

```
1 import pygame
2 from random import *
3
4 breite = 500
5 hoehe = 433 # ungefähr sqrt(3)/2 * breite; sqrt =
6 square root = Quadratwurzel
7 iterationen = 100000
8 weiss = (255, 255, 255) # Ein Tupel aus drei Zahlen, kodiert die
9 # Farbe Weiss im RGB-Modell, das wir später kennenlernen werden.
10 # Koordinaten des Punkts A
11 xA = 0
12 yA = hoehe
13
14 # Koordinaten des Punkts B
15 xB = breite
16 yB = hoehe
17
18 # Koordinaten des Punkts C
19 xC = int(breite / 2) # Der Befehl "int" macht eine Kommazahl (=
20 # reelle Zahl = real number) durch Abschneiden der Nachkommastellen zu einer
21 # ganzen Zahl (= integer).
22 yC = 0
23
24 pygame.init()
25 pygame.display.set_caption("Das Chaos-Spiel")
26
27 leinwand = pygame.display.set_mode((breite + 1, hoehe + 1))
28
29 pygame.draw.line(leinwand, weiss, (xA, yA), (xB, yB))
30 pygame.draw.line(leinwand, weiss, (xB, yB), (xC, yC))
31 pygame.draw.line(leinwand, weiss, (xC, yC), (xA, yA))
32
33 pygame.display.update()
34
35 x = randint(0, breite)
36 y = randint(0, hoehe)
37
38 for i in range(0, iterationen):
39     zufall = randint(1, 3)
40     if zufall == 1:
41         x = (x + xA) / 2
42         y = (y + yA) / 2
43     if zufall == 2:
44         x = (x + xB) / 2
45         y = (y + yB) / 2
46     if zufall == 3:
47         x = (x + xC) / 2
48         y = (y + yC) / 2
49     leinwand.set_at((round(x), round(y)), weiss) # Zeichnet ein
50     # Pixel am Punkt (x,y) bzw. genauer an den gerundeten Koordinaten.
51     pygame.display.update((round(x), round(y), 1, 1))
52     # denn das ist deutlich schneller als
53     # pygame.display.update()
54
55     if i % 10000 == 0: # "Prozent-
56         # Division" liefert den Rest der Division.
57         print(i)
58
59 print("Fertig!") # Ausgabe der
60 # Zeichenkette (= string) zwischen den Anführungszeichen.
61 print(f"Ich habe {iterationen} Punkte gezeichnet.") # f-string =
62 # formatted string
63
64 pygame.time.delay(2000)
65 pygame.quit()
```