Iteration 1:

Algoritme:

Start

Define 3 points with random X and Y value

Draw triangle from points

Make a object with random X and Y value inside the triangle

Draw the object

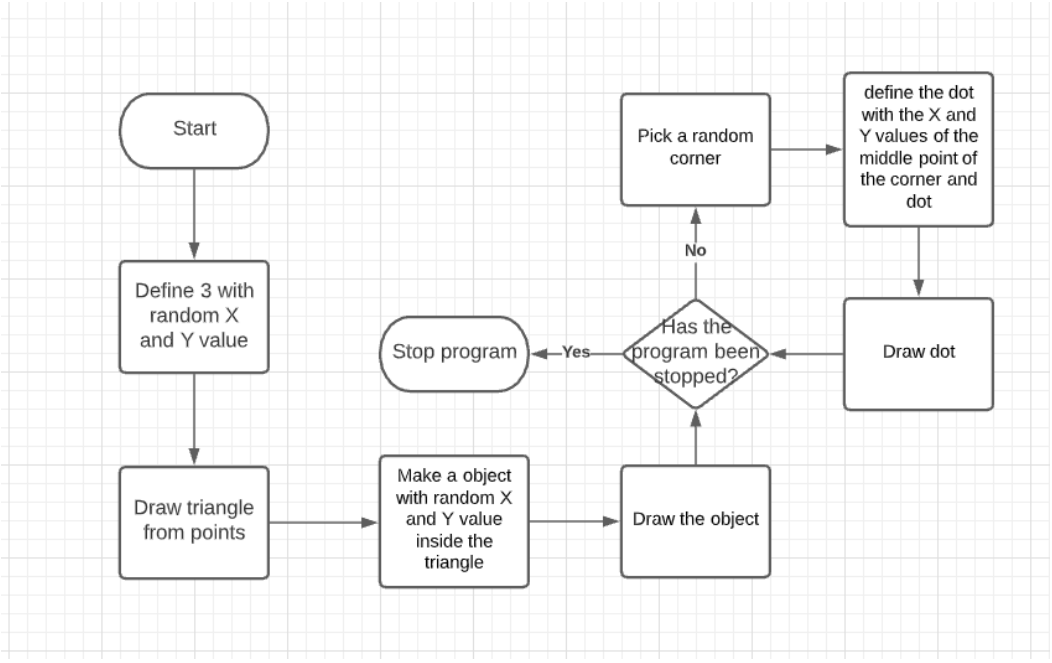
While the game hasnt been stopped

    Pick a random corner

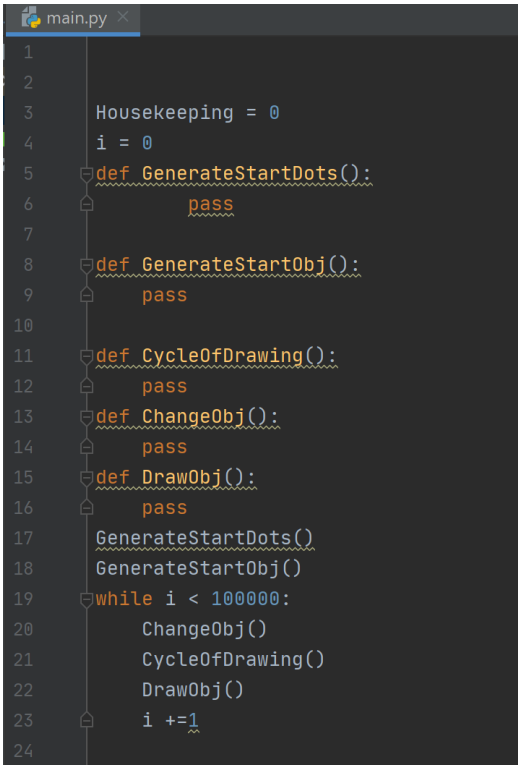
    define the dot with the X and Y values of the middle point of the corner and dot

    Draw dot

Flowchart:

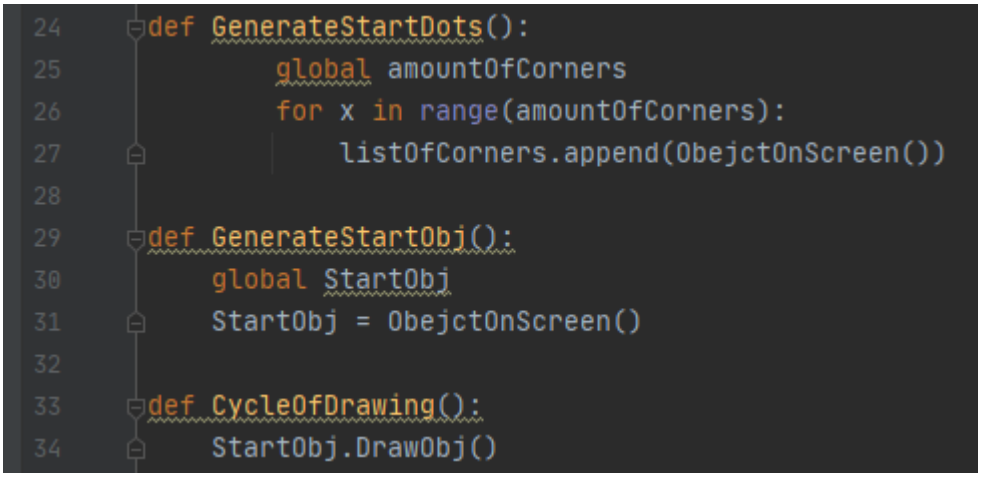


Kodeskelet:



Iteration 2:

Jeg startet med at fylde klassen for Objects on screen ud. Der skulle ikke meget til da der kun skulle være tilfældige kordinater og en tom draw funktion for nu.
Derefter udfyldte jeg alle de funktioner der skulle køres, alle ud over change har ikke ændret sig siden dette skridt.



Derefter blev Turtle funktioneliet tilføjet. Det var ret simpelt da jeg fulgte tutoriallen på https://realpython.com/beginners-guide-python-turtle/. Vi starter koden med at oprette turtle, og viduet den tegner i

```
s = turtle.getscreen()
t = turtle.Turtle()
t.hideturtle()
t.speed(100000)
turtle.screensize(0,800)
```

.

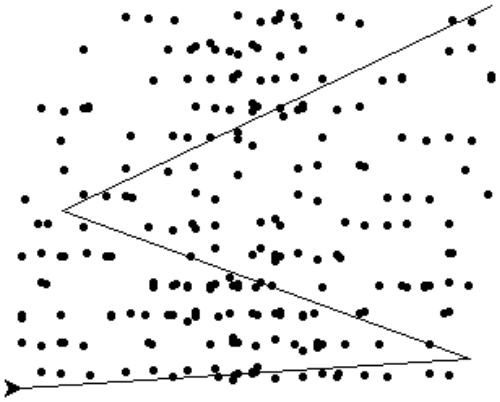
Så blev der tilføjet draw funktionen for både prikker og trekanten. Den tager to runder rundt i trekanten så trekanten bliver lukket af og der ikke sidder en side åben.

```
class ObejctOnScreen:

    def __init__(self):
        self.x = random.randint(0,400)
        self.y = random.randint(0,400)
    def DrawObj(self):
        t.goto(self.x,self.y)
        t.dot(3)
```

```
c = 0
for corner in listOfCorners * 2:
    if c == 0:
        t.penup()
    else:
        t.pendown()
    t.goto(corner.x,corner.y)
    c += 1
t.penup()
```

Derefter blev change funktionen lavet. Der var mange variationer der ikke virket. Jeg viser kun den mest spændene. Den første variation som tegnet noget på skærmen, fik det til at ligne den tegnede tilfældigt. Dette var fordi den tog det abolutte midtpunkt af de to punkter. Det skabte problemer både fordi det var nu floats og ikke ints. Og fordi midtpunktet af skærmen er 0,0. Gjorde det at prikkernu fik de forkert kordinater.



Change funktionen var det sidste der manglede og virke og endte med at se sådan ud.

```
def ChangeObj():
    RandomCorner = random.randint(0,2)
    ChosenCorner = listOfCorners[RandomCorner]
    StartObj.x = (StartObj.x + ChosenCorner.x) / 2
    StartObj.y = (StartObj.y + ChosenCorner.y) / 2

```