

```

import turtle

#input iterations and size
iteration=int(input('How many iterations?'))
length=int(input('What length for each segment (recommended is
3-10)'))
angle=int(input("What angle do you want the turns? (90 is
standard)?"))

#set counter to 1
count=1

#initialize iteration variables (start with right)
r = 'R'
l = 'L'
prev = r
current = prev

#generate iterations (I figured out the algorithm by manually folding
and looking for patterns.)
#I also looked at many other python codes to check my work since I
could only look at so many folds.
while count<iteration:
    #reverse previous iteration
    prev = prev[::-1]
    #add a right to the end of previous iteration
    current = prev + r
    #change right to left and left to right (this happens when you
fold)
    for char in range(0,len(prev)):
        if prev[char] == r:
            prev = (prev[:char]) + (l) + (prev[char+1:])
        elif prev[char] == l:
            prev = (prev[:char]) + (r) + (prev[char+1:])
    #add the current to the previous iteration
    current = (current) + (prev)
    #print R/L
    print("Iteration ", count, ":")
    print(current)
    #make the current iteration the previous and increase the counter
    prev = current
    count=count+1

turtle.ht()
turtle.speed(0)
turtle.forward(length)

#go through each peak/valley in turtle
for char in range(0,len(current)):

```

```
#if right:
if current[char] == (r):
    turtle.right(angle)
    turtle.forward(length)
#if left
elif current[char] == (l):
    turtle.left(angle)
    turtle.forward(length)
```