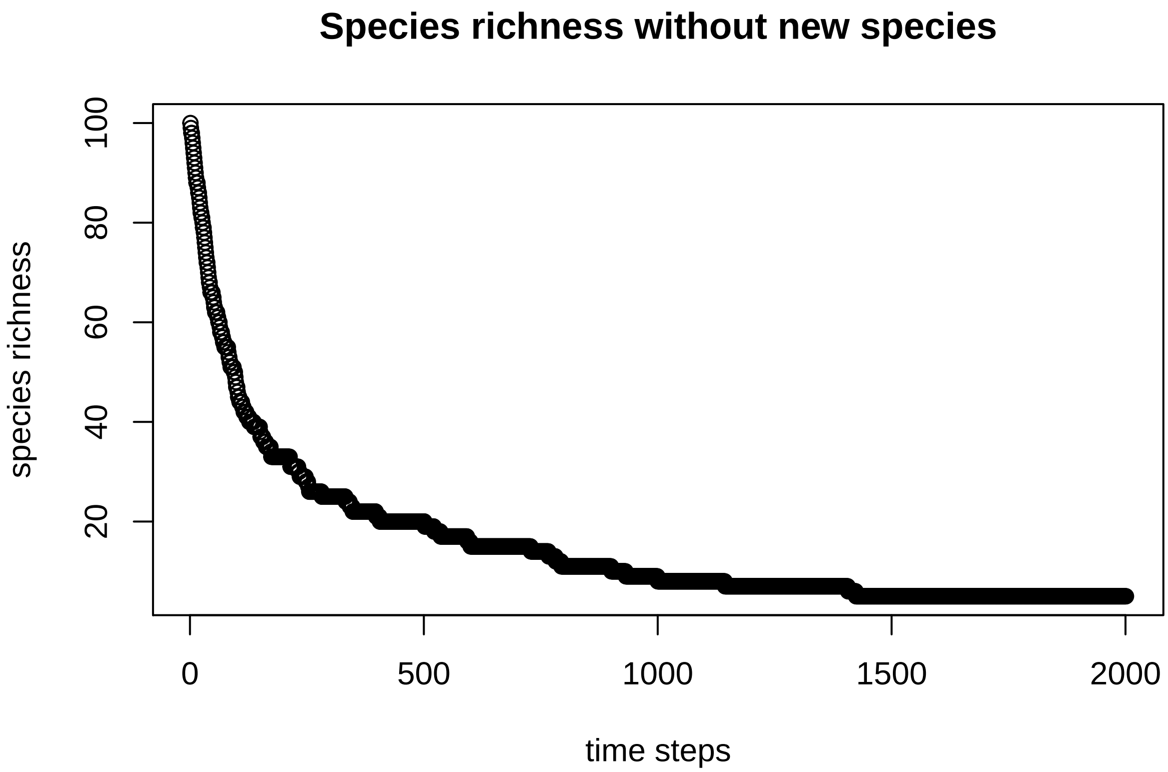
HPC Coursework

PetraGuy

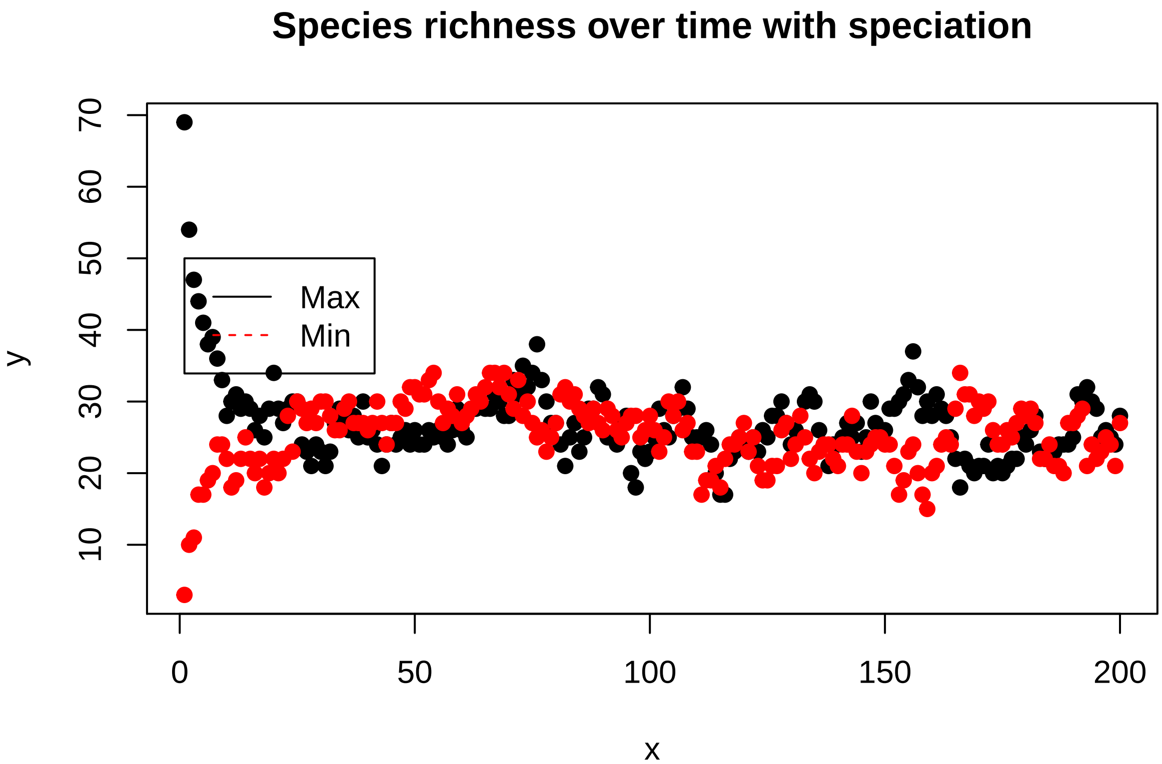
4 December 2017

```r
question_8()
```

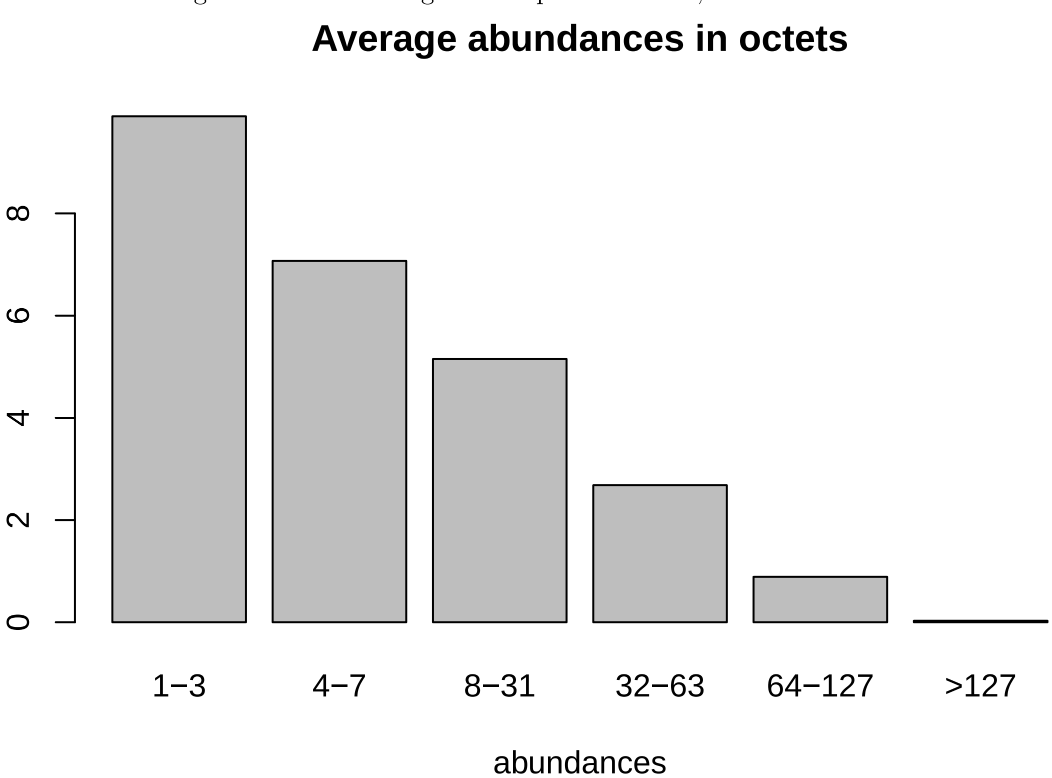


The graph plots species richness of community starting with maximum diversity of 100 species. The speciation rate is 0.1. In this model, new species are not introduces,therefore each step either maintains or reduces richness, so eventually the richness will decrease. For example, if composition was initally 1,2,3,4,5, the next step must reduce richness to for example, 2,2,3,4,5. Another step will either maintain this, e.g. 5,2,3,4,5, or reduce it, 2,2,3,3,5. There is not a mechanism in neutral step for increasing richness and therefore over many steps it will approach 1.

```r
question_12()
```

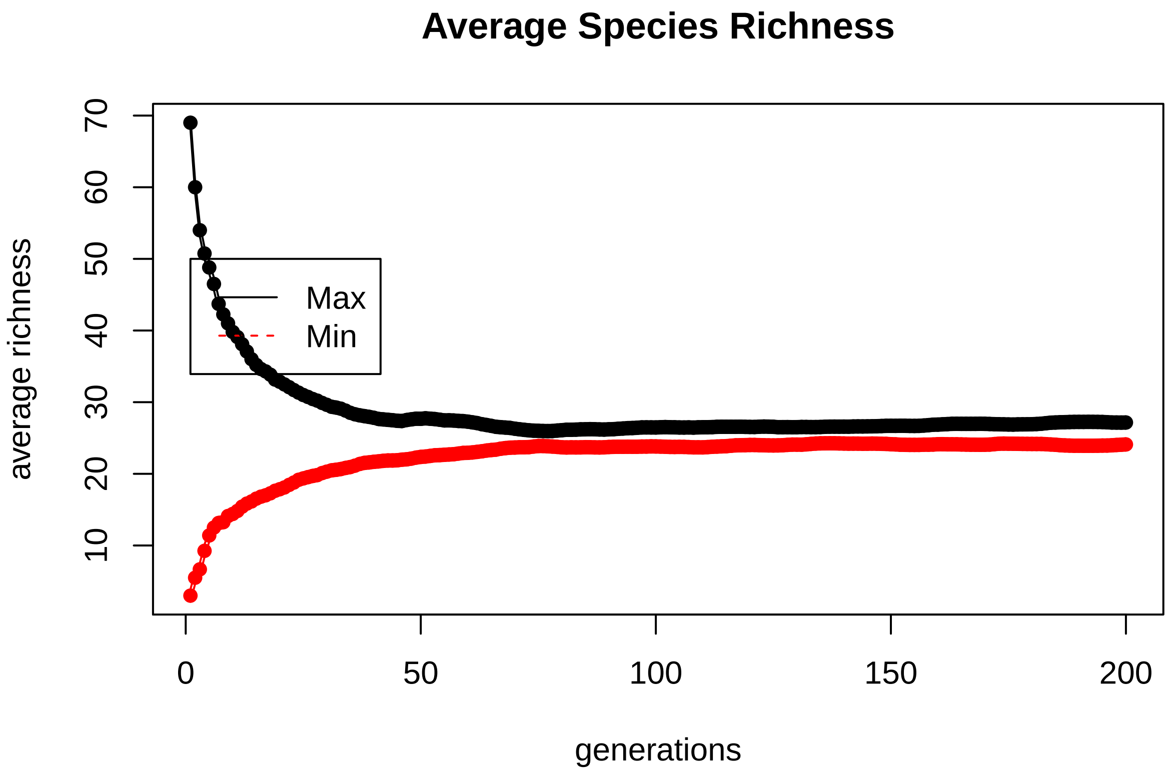


Question 12. Species richness as a time series over 200 generations. Speciation rate is 0.1 and maximal and minimal communities of 100 species were used. The speciation rate causes both communities to approach the same value after sufficient generations. In this simulation a species richness of approximately 30 is reached after around 50 generations. The higher the speciation rate, the richer the final community and vice versa.



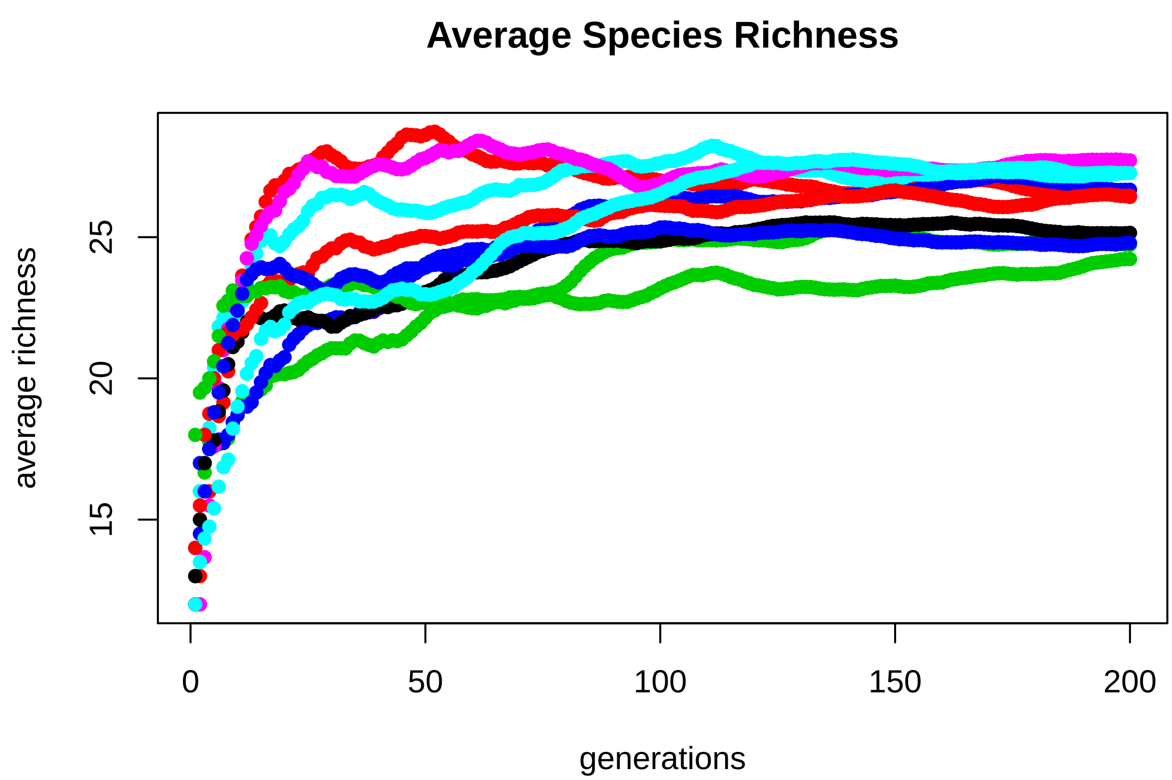
The graph shows the average abundances of the community after the burn-in of 200 generations. The distribution of abundances would change if the speciation rate changed. A higher speciation rate would result in a richer community and therefore the first octet would have a higher frequency. A low speciation rate results in a community of low richness and therefore the frequencies of the larger octets increase at the expense of the smaller.

```
challenge_A()
```



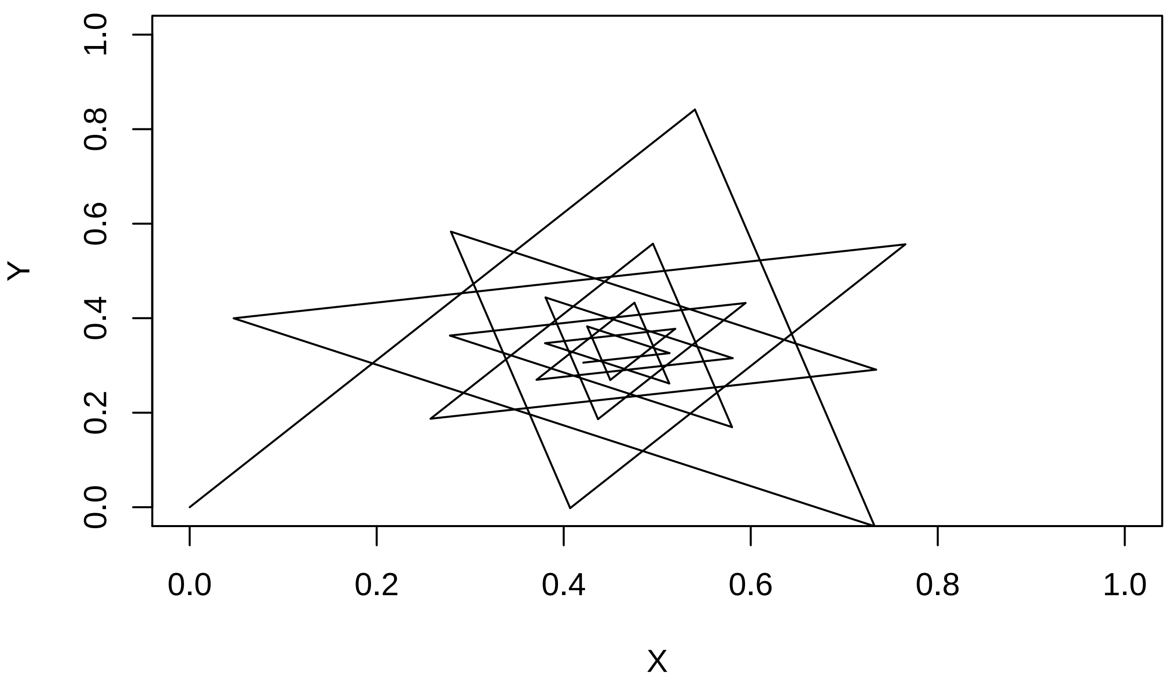
The graph is a plot of average richness values as the times increments. The confidence inervals were calcualted using the variance of the values from 50 to 100 increments. If the entire series was used, the confidence interval would be larger.

```
challenge_B()
```




```
chaos_game()
```

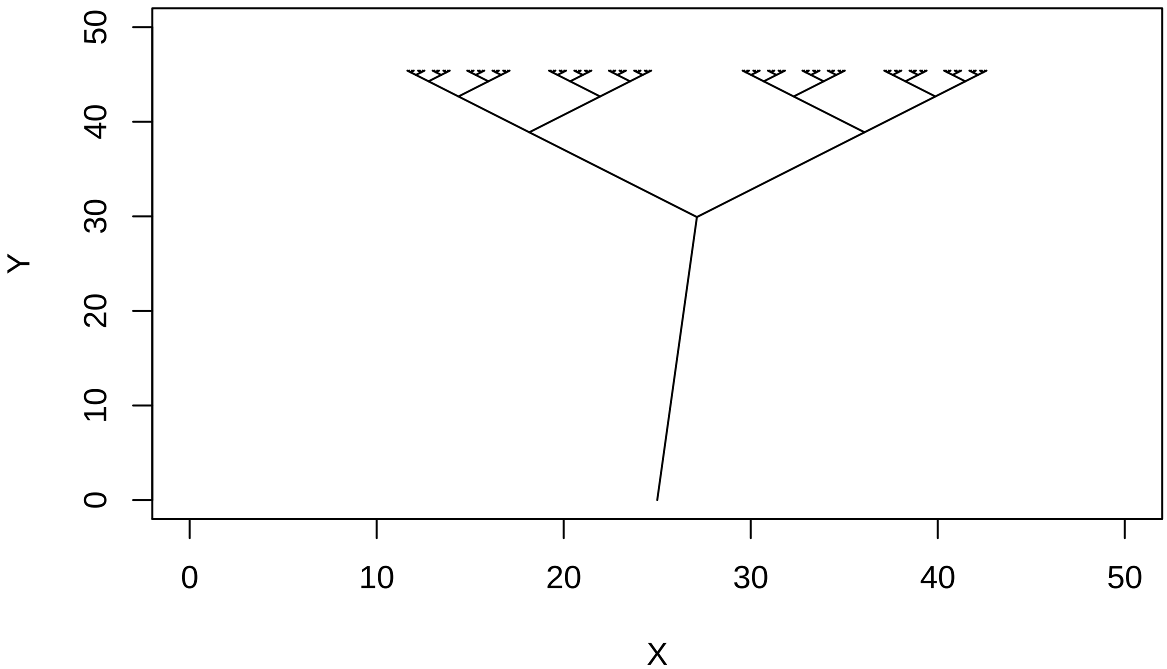
The code produces a Sierpinski Gasket type picture.



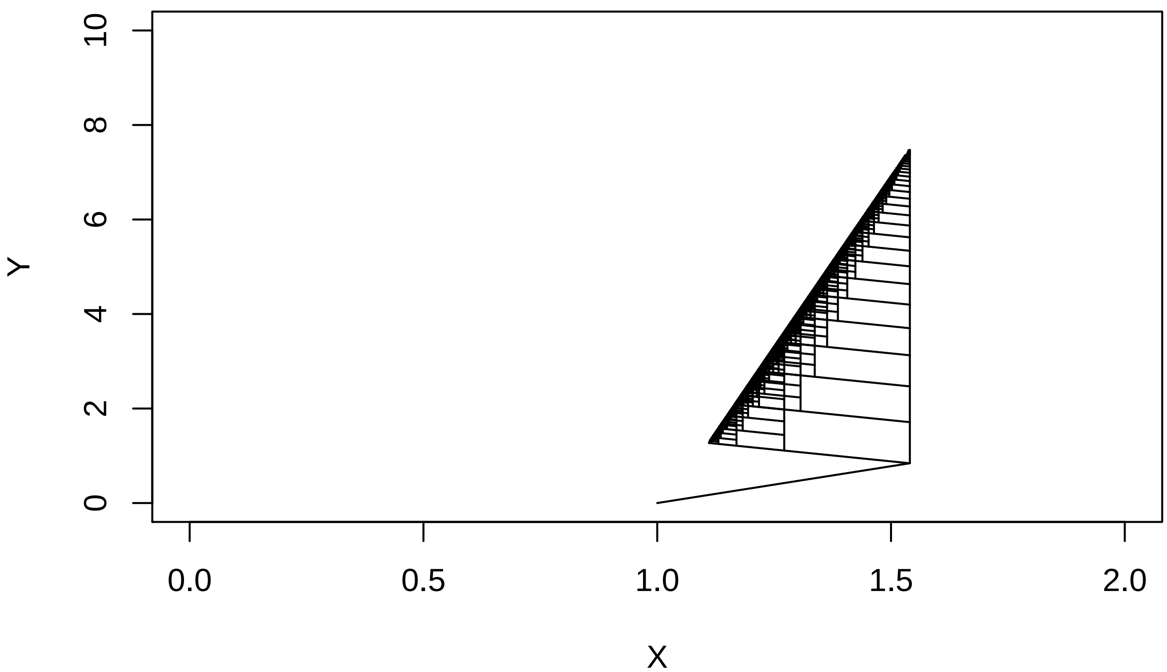
Q22

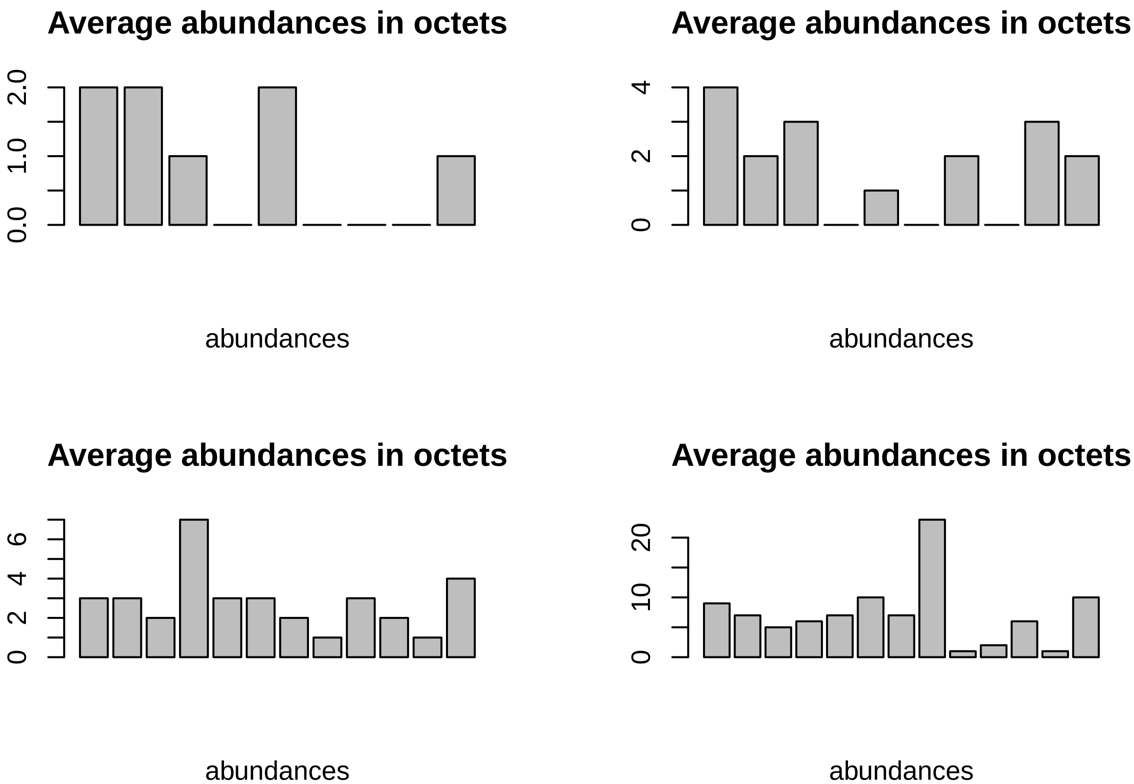
Fractals. The code produces a spiral, but includes a nested, recurssive loop because it continually calls itself. Putting an if (distance > 0.1) statement before calling spiral within spiral ensure that the programme will stop.

```
plot(NA, xlim=c(0,50), ylim=c(0,50), xlab="X", ylab="Y")
tree(c(25,0),30,1.5)
```

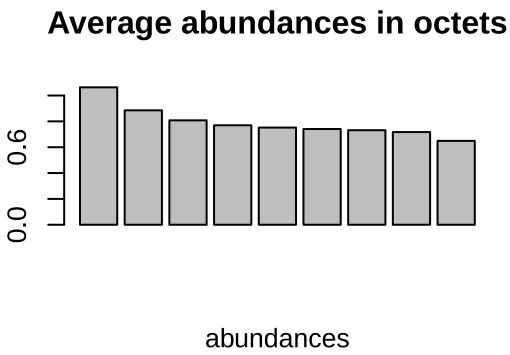


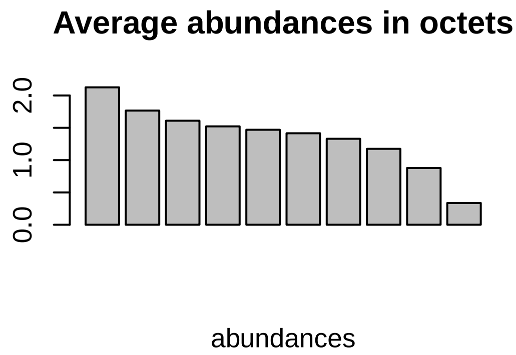

```
plot(NA, xlim=c(0,2), ylim=c(0,10), xlab="X", ylab="Y")
fern(c(1,0),1,1)
```

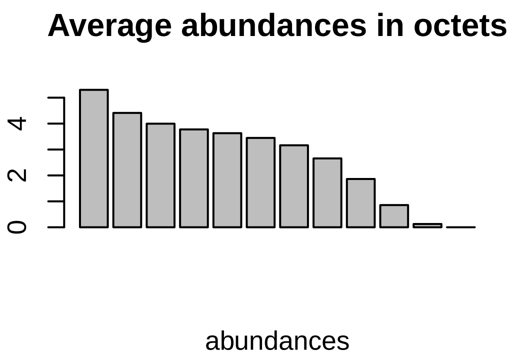


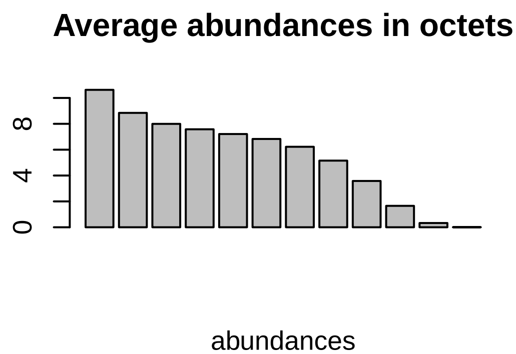


These are the octets from the coalescence simulation, which look very similar to the results of the simulation from the HPC (!). Not totally sure why one is quiker - except that in the coalescence model you generate one community, then do the size of the community calculations, so for our sizes,that's a maximum of 10,000 cyles in the loop. For the neutral model you generate a new community of size N many times during the burn in, then you continue recreating new communities for the presest time in order to take an average over all the cyles. Therefore there are many more calculations.


Average abundances in octets
0.0
0.6
abundances


Average abundances in octets
0.0
1.0
2.0
abundances


Average abundances in octets
0
2
4
abundances


Average abundances in octets
0
4
8
abundances