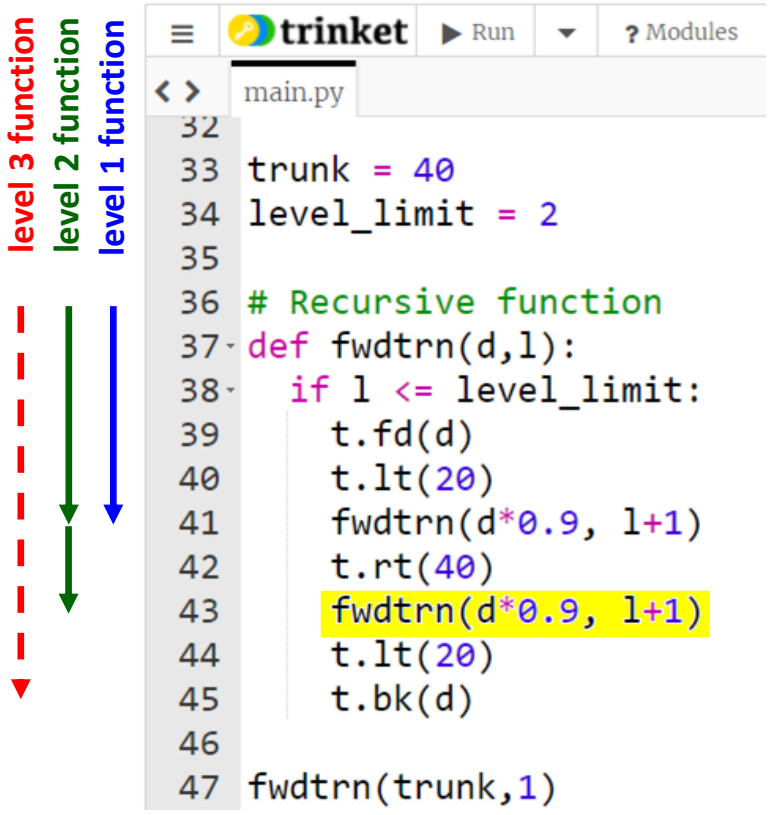
Continue unfinished level 2 fn

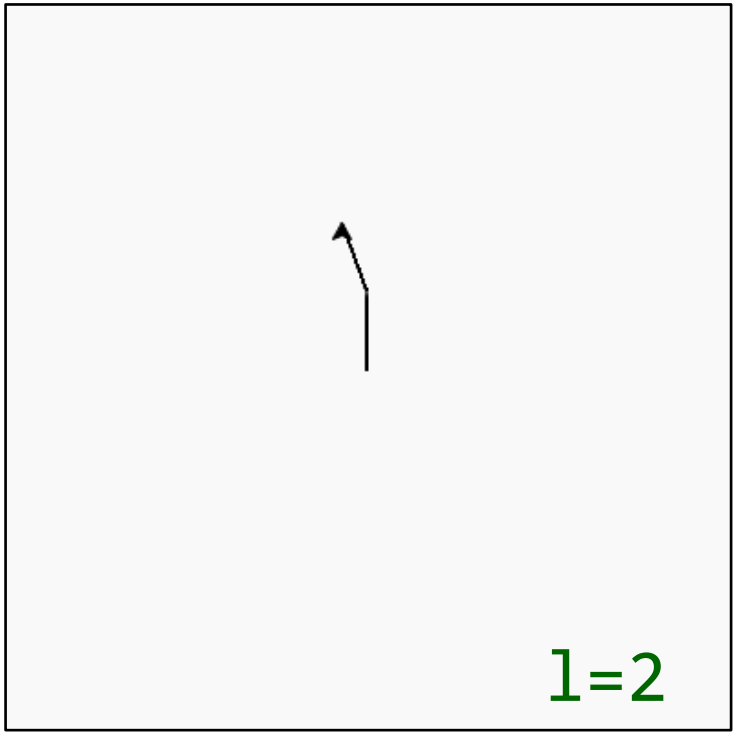
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2





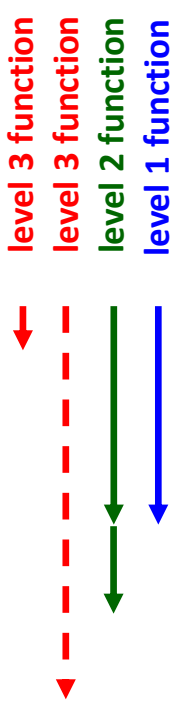
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2





Call function with l=3

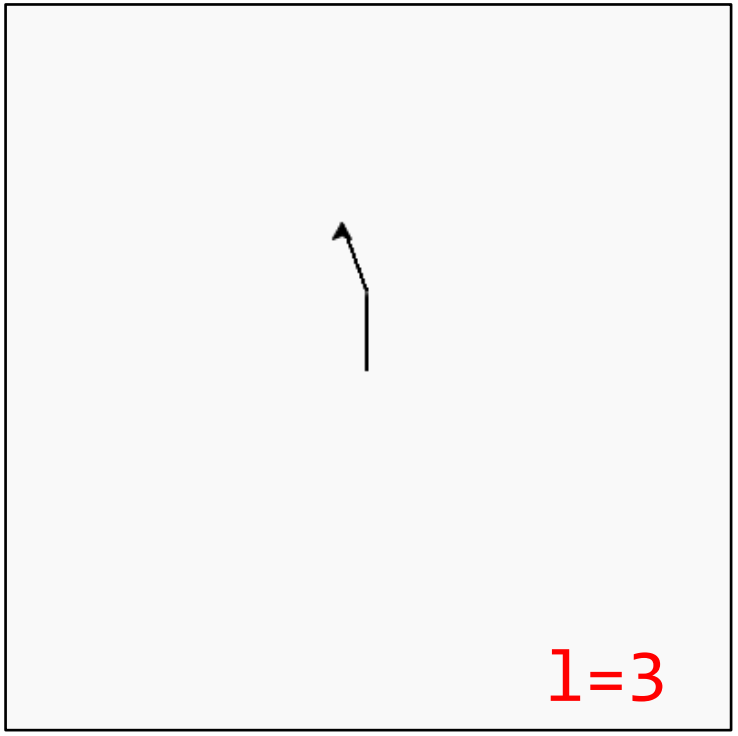
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

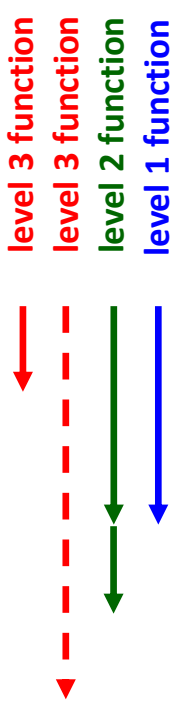
main.py

```
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=3

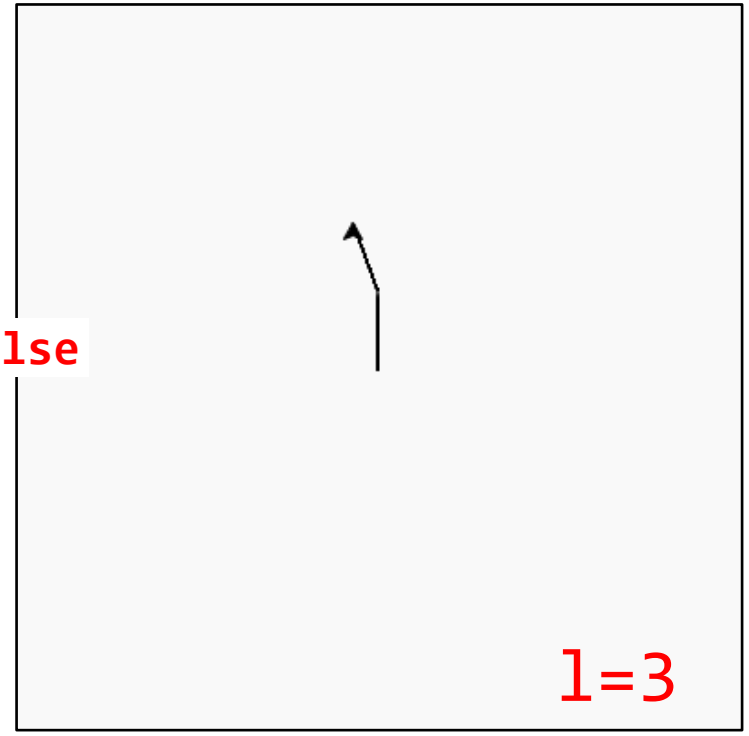
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



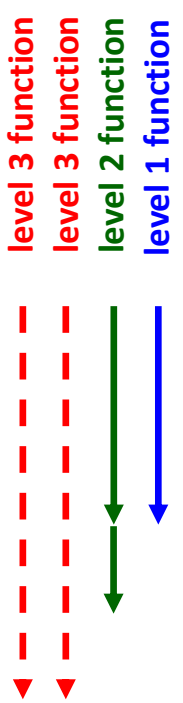
trinket ▶ Run ? Modules

main.py

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38   if l <= level_limit: 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

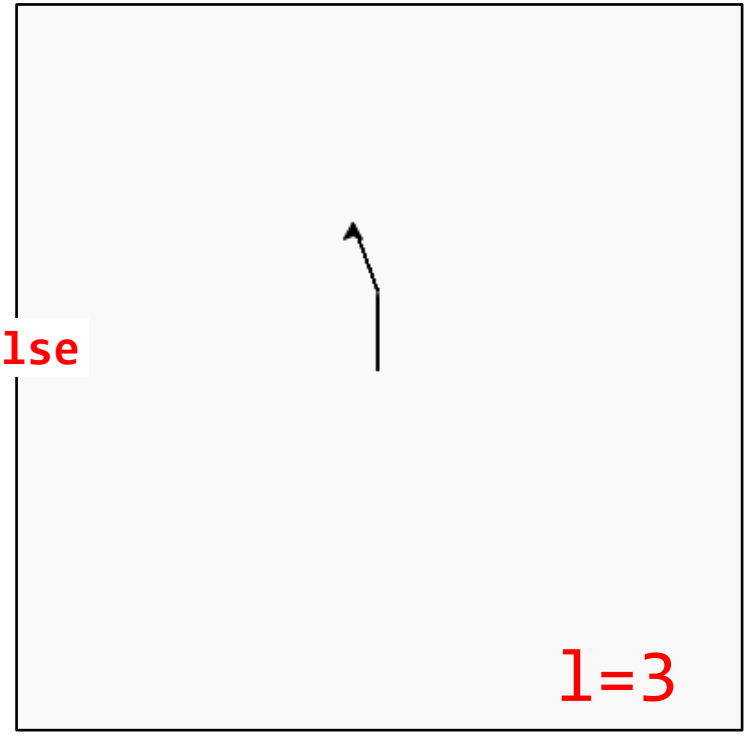


trinket ▶ Run ? Modules

main.py

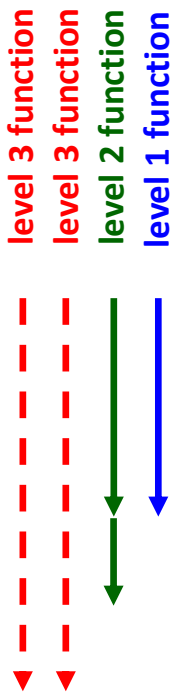
```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38   if l <= level_limit:
39     t.fd(d)
40     t.lt(20)
41     fwdtrn(d*0.9, l+1)
42     t.rt(40)
43     fwdtrn(d*0.9, l+1)
44     t.lt(20)
45     t.bk(d)
46
47 fwdtrn(trunk,1)
```

False



Complete level 3 fn

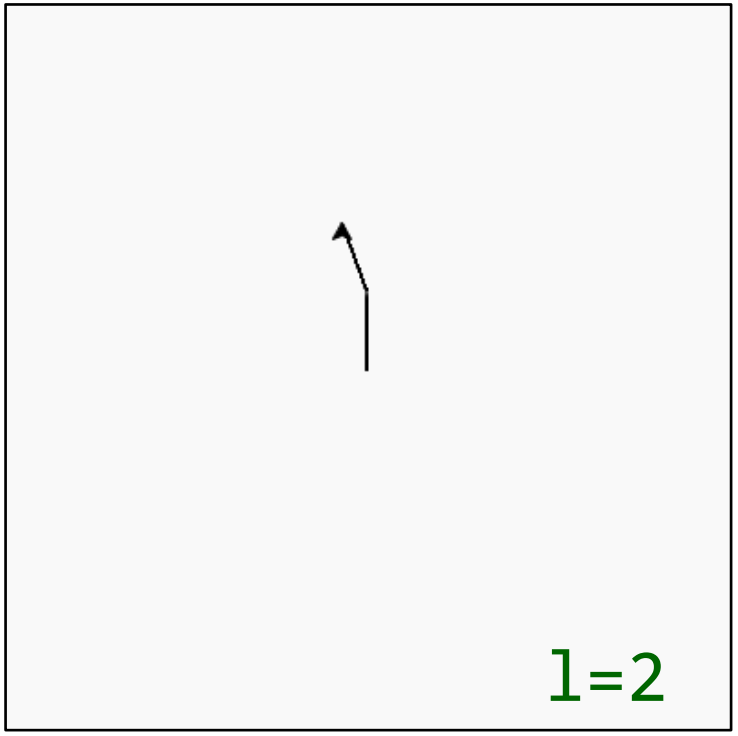
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

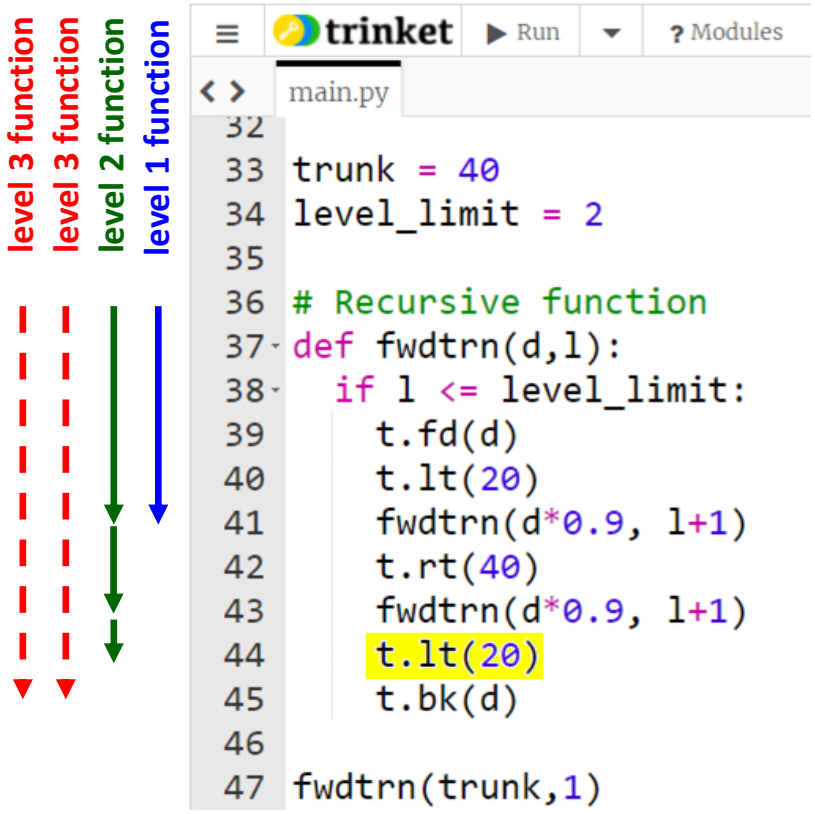
main.py

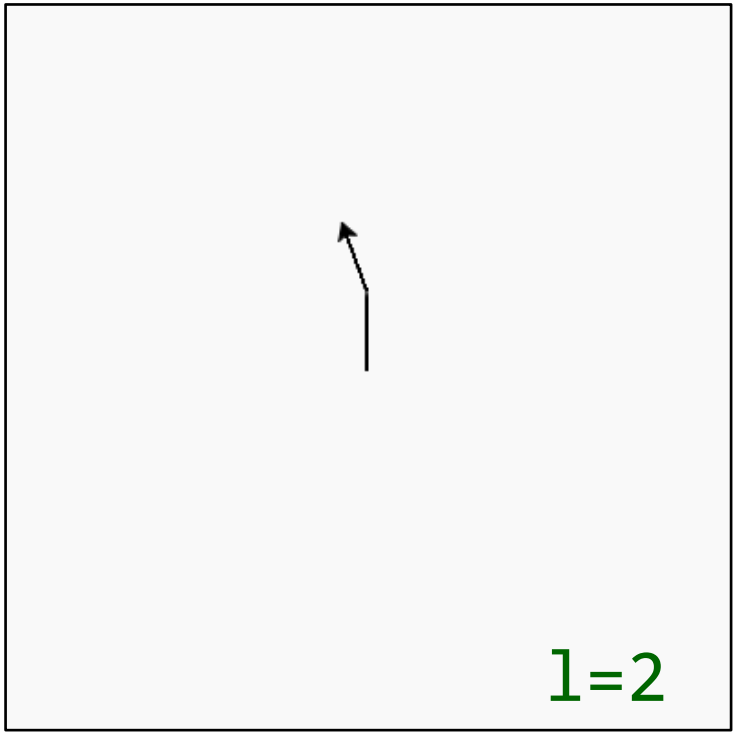
```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38   if l <= level_limit:
39     t.fd(d)
40     t.lt(20)
41     fwdtrn(d*0.9, l+1)
42     t.rt(40)
43     fwdtrn(d*0.9, l+1)
44     t.lt(20)
45     t.bk(d)
46
47 fwdtrn(trunk,1)
```



Continue unfinished level 2 fn

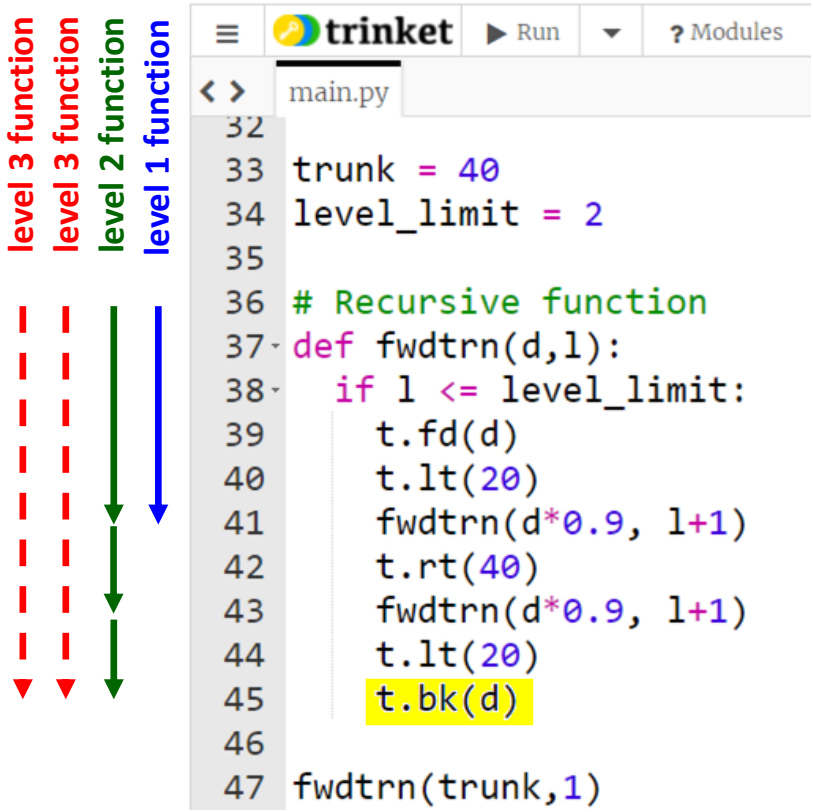
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2

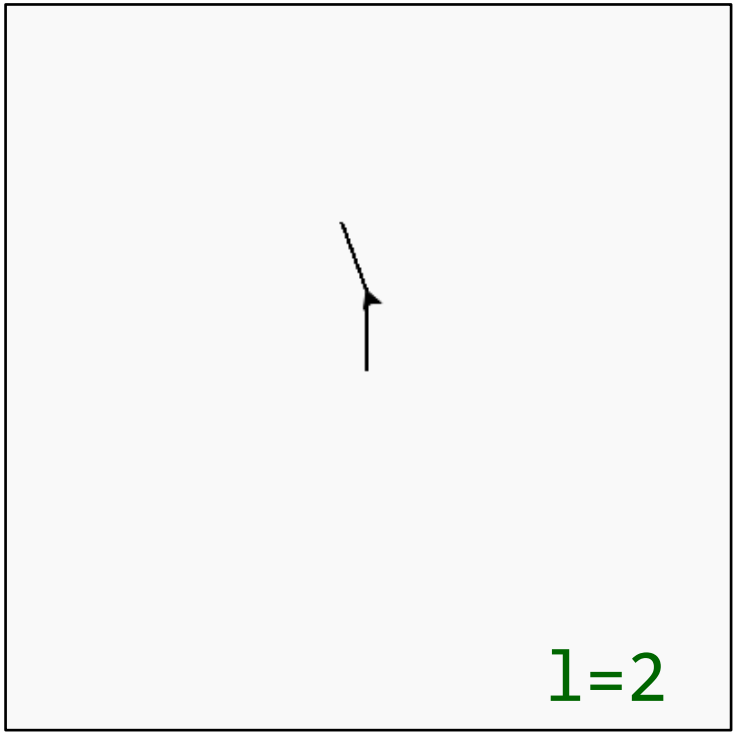




Continue unfinished level 2 fn

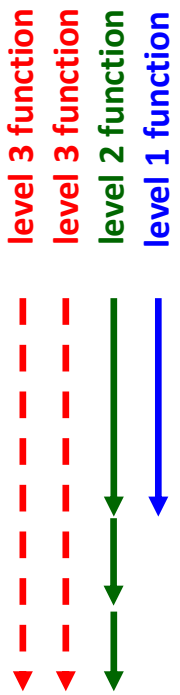
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2





Complete level 2 fn

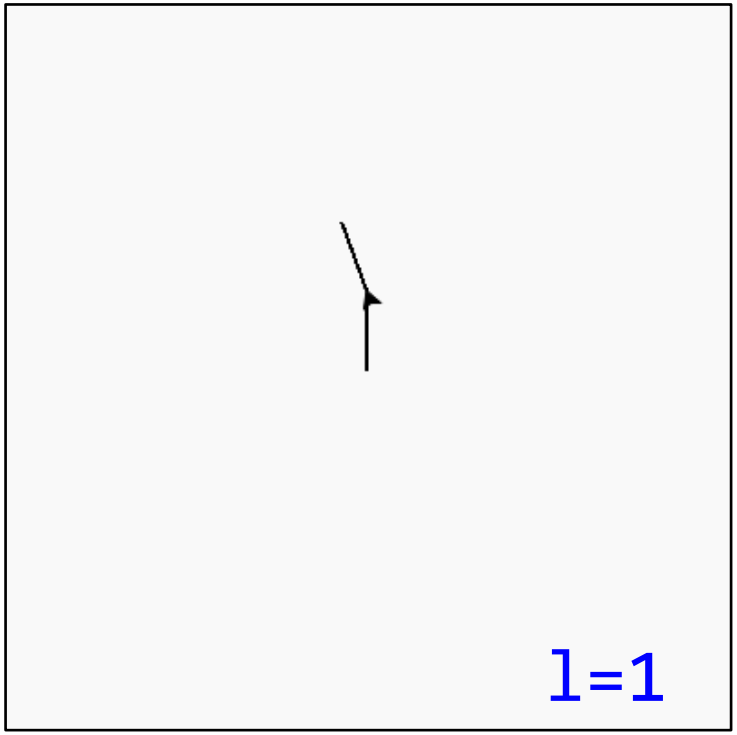
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

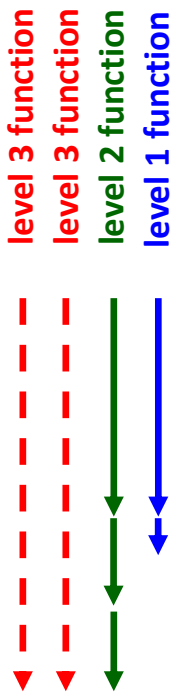
main.py

```
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)
```



Continue unfinished level 1 fn

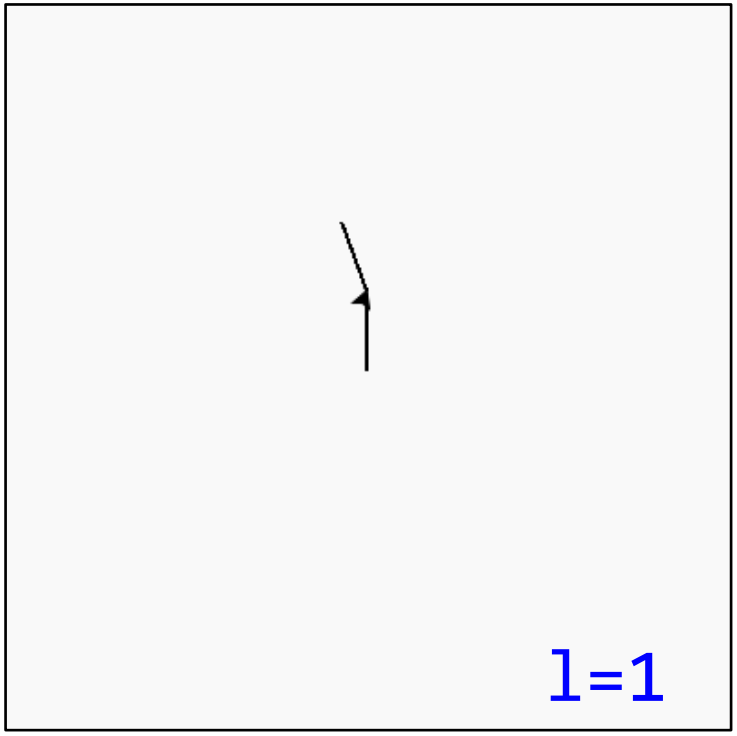
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



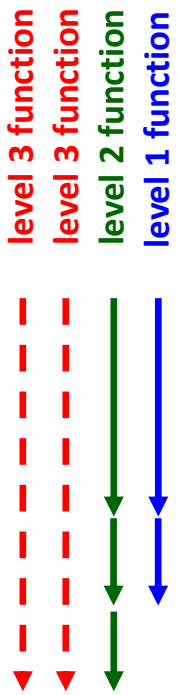
trinket ▶ Run ? Modules

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)
```



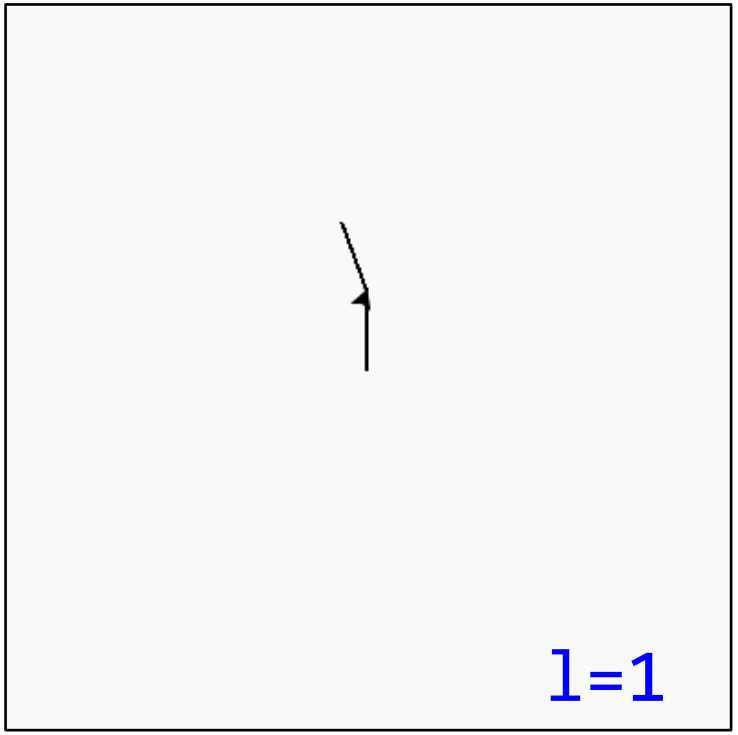
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

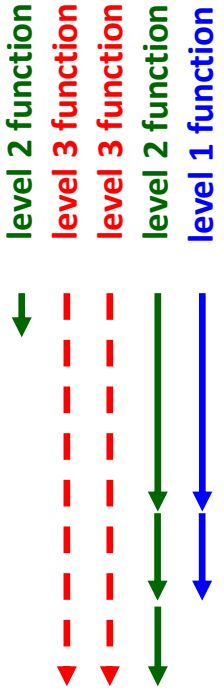
main.py

```
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=2

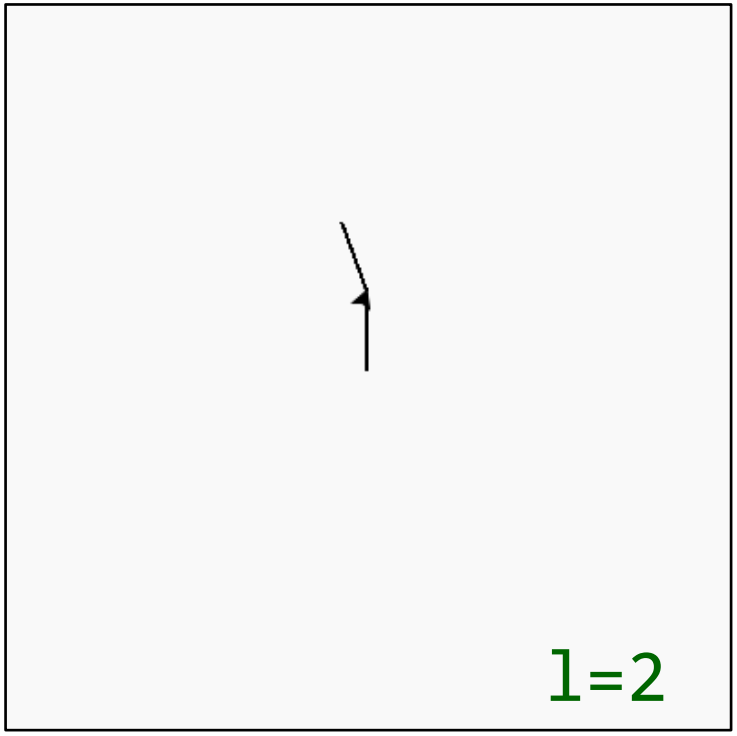
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

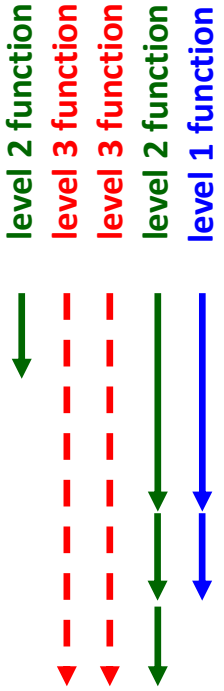
main.py

```
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=2

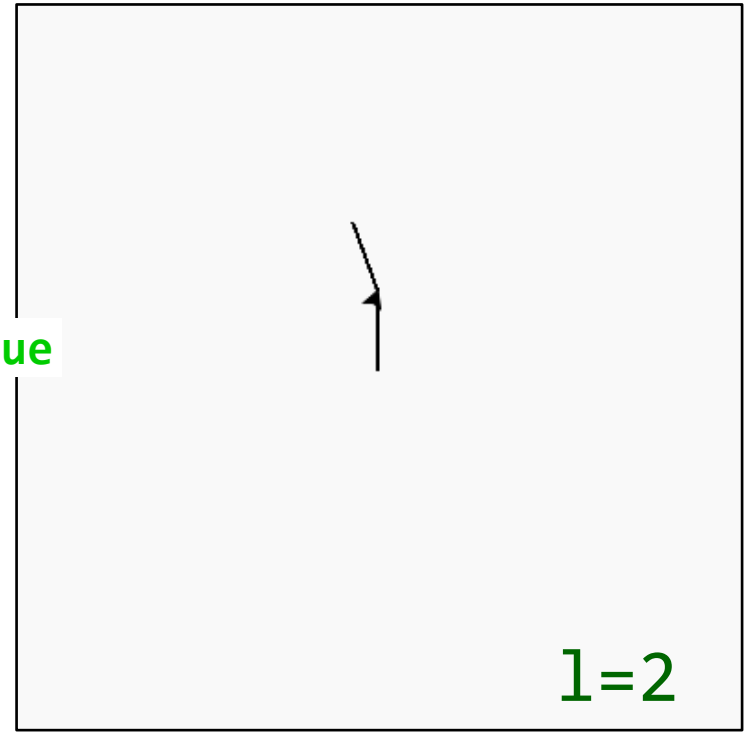
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



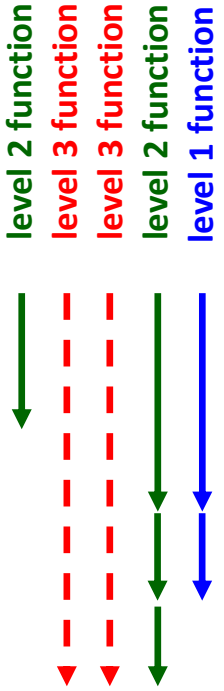
trinket ▶ Run ? Modules

main.py

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38   if l <= level_limit: 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)
```



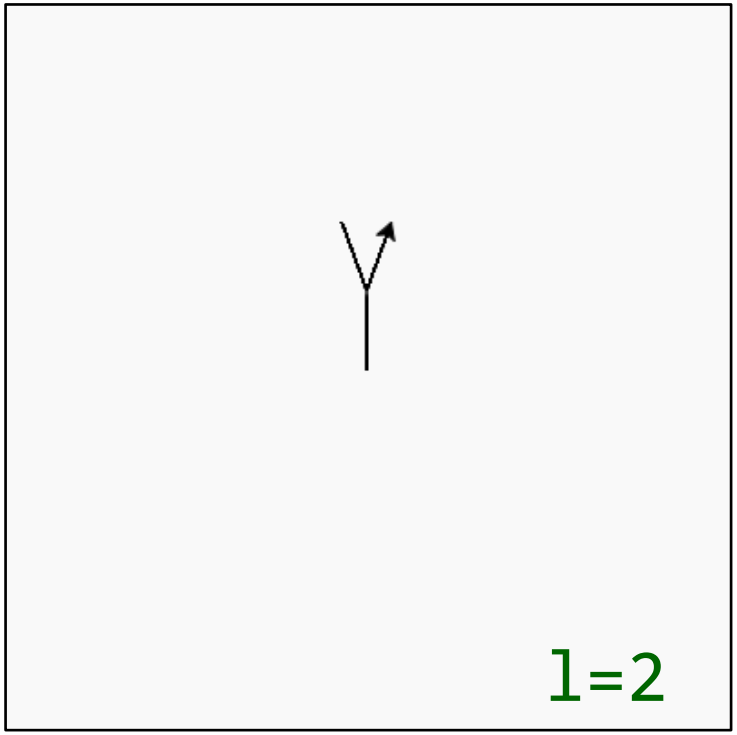
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



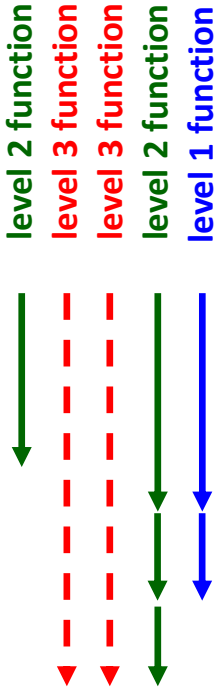
trinket ▶ Run ? Modules

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)
```



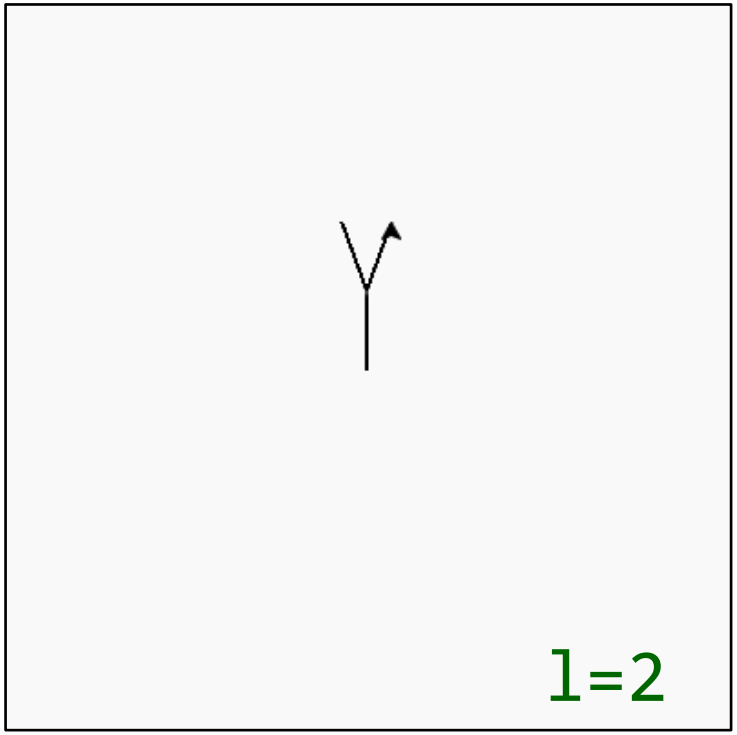
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



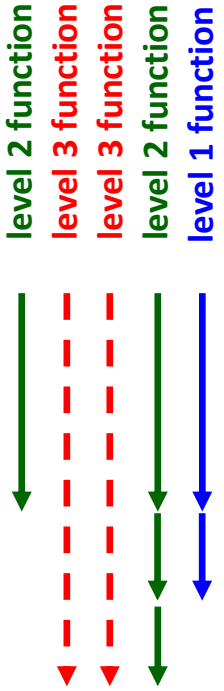
trinket ▶ Run ? Modules

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)
```



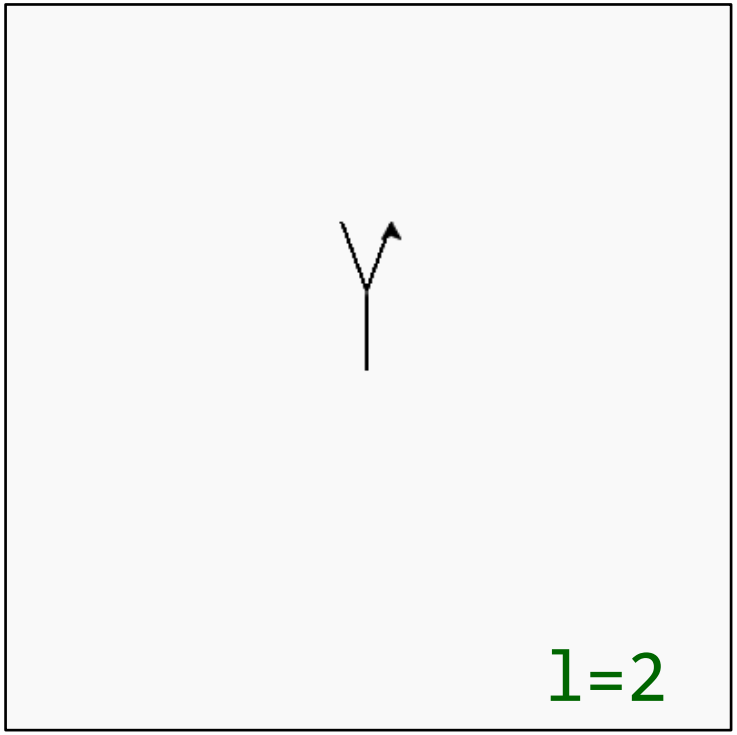
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

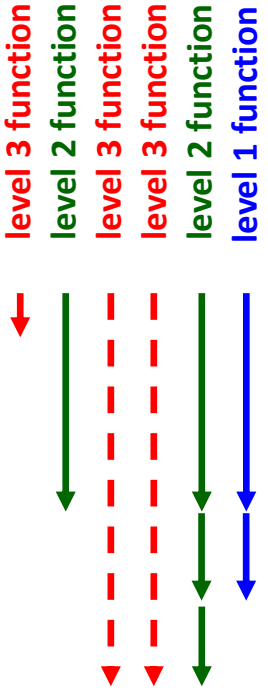
main.py

```
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=3

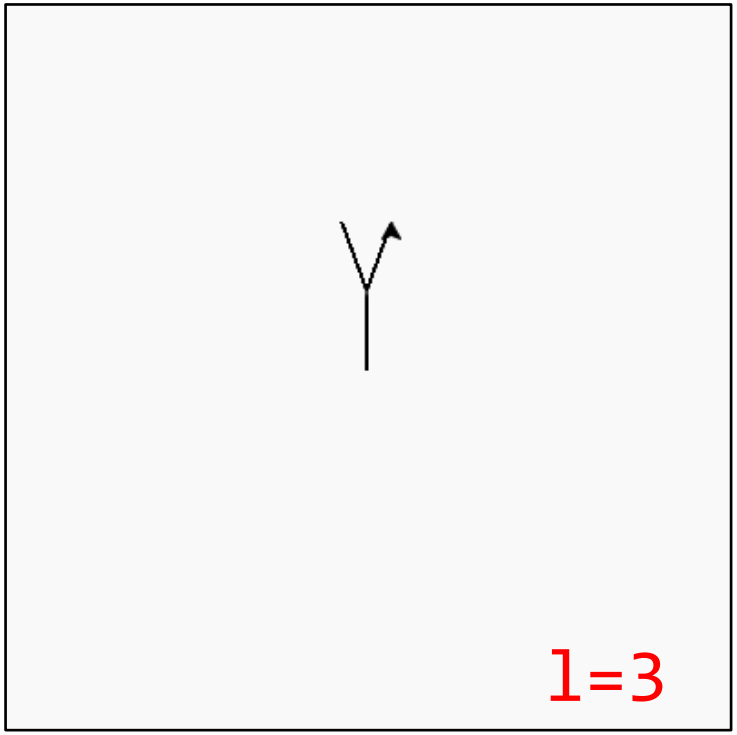
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

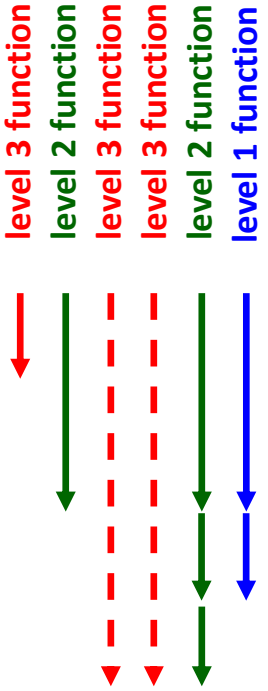
main.py

```
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=3

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

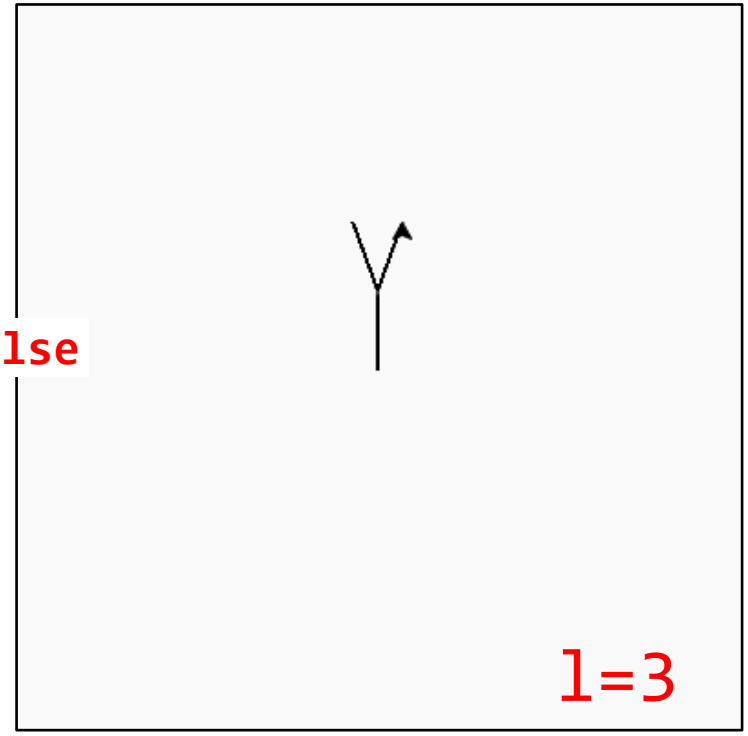


trinket ▶ Run ? Modules

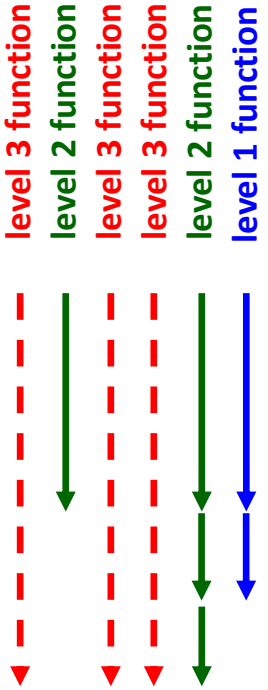
main.py

```
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)
```

False



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

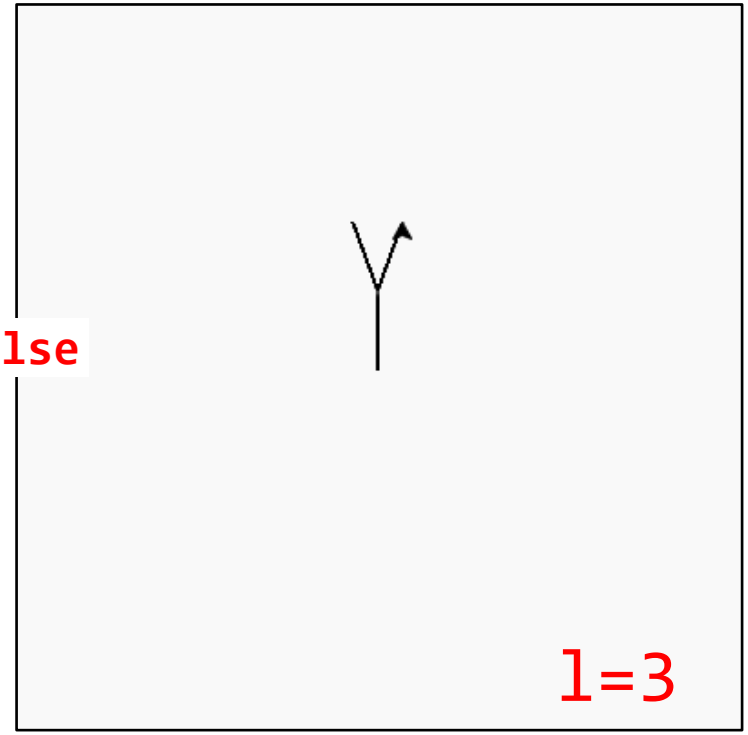


trinket ▶ Run ? Modules

main.py

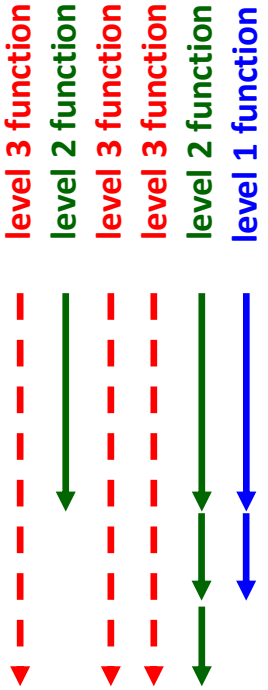
```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38   if l <= level_limit:
39     t.fd(d)
40     t.lt(20)
41     fwdtrn(d*0.9, l+1)
42     t.rt(40)
43     fwdtrn(d*0.9, l+1)
44     t.lt(20)
45     t.bk(d)
46
47 fwdtrn(trunk,1)
```

False



Complete level 3 fn

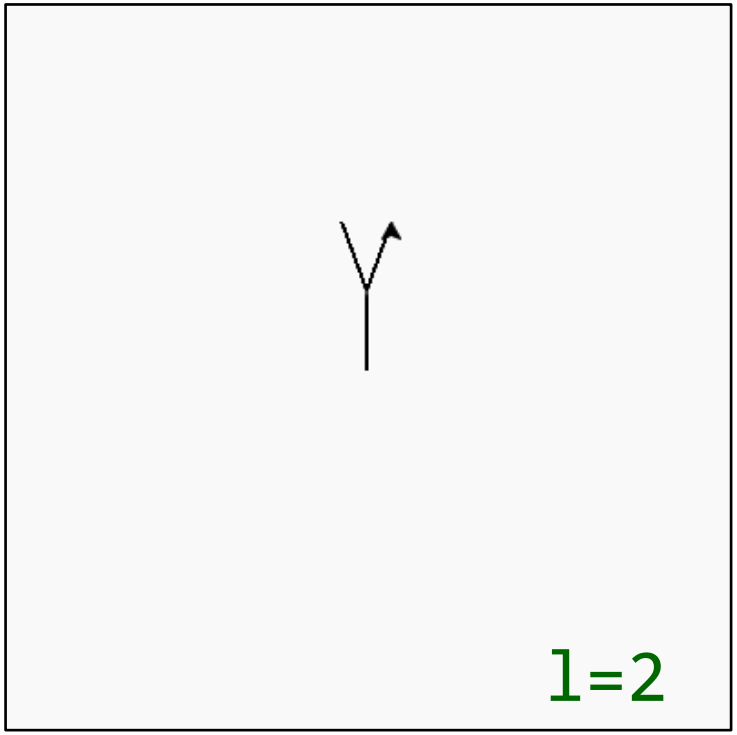
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

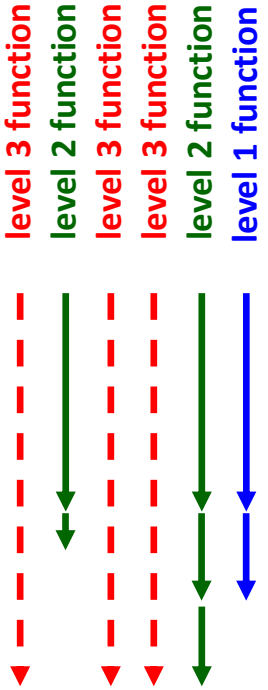
main.py

```
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)
```



Continue unfinished level 2 fn

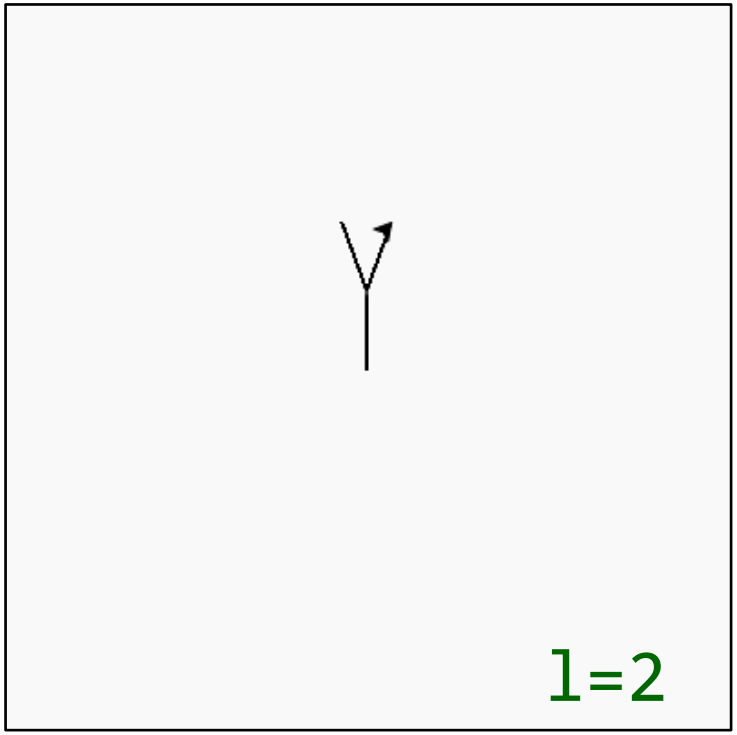
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



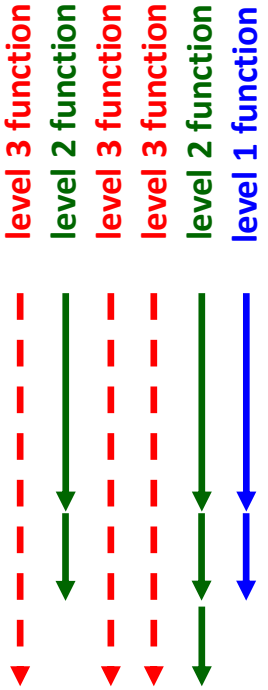
trinket ▶ Run ? Modules

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)
```



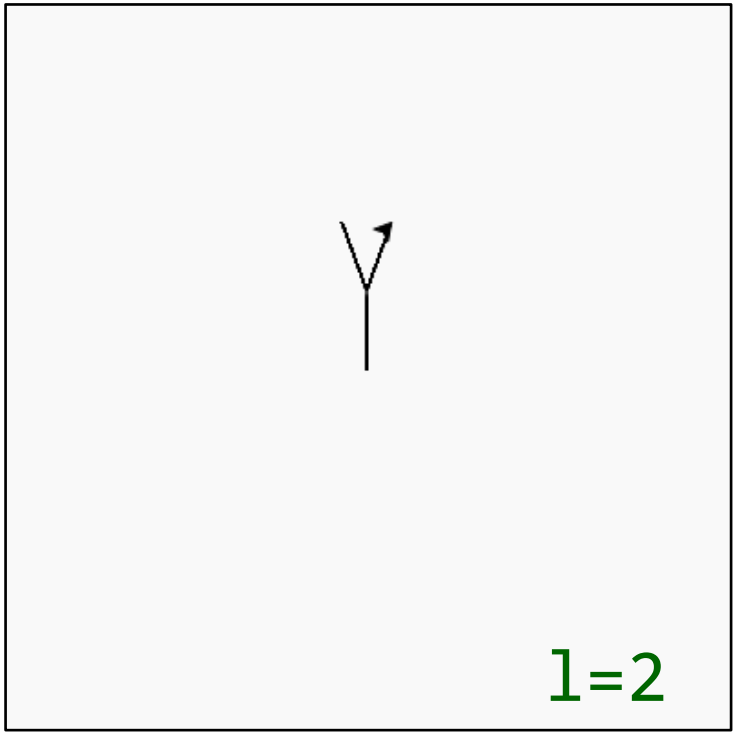
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

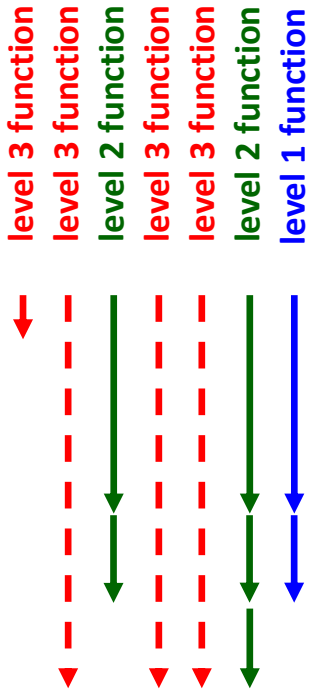
main.py

```
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=3

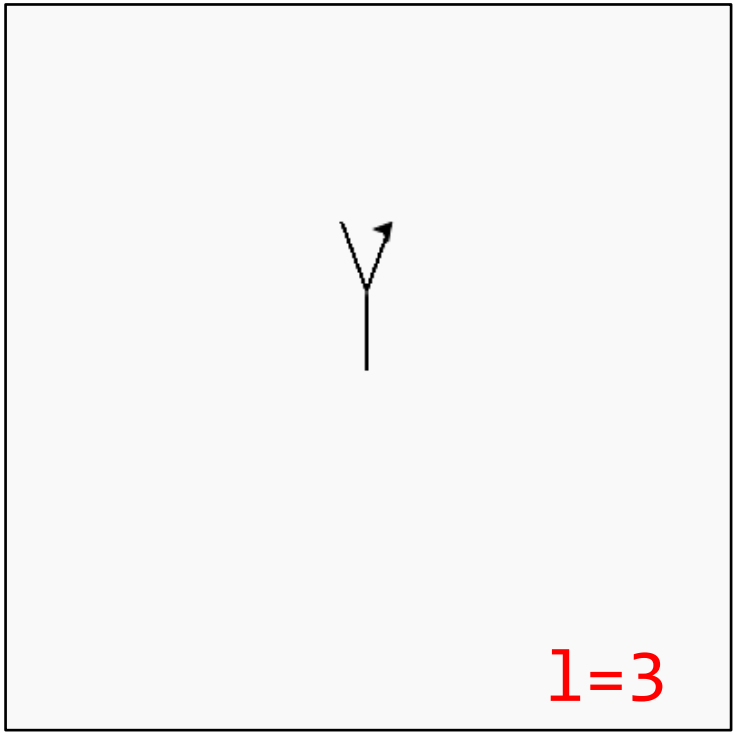
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

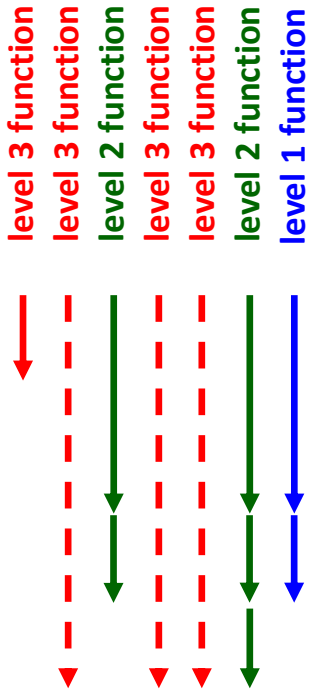
main.py

```
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=3

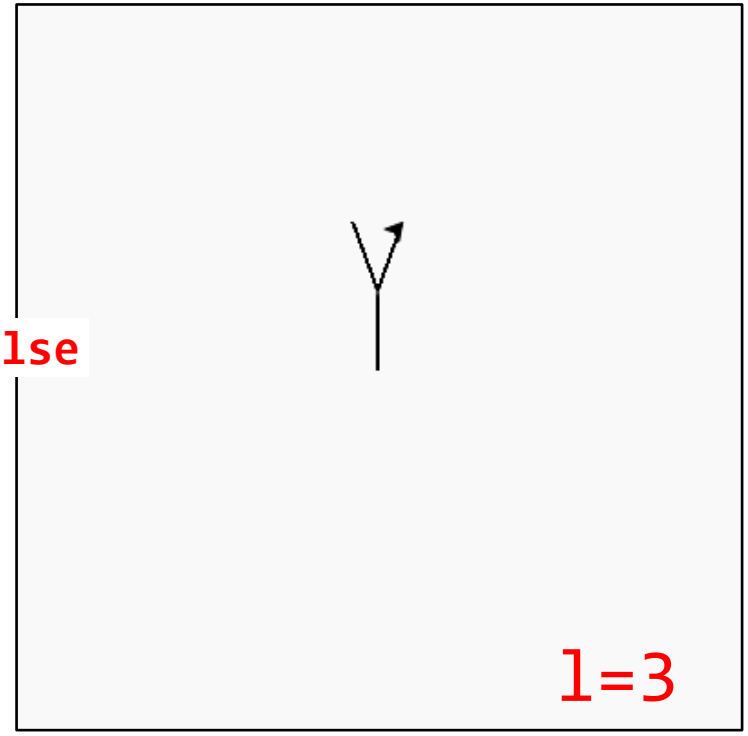
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

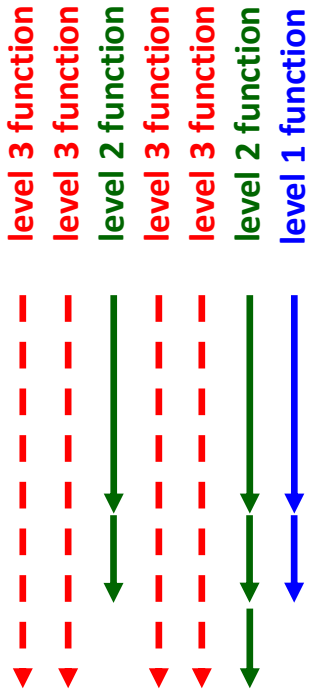
main.py

```
32
33 trunk = 40
34 level_limit = 2
35
36 # Recursive function
37 def fwdtrn(d,l):
38   if l <= level_limit: 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

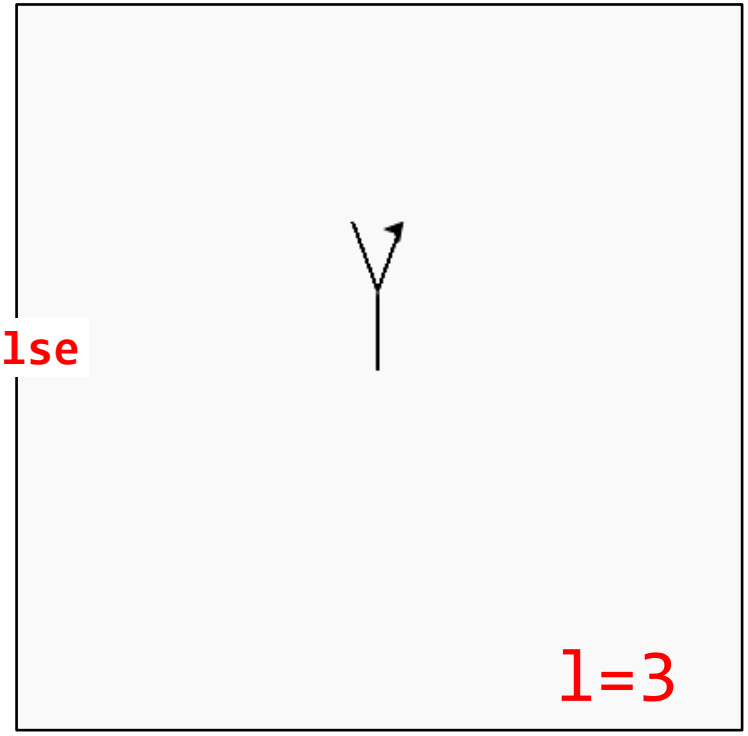
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

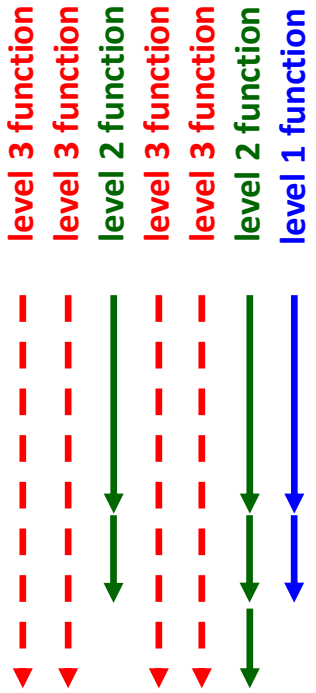
main.py

```
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)
```



Complete level 3 fn

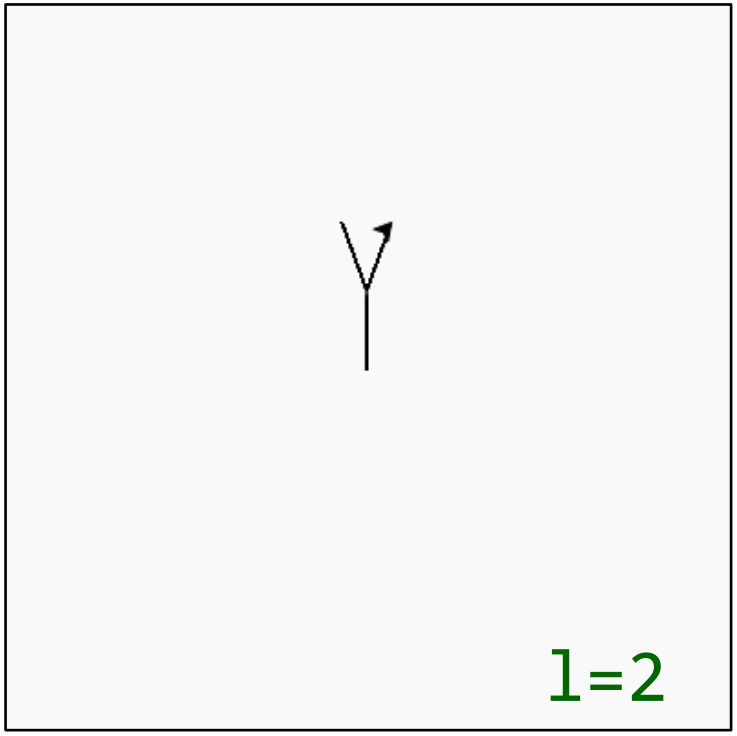
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

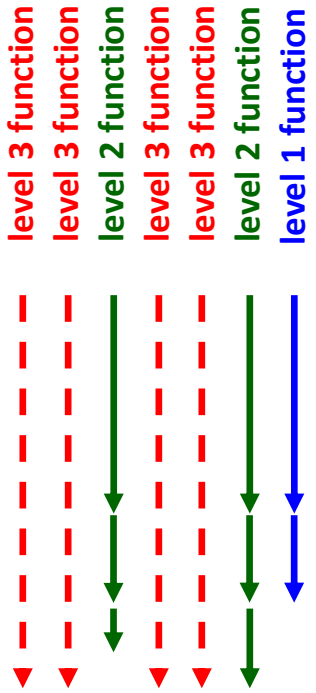
main.py

```
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)
```



Continue unfinished level 2 fn

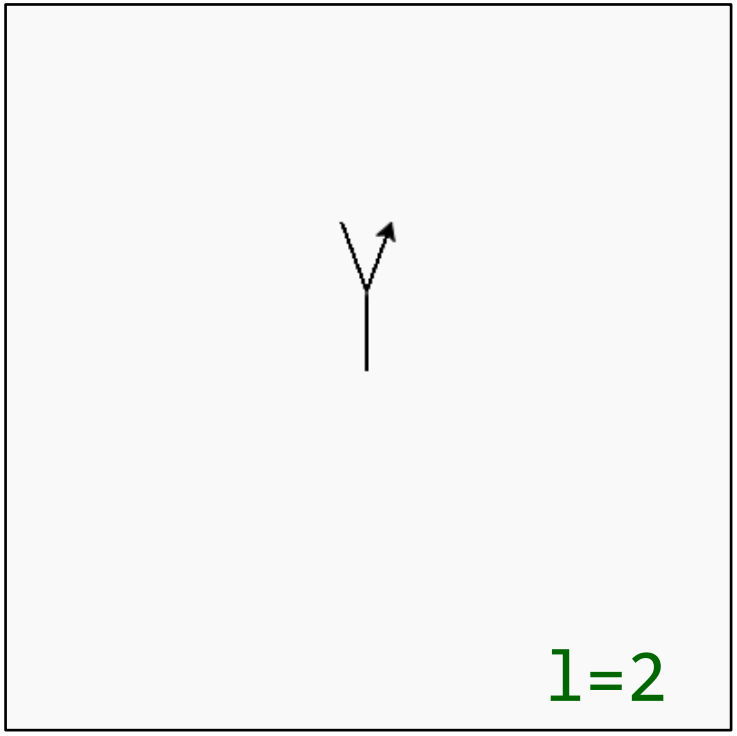
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



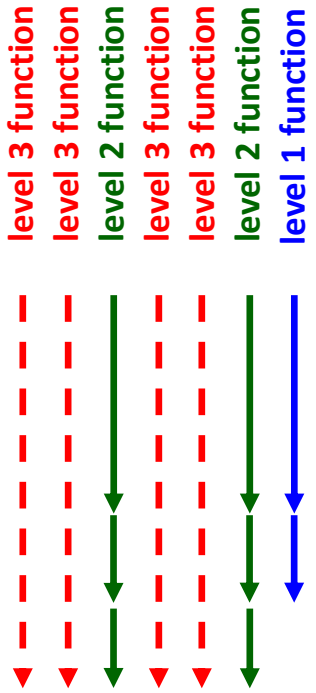
trinket ▶ Run ? Modules

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)
```



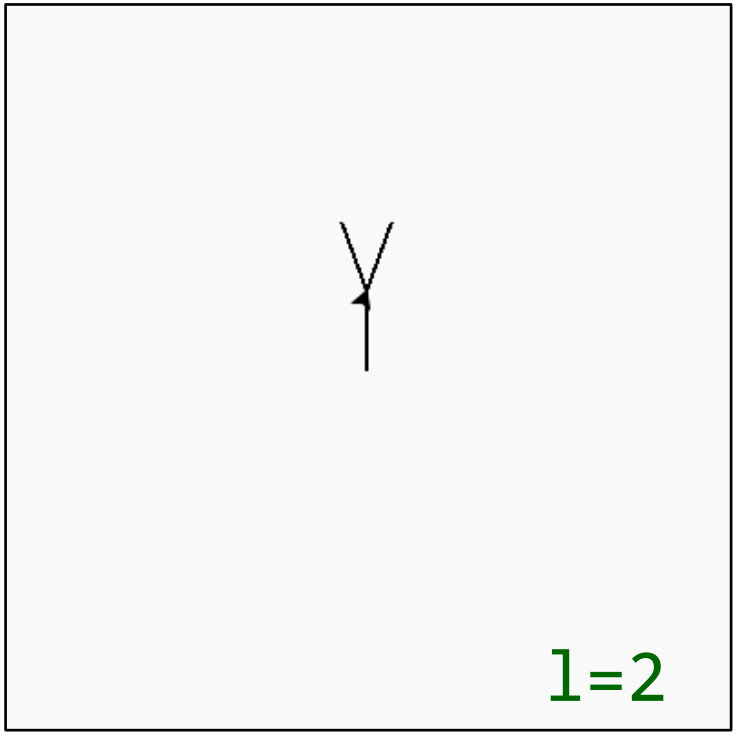
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

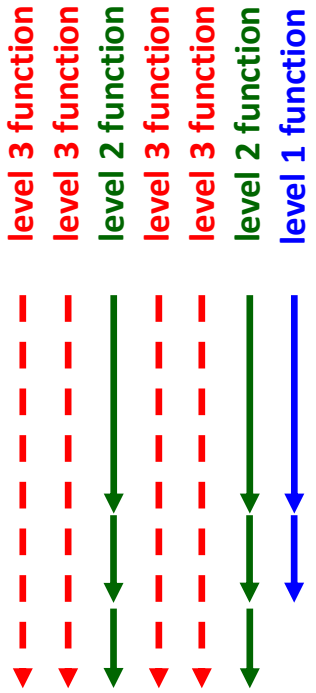
main.py

```
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)
```



Complete level 2 fn

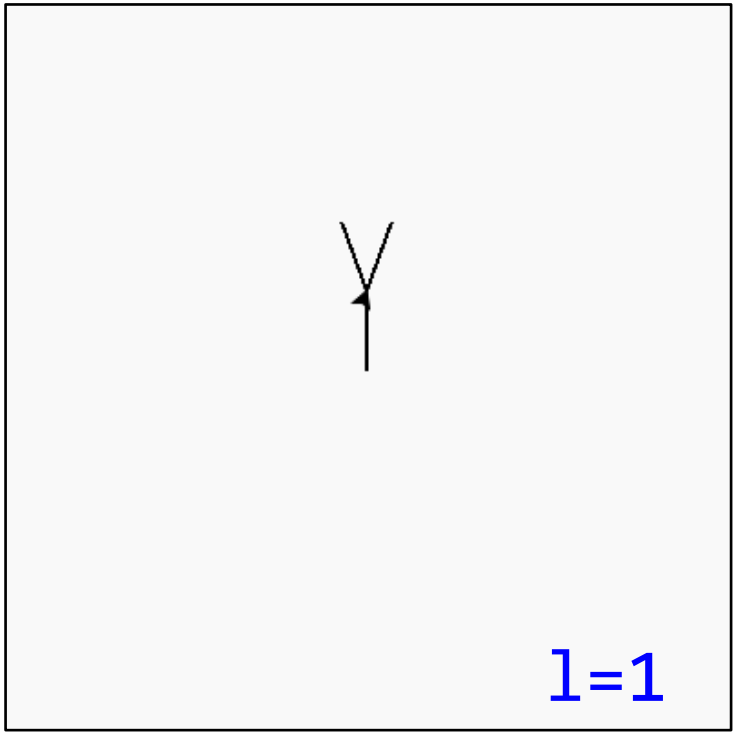
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



trinket ▶ Run ? Modules

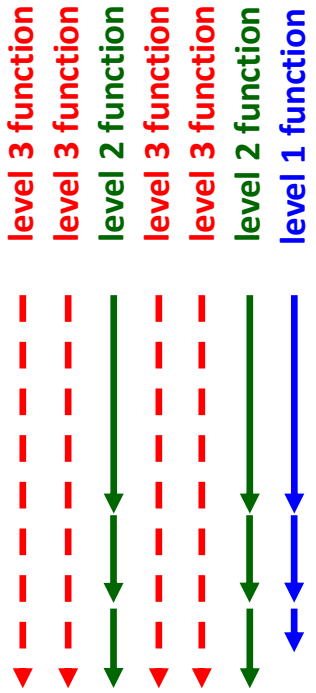
main.py

```
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)
```



Continue unfinished level 1 fn

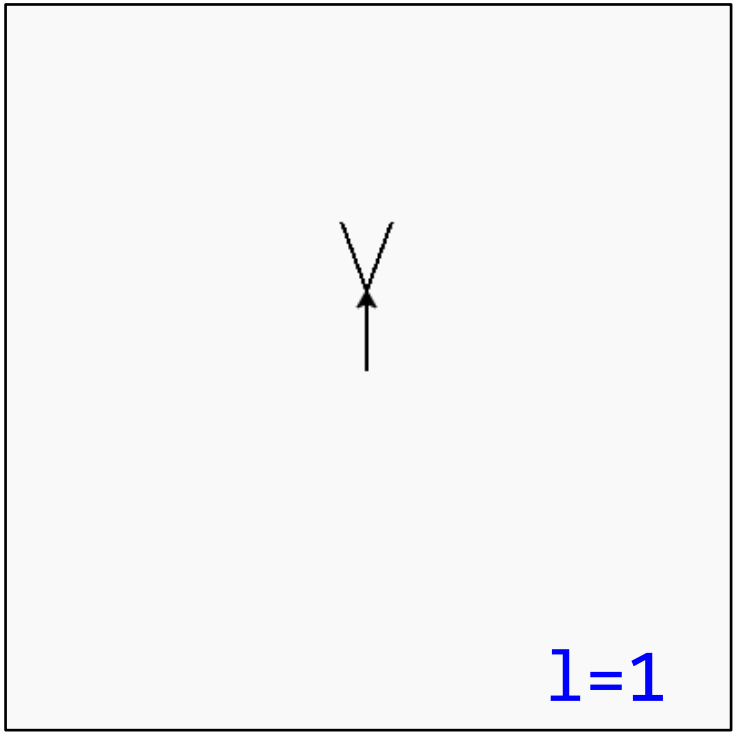
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



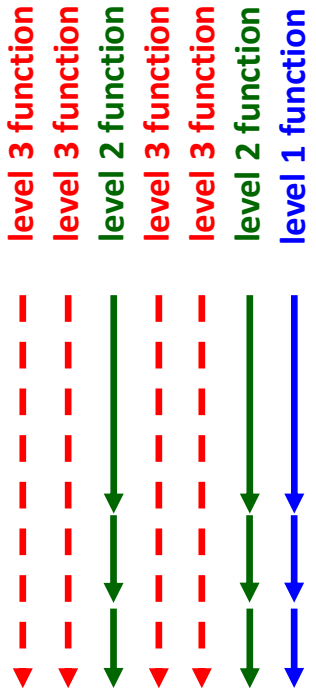
trinket ▶ Run ? Modules

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)
```



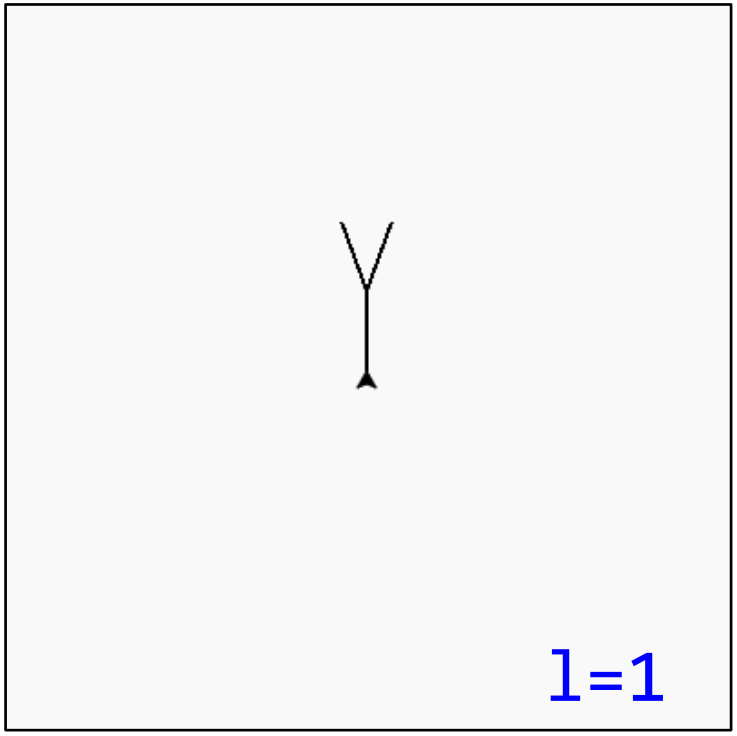
Step-by-step walkthrough of fractal tree code for a tree with number of levels set to 2



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)
```



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.)