

Lecture 27 | Student class symbol table

```
1 class Student:
2     ① def __init__(self, name, year):
3         self.name = name
4         self.year = year
5
6     def getName(self):
7         return self.name
8
9     ② def getYear(self):
10        return self.year
11
12    ③ def setYear(self, newYear):
13        self.year = newYear
14
15
16 if __name__ == '__main__':
17     # Symbol table example
18     ① jane = Student('Jane', 2024)
19     ② print(jane.getYear())
20     ③ jane.setYear(2023)
```

line 13: 2023
self.year = newYear

Set year

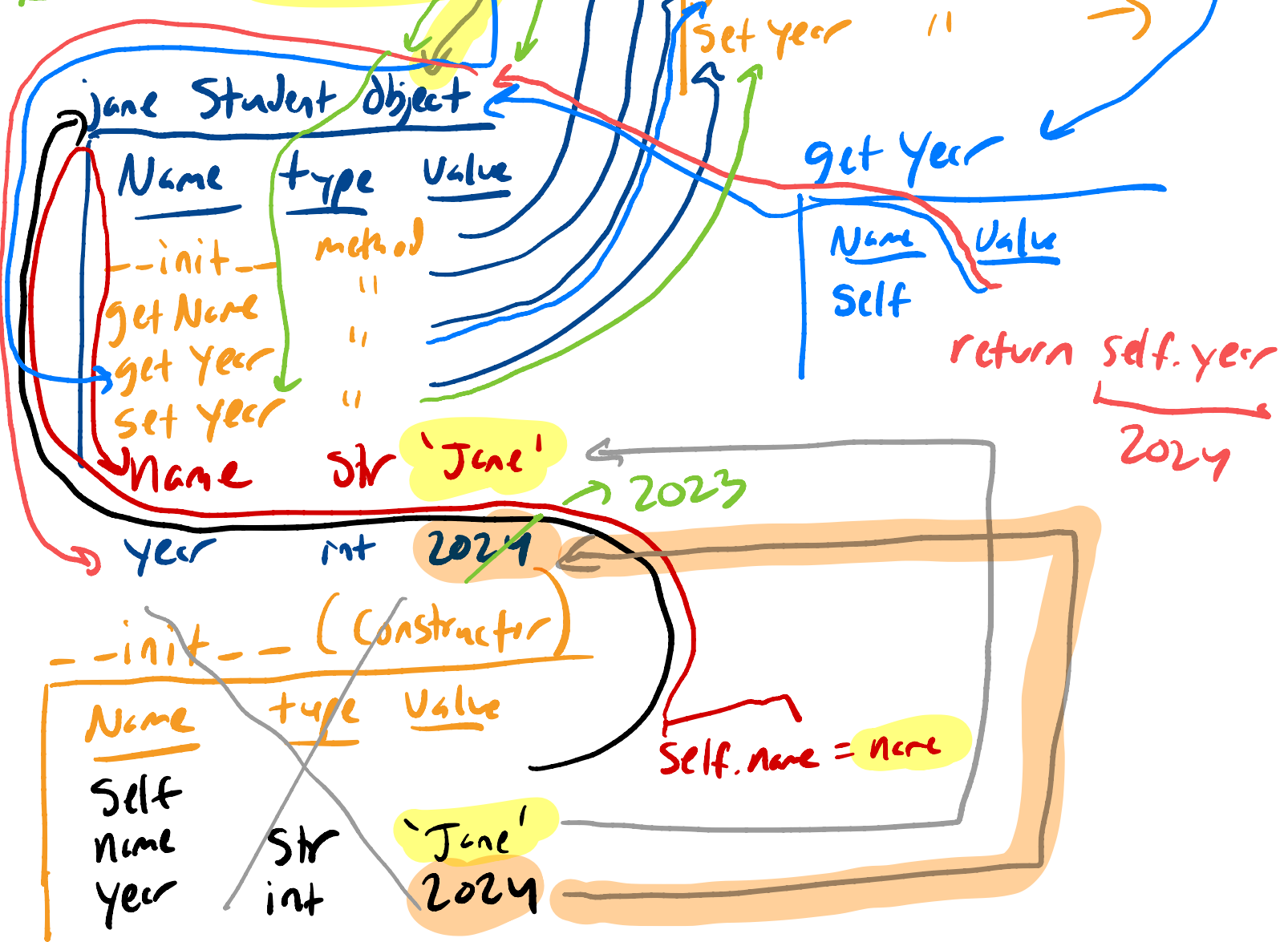
<u>Name</u>	<u>Value</u>
self	
newYear	2023

Main

<u>Name</u>	<u>Type</u>	<u>Value</u>
Student	class	
jane	Student object	

Student class

<u>Name</u>	<u>Type</u>	<u>Value</u>
--init--	method	→
getName	method	→
get Year	"	→
set year	"	→



① jane = Student ('Jane', 2024)

line 3: Self.name = name] in the constructor
 ↑ hop to a different symbol table

line 4: self.year = year

⇒ After line 4 ⇒ `--init--` symbol table goes away

go to line 1:

jane = Student('Jane', 2024)

Jane variable created in main symbol table

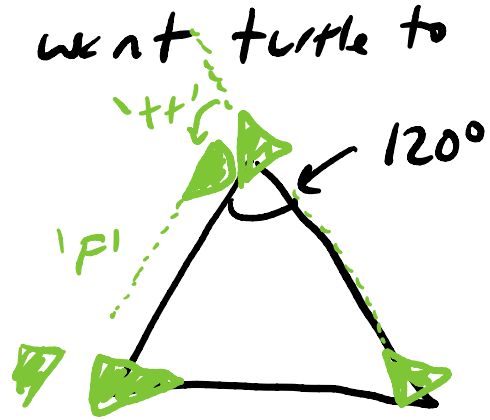
line 19: jane.getYear() } start in main code
2024

line 20: jane.setYear(2023) } start in main code

L-systems

Koch Snowflake

initial shape: 1st shape we want turtle to draw



Represent initial shape using an L-system String:

'F++F++F'



Characters we know how to interpret to make turtle move:

- 'F' means move turtle forward by distance d
- '-' means turn turtle right by angle θ
- '+' means turn turtle left by angle θ

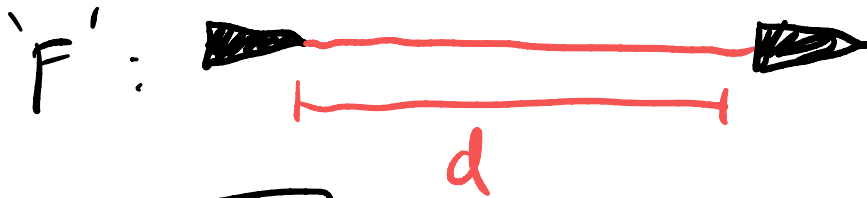
L-system alphabet: all chars we know how to interpret

We must set:

- distance d } up to us
- angle θ } based on what shape we want to make

for Koch snowflake (triangle): 60°

initial
Shape



before



$\theta = 60^\circ$

After



60° right turn



turn left 60°