Stackenlichten

LED-Spielereien für Anfänger mit Neopixeln und fadecandy-Board

@kellertuer

MetaMeute, Chaotikum e.V.

MetaNooK, Lübeck, 10. November 2017

Moin.

NeoPixel & fadecandy – Technik

Stackenlichten – Praxis

Let’s blink already! – Demonstration

Ziele

....meines Projektes

- Retro-Pixelschubsen
- ...mit fraktalem/abstraktem Licht
- (ein bischen) Hardwarebasteln

Inspiration I: Magic Shifter
magicshifter.net

....des Vortrags

- Spielen mit LEDs macht Spaß
- Einstieg ist sehr einfach
- wenig Hardware- und Software-Wissen notwendig (für den Einstieg)

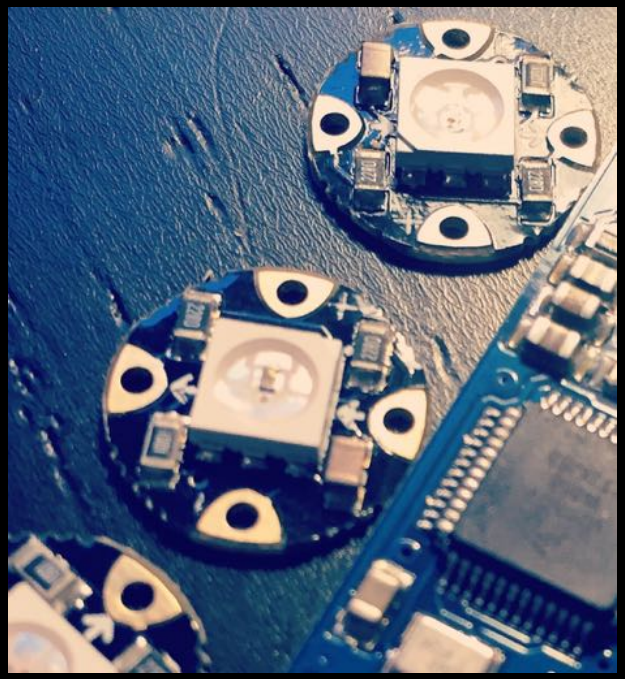
NeoPixel, Fadecandy & Co

Flora NeoPixel

- WS2845 LED von adafruit
  learn.adafruit.com/adafruit-neopixel-uberguide
- Anschlüsse: +5V, Gnd, Data In, Data Out
- selfclock data, 24 bit pro LED, MSB first
- jede LED „zweigt“ einmal 24 Bit ab.

⇒ jede einzeln ansteuerbar

- offenes Datenblatt
- Microcontroller als Steuergerät
  (Arduino, letztes Jahr: Ivo, Nils)



Flora RGB NeoPixel

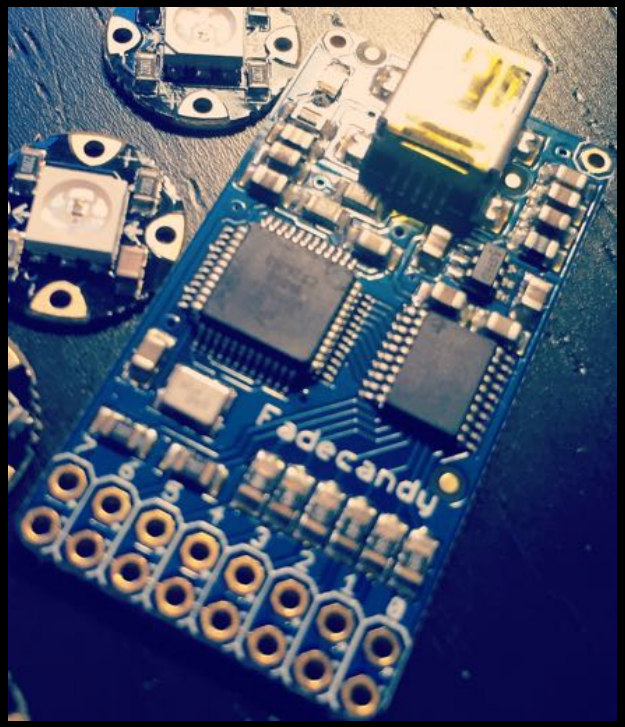
auch als Streifen, Matrix, Ring, Stick, Jewel,

klassische LED, SMT,...

Flora: 1,50€/LED, als Streifen: 0,41 €/LED

fadecandy

- optimiertes Dithering
- pro Pin 64 LEDs, also bis zu 512 LEDs
- einfacher Betrieb: per USB & kleinem Server
- Open Pixel Control (OPC) Schnittstelle
- direkt in Python/Javascript adressierbar
- kein Speicher onBoard, dafür: adafruit Circuit Playground Classic



fadecandy-Board

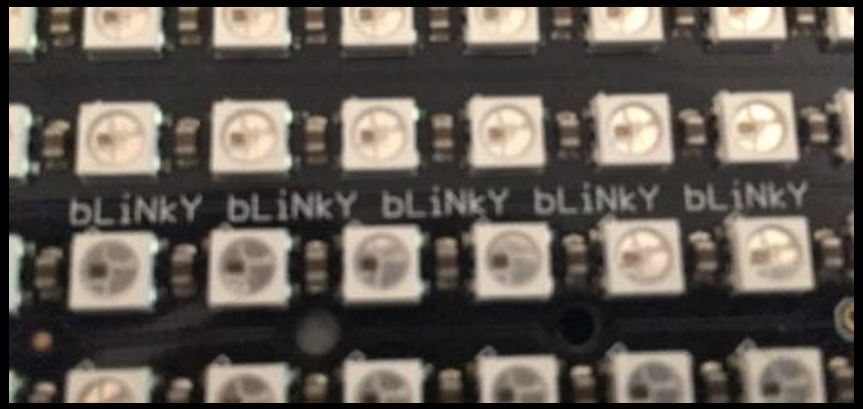
von scanlime und adafruit

25 €

weitere Technik

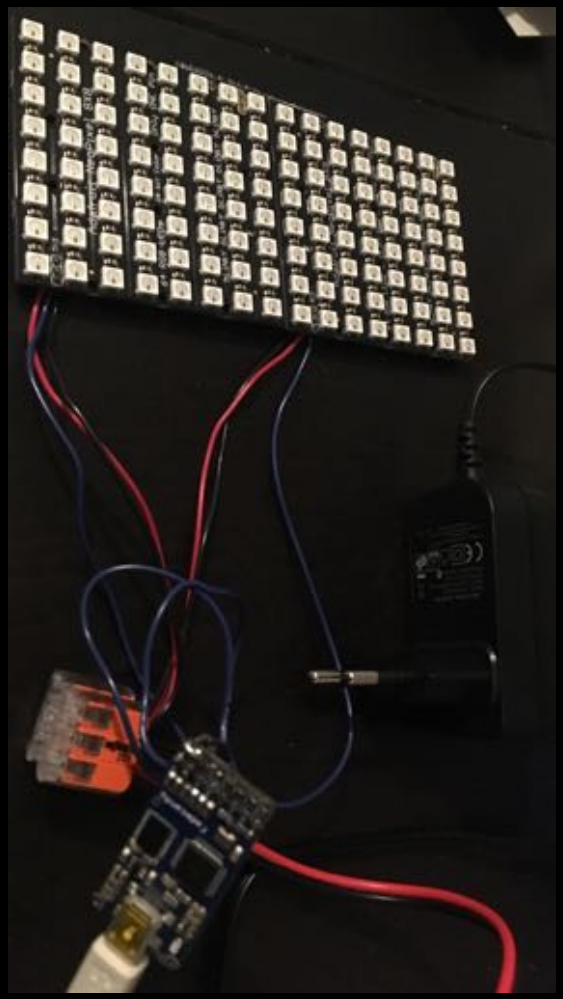
- Netzteil/Batteriefach für die Stromversorgubng
- (fadecandy:) Micro-USB auf USB-Kabel
- (opt.) Kabel verlöten / vernähen
- (opt.) evtl. ein Raspberry Pi für dedizierten `fc-server`

⇒ Go blinky!



Das Einstiegs-Beispiel: NeoPixel Matrix

- 2 Matrizen mit je 8×8, also 128 px insg.
- jeweils einem Port des fadecandy-Boards
- Standard-5V-Netzteil (etwa vom Handy)
  vielleicht nicht für alle LEDs
- Schritte bis zu leuchtenden LEDs
  1. Verkabeln (Kroko-Klemmen statt Löten)
  2. `fc-server` starten
  3. Testseite
  4. Code...



30 € / Matrix (0,46/pixel)

Code-Beispiel: Ein Zufallspixel mit Zufallsfarbe

```
from opc import Client       # OPC Client laden
import time
from numpy.random import randint
c = Client('localhost:7890')  # Verbinden
c.set_interpolation(False);   # direkt Setzen
data = [ [0,0,0] ] * 512      # leeres Bild
i=randint(128)                # i-te LED mit
data[i] = randint(256,size=3) # Zufallsfarbe
print(str(i)+':'+str(data[i]))
c.put_pixels(data)            # Farbe setzen
time.sleep(3)                 # 3 Sek. warten
data[i] = [0]*3
c.put_pixels(data)            # und wieder aus
```

Beispiel: AmpliTie – Tutorial mit Video

CC-BY-SA 3.0 Becky Stern/adafruit

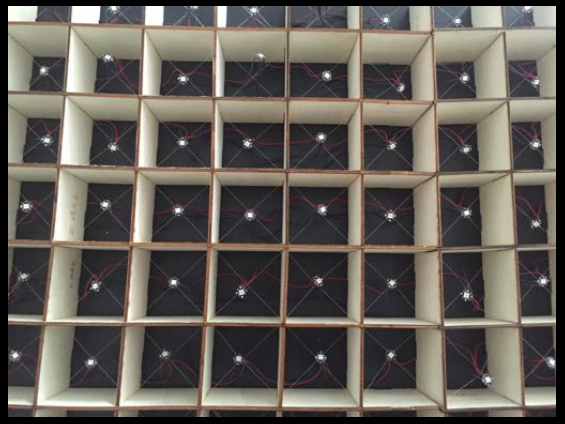
Stackenlichten

Die Anfänge

...oder: warum der Talk nicht schon auf der MetaNook 2016 war...

- „wie Blinkenlichten fürs Wohnzimmer“
- festes 8 × 8cm Gitter, steckbar
- LEDs fest verlötet mit 15cm Kabel dazwischen
- LEDs mit Bindfaden in der Mitte fixiert
- 132,5 × 66,1 × 8 cm (im Zug?!)

- auf dem 33c3: viele solche Systeme (wenn auch selten so groß)



Die Anfänge

...oder: warum der Talk nicht schon auf der MetaNook 2016 war...

- „wie Blinkenlichten fürs Wohnzimmer“
- festes 8×8cm Gitter, steckbar
- LEDs fest verlötet mit 15cm Kabel dazwischen
- LEDs mit Bindfaden in der Mitte fixiert
- $132{,}5 \times 66{,}1 \times 8$ cm (im Zug?!)

- auf dem 33c3: viele solche Systeme (wenn auch selten so groß)

Stackenlichten

- Blinkenlichten (2001, Berlin)

- Stackenblochen (2005)

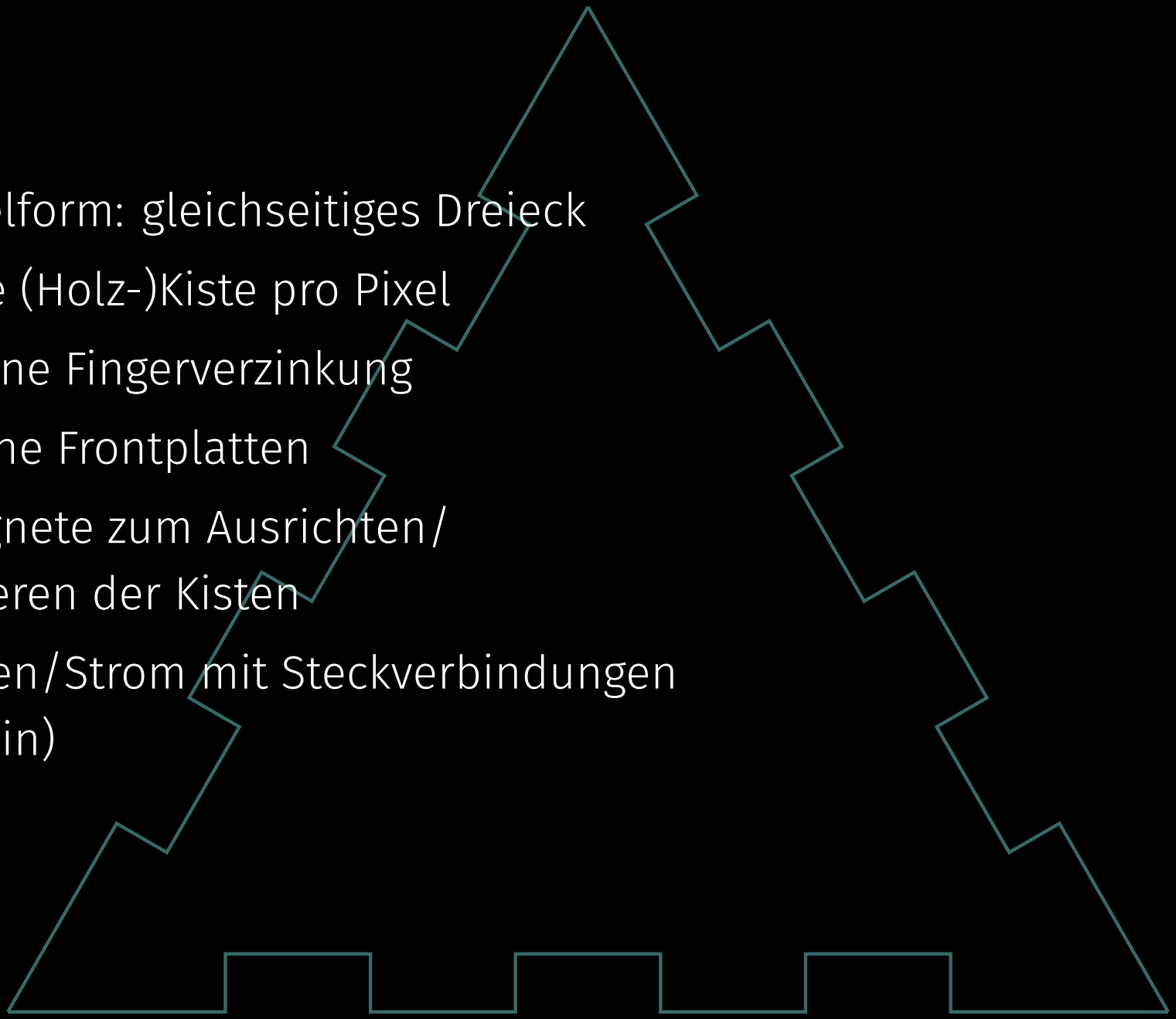
Stackenlichten

- Blinkenlichten (2001, Berlin)

- Stackenblochen (2005)

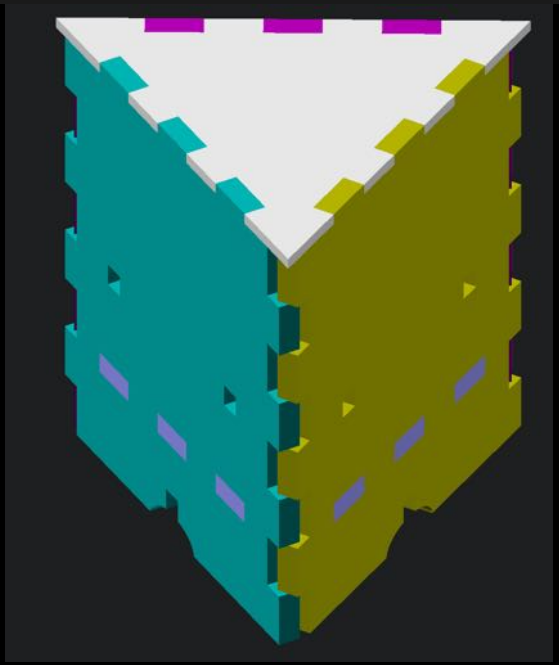
Idee vom 33c3: modulare Dreiecke

- Pixelform: gleichseitiges Dreieck
- Eine (Holz-)Kiste pro Pixel
- offene Fingerverzinkung
- kleine Frontplatten
- Magnete zum Ausrichten/ Fixieren der Kisten
- Daten/Strom mit Steckverbindungen (3-Pin)



Idee vom 33c3: modulare Dreiecke

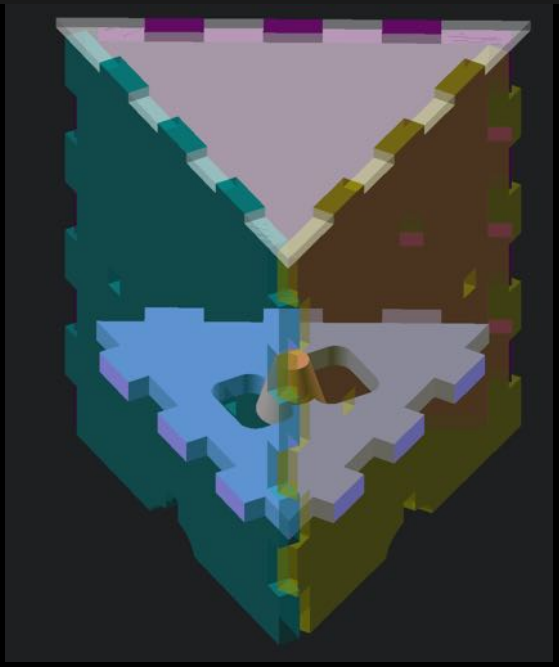
- Pixelform: gleichseitiges Dreieck
- Eine (Holz-)Kiste pro Pixel
- offene Fingerverzinkung
- kleine Frontplatten
- Magnete zum Ausrichten/ Fixieren der Kisten
- Daten/Strom mit Steckverbindungen (3-Pin)



2. Entwurf (openSCAD)

Idee vom 33c3: modulare Dreiecke

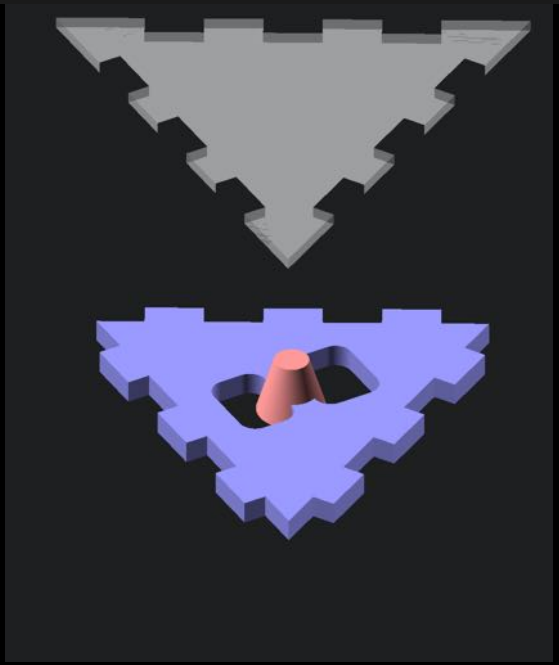
- Pixelform: gleichseitiges Dreieck
- Eine (Holz-)Kiste pro Pixel
- offene Fingerverzinkung
- kleine Frontplatten
- Magnete zum Ausrichten/ Fixieren der Kisten
- Daten/Strom mit Steckverbindungen (3-Pin)



Bodenplatte & „LED-Fuß“

Idee vom 33c3: modulare Dreiecke

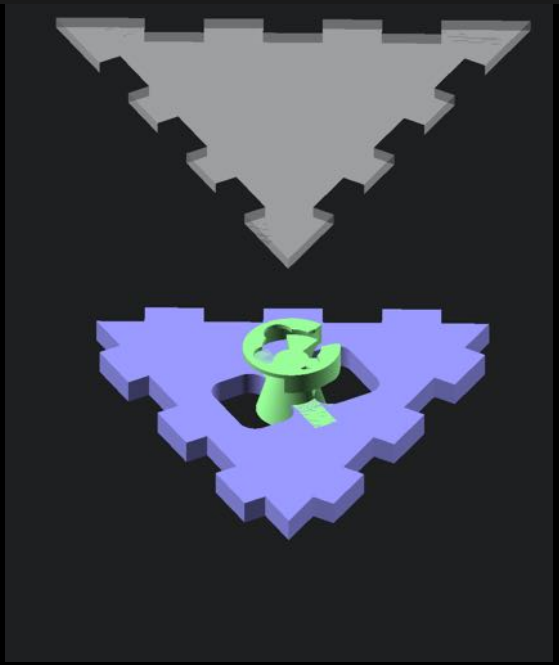
- Pixelform: gleichseitiges Dreieck
- Eine (Holz-)Kiste pro Pixel
- offene Fingerverzinkung
- kleine Frontplatten
- Magnete zum Ausrichten/ Fixieren der Kisten
- Daten/Strom mit Steckverbindungen (3-Pin)



Fuß & LED verklebt

Idee vom 33c3: modulare Dreiecke

- Pixelform: gleichseitiges Dreieck
- Eine (Holz-)Kiste pro Pixel
- offene Fingerverzinkung
- kleine Frontplatten
- Magnete zum Ausrichten/ Fixieren der Kisten
- Daten/Strom mit Steckverbindungen (3-Pin)



Fuß & LED gesteckt

Weitere Iden

- Graph-basierte Pixel-Algorithmen

  (endlich Pixel schubsen!)
- Spiele: Tetris, Robots, Snake,...
- MVC

Eskalationsstufen

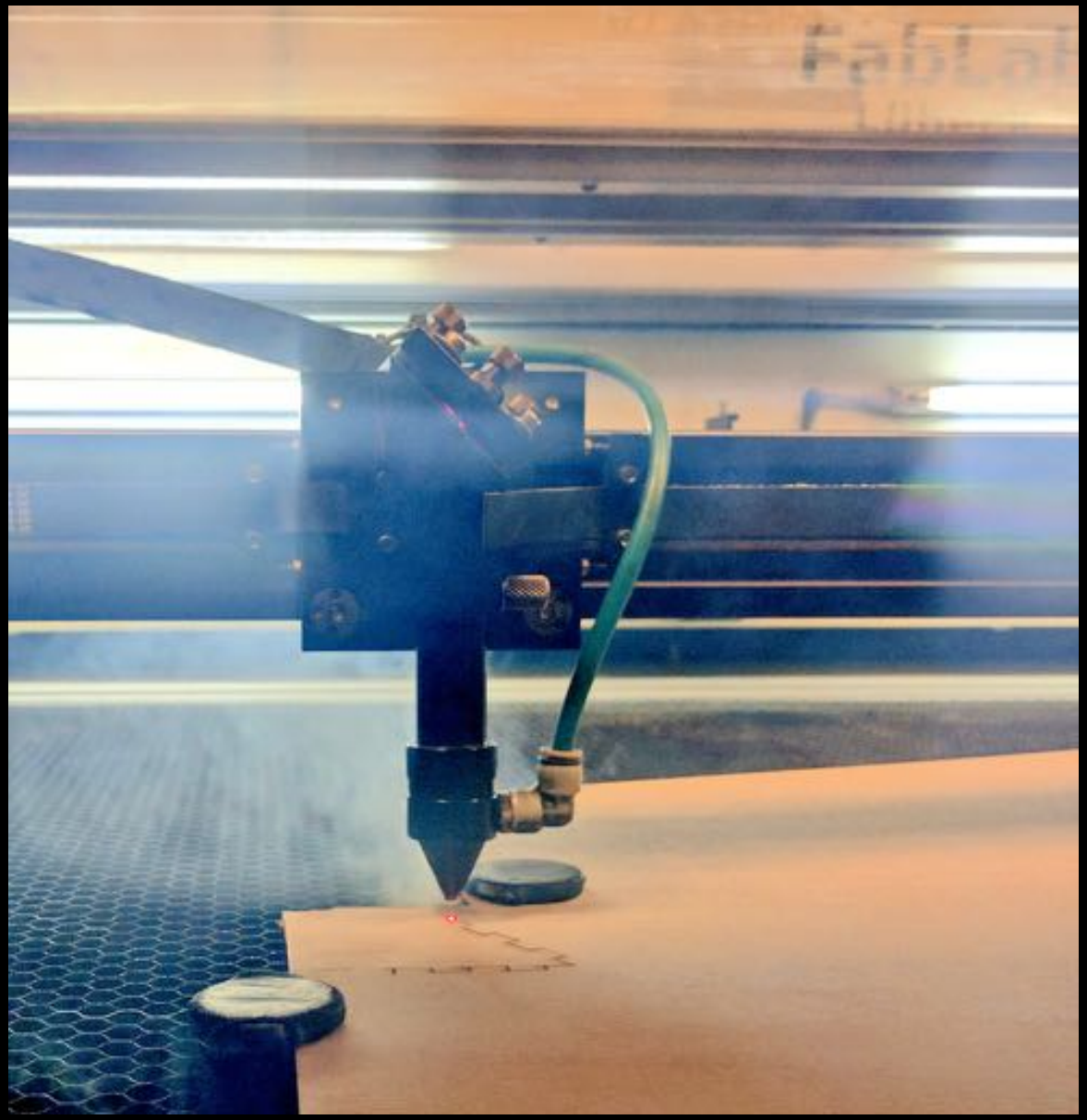
- Löten gelernt
- Python vertieft
- Umgang mit 3D-CAD (OpenSCAD)
- Lasern
  - Holz

    (Kudos Ivo)
  - Plexiglas

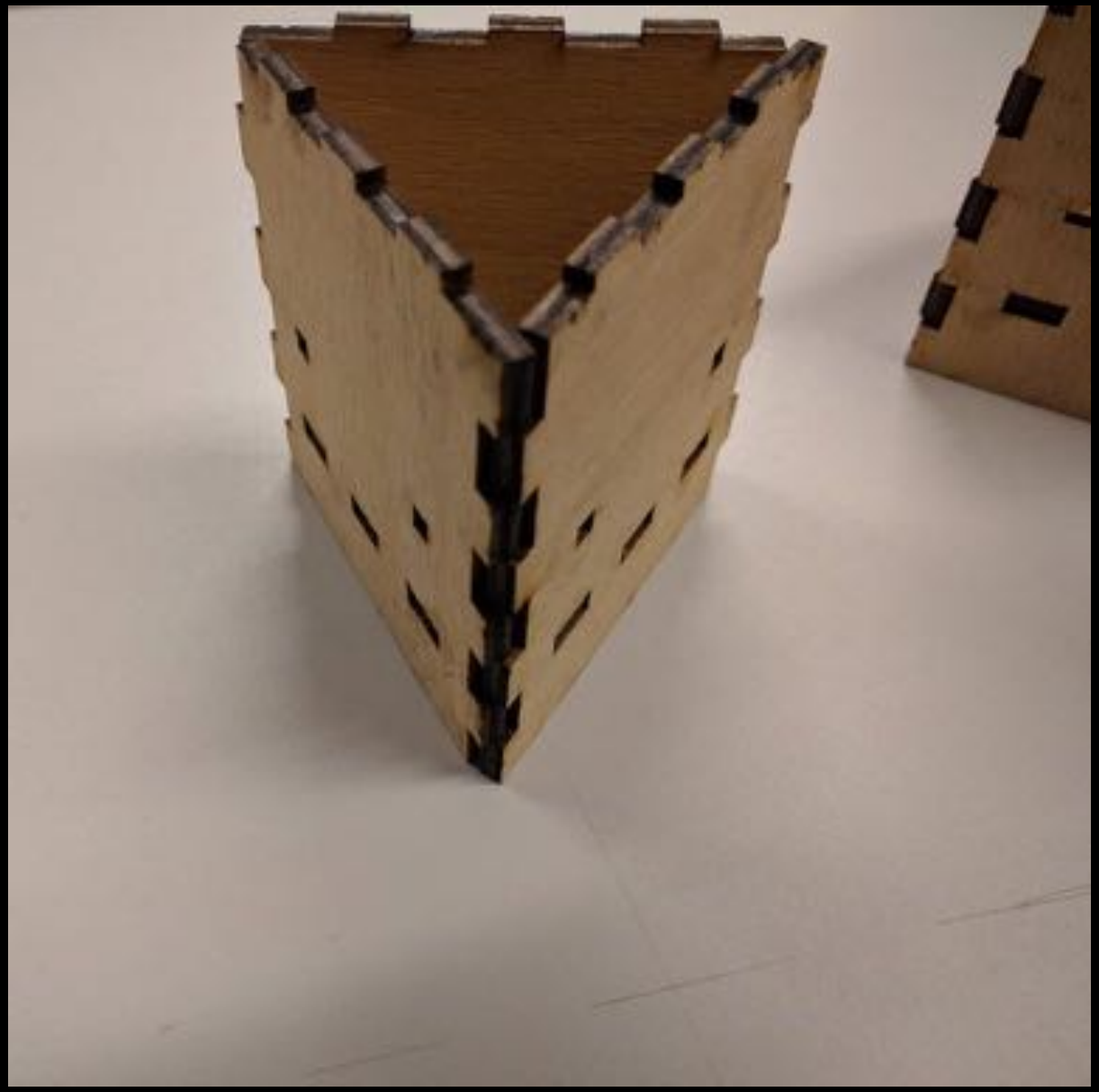
    (Kudos Maurice & FabLab KA)
- 3D-Druck der LED-Halterungen

  (Kudos eBrnd)
- eigener 3D-Drucker
- Retro-Pixelschubsen

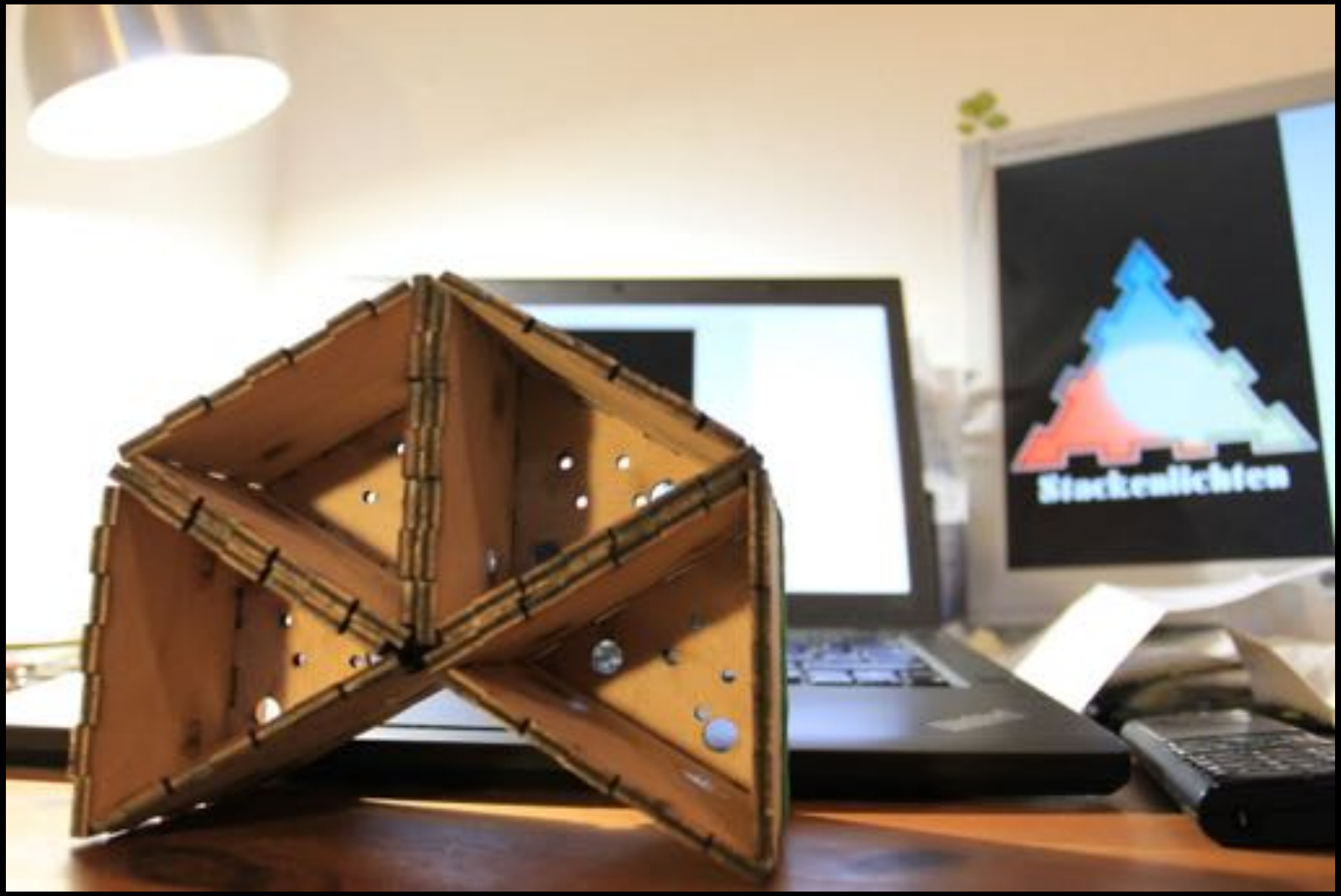
Bilder



Bilder

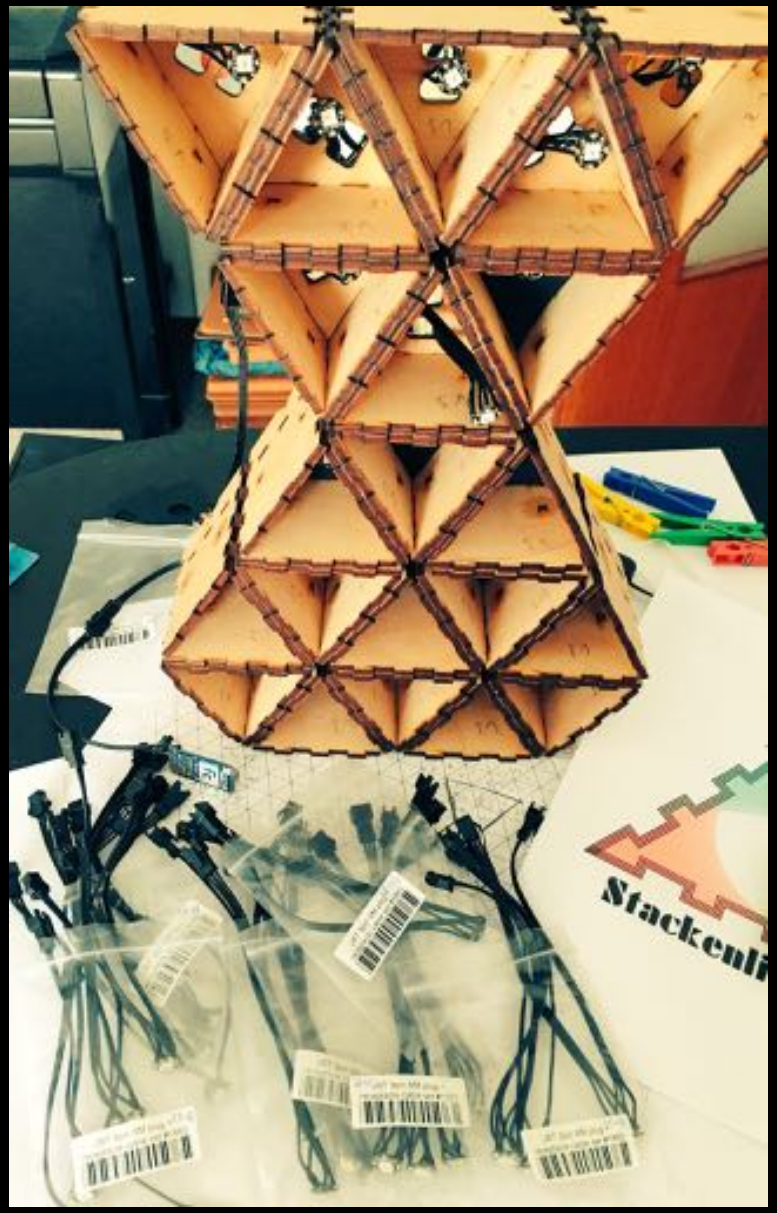


Bilder



Bilder

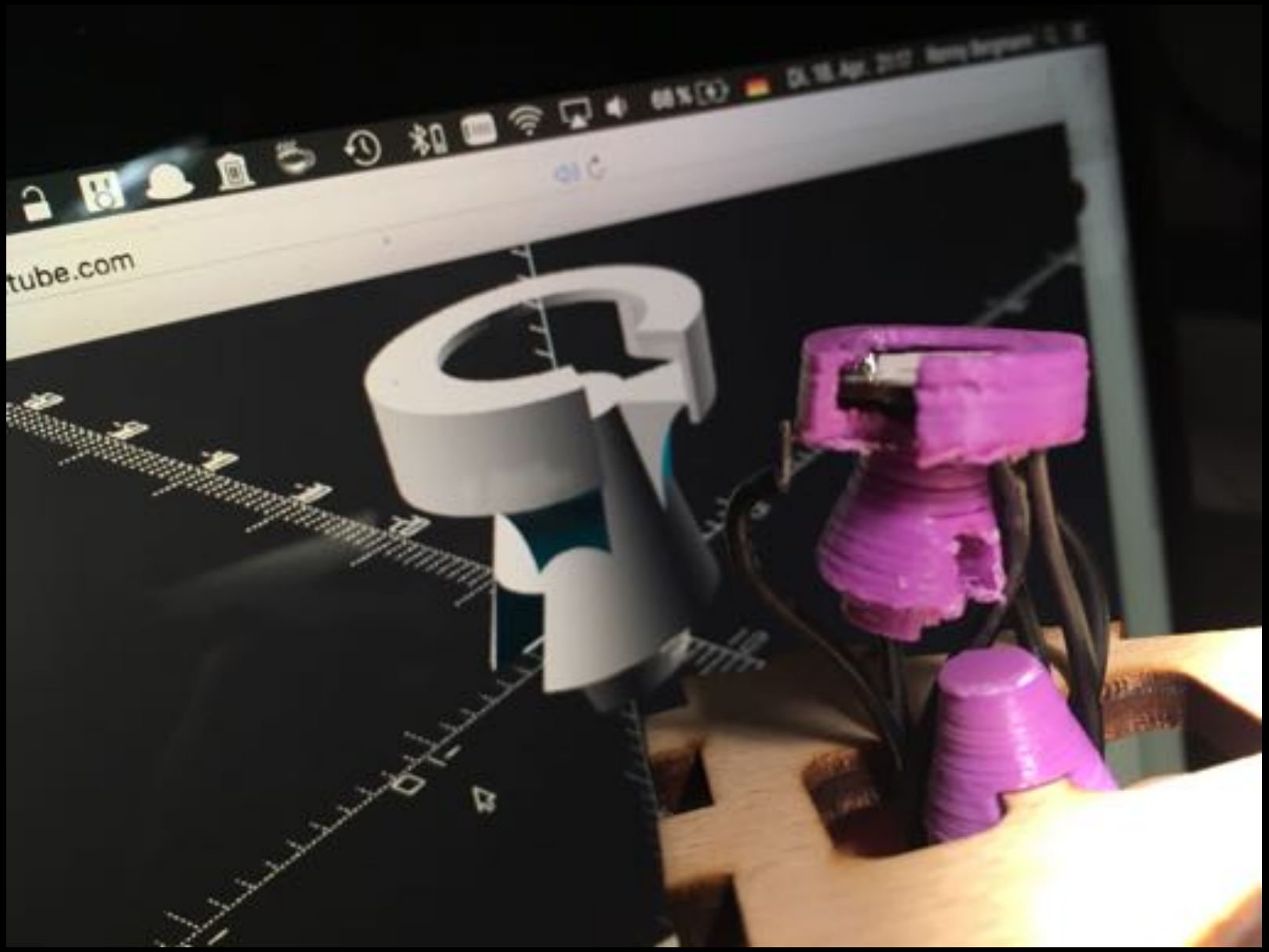
Bilder



Bilder

Bilder

Bilder

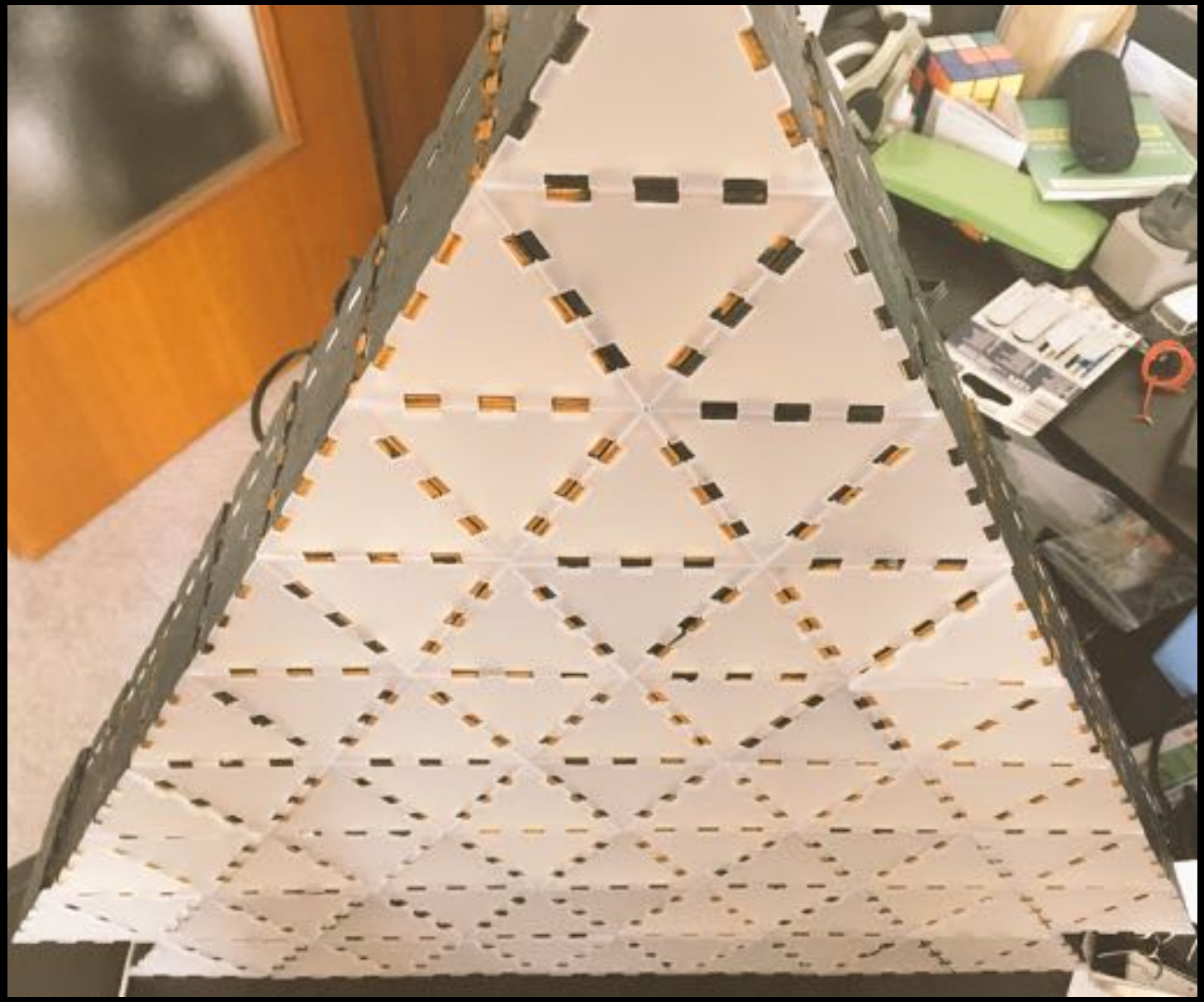


Bilder

Bilder

Bilder

Bilder



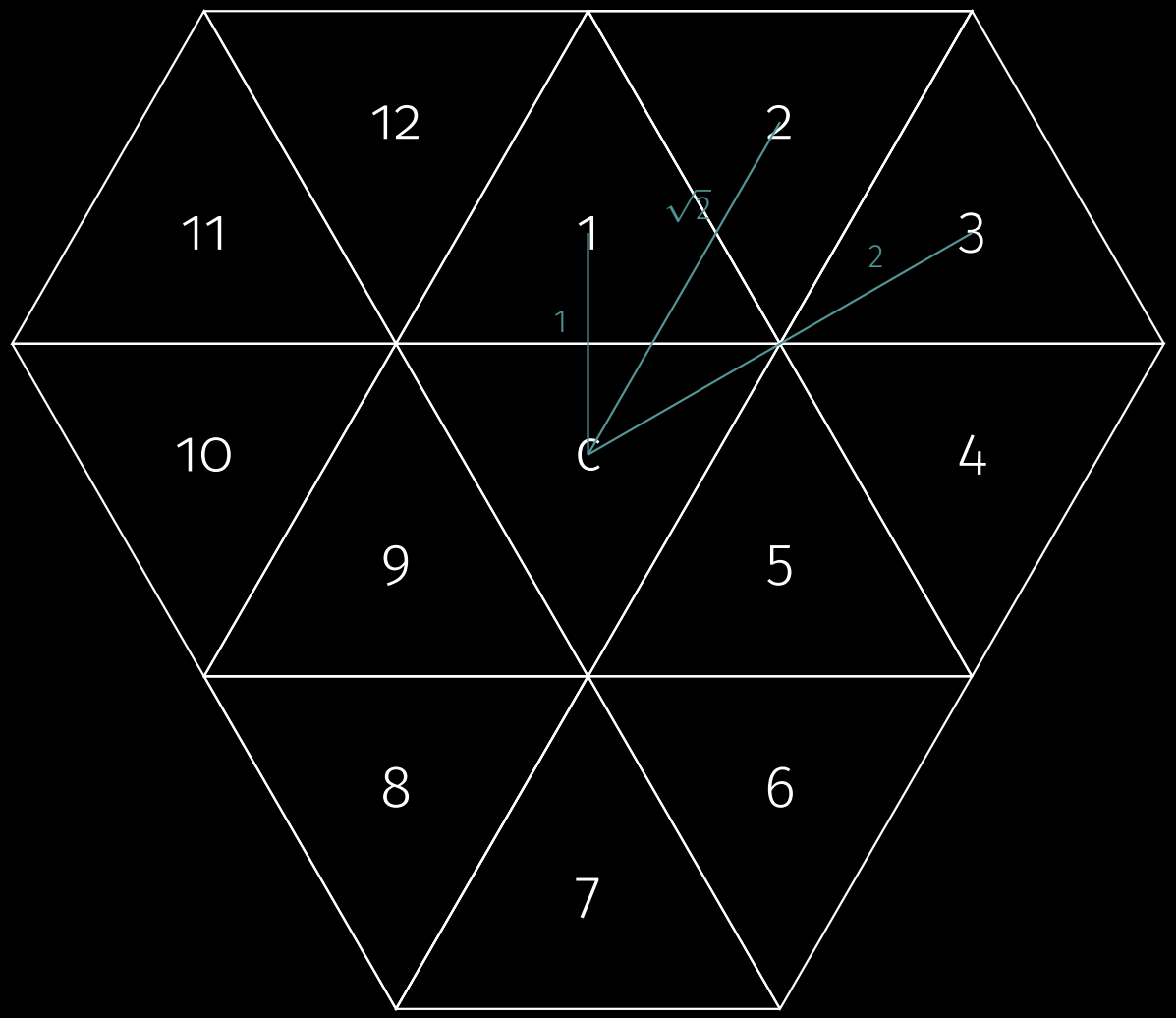
Bilder

Bilder

Bilder

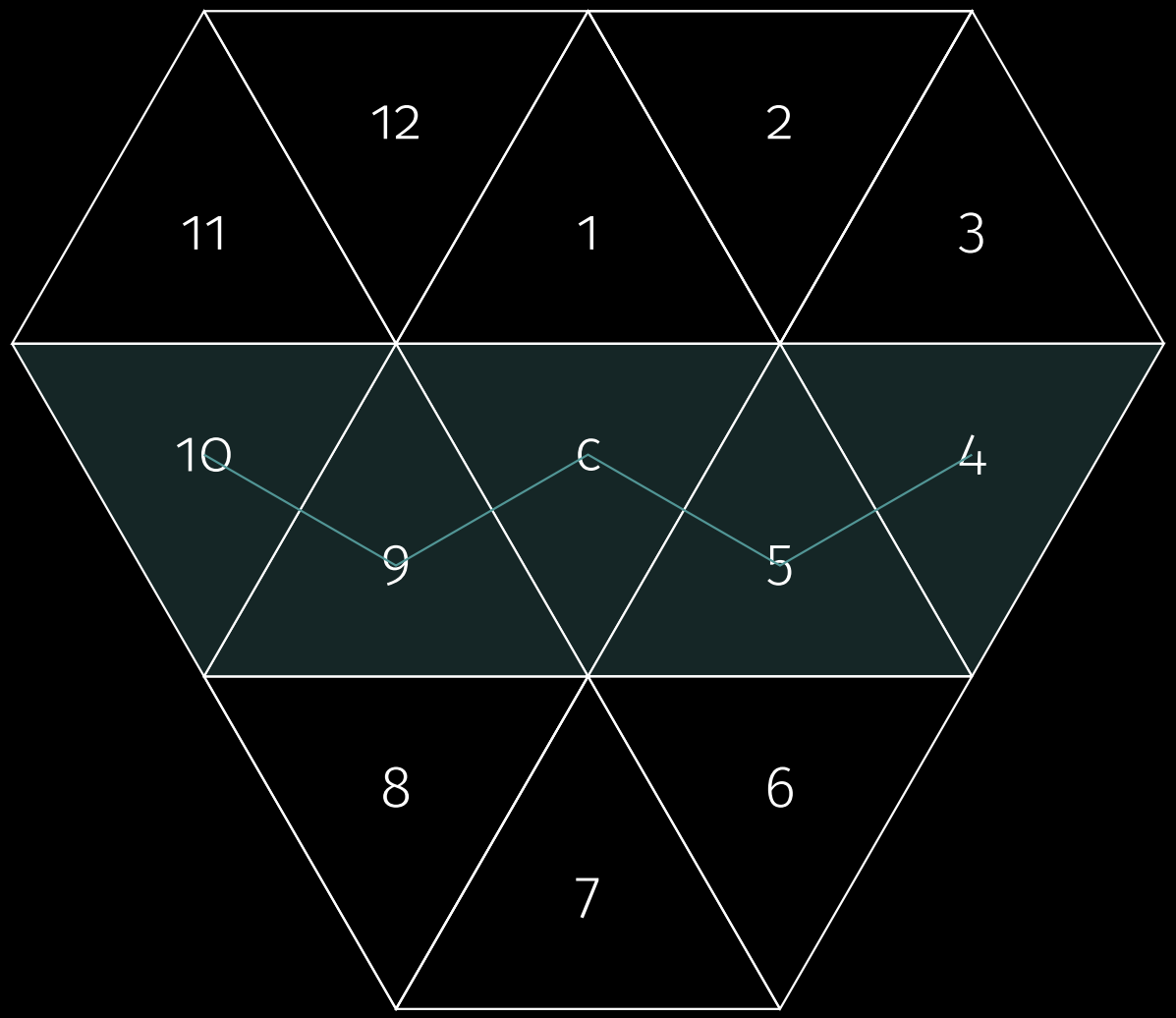
Bilder

Zum Graphen in den Trixeln



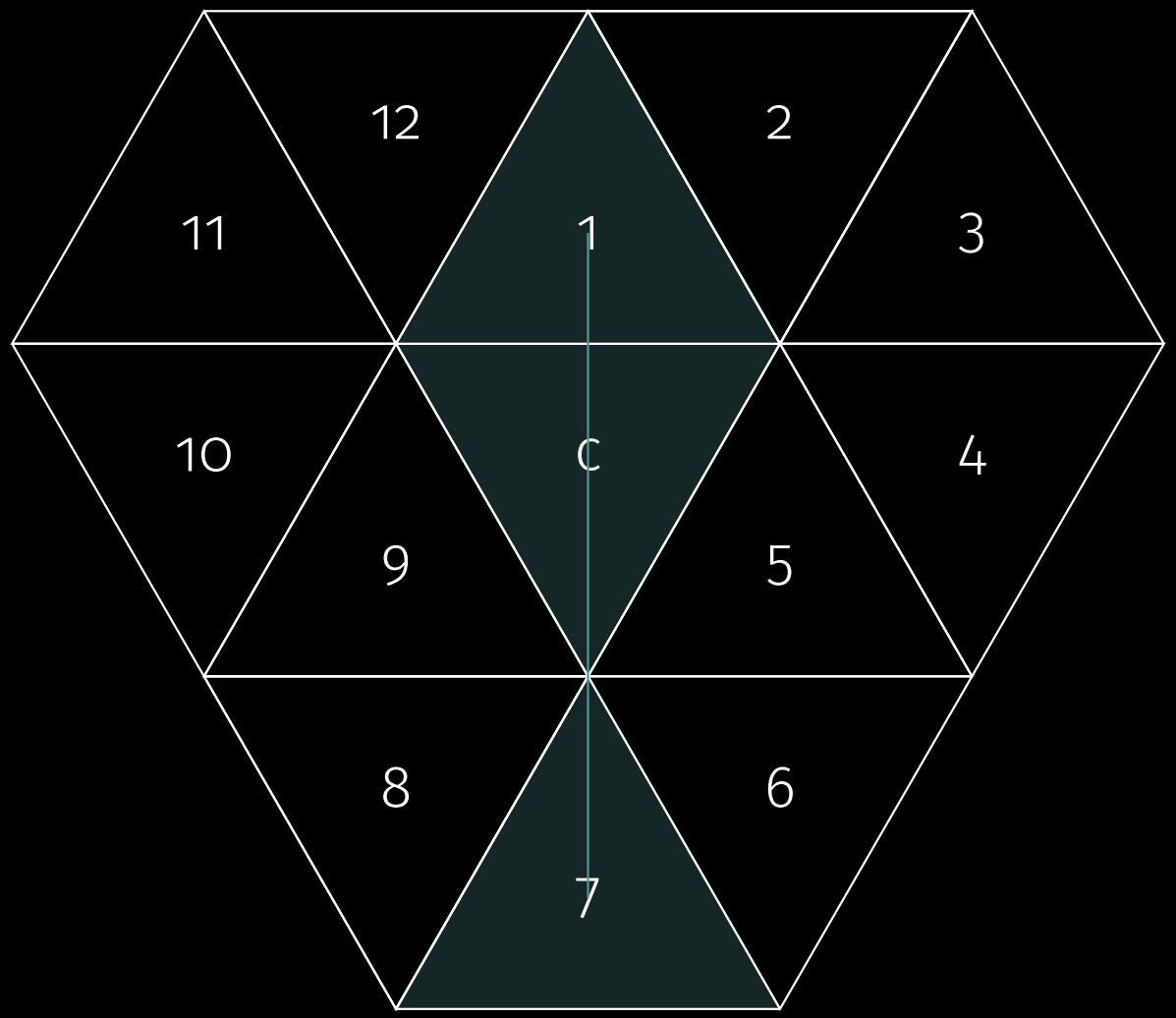
12 Nachbarn pro Pixel, alle 30°.

Zum Graphen in den Trixeln



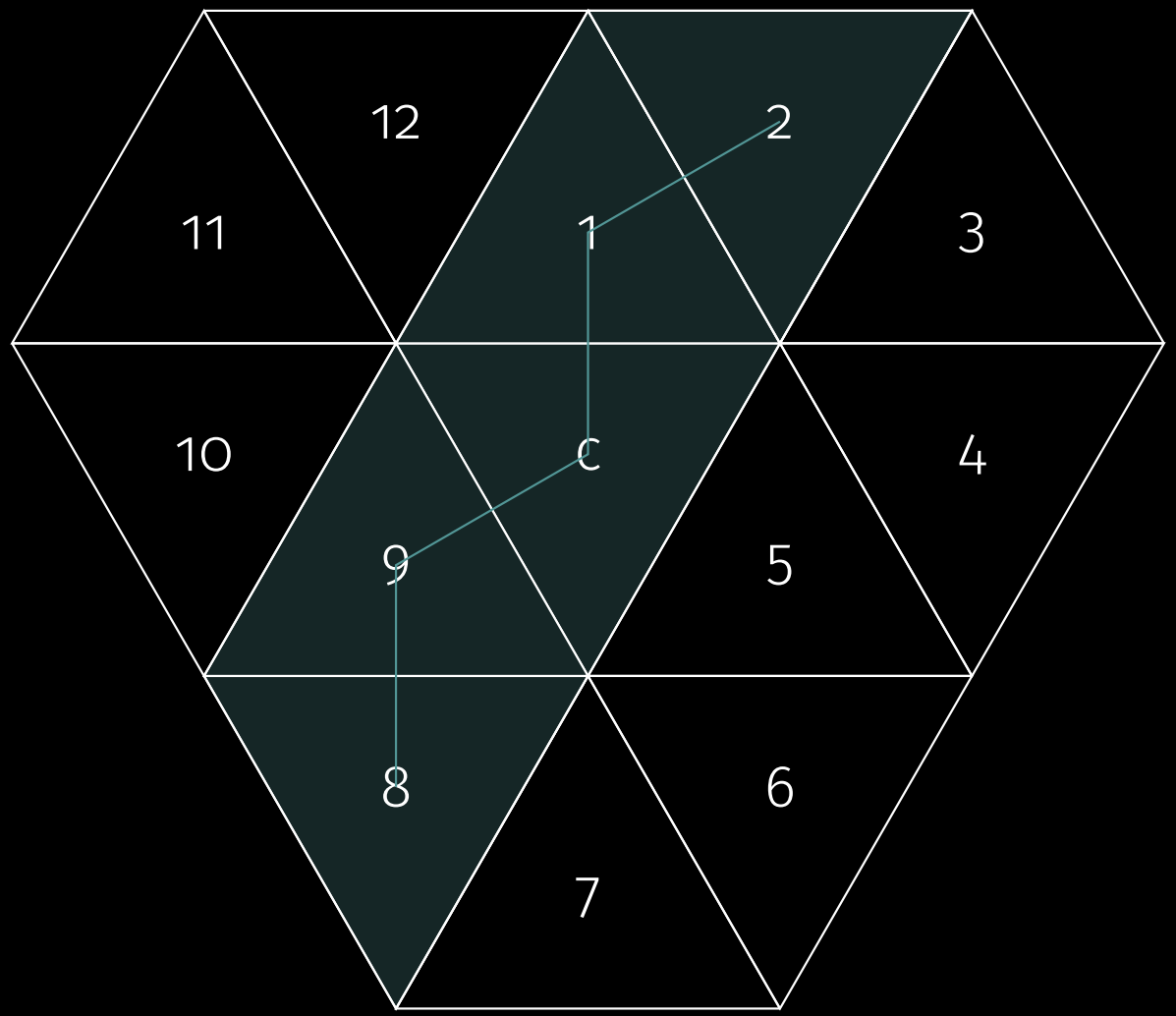
Horizontal: Sieht gut aus, komischer Weg (120° & dessen 60°)

Zum Graphen in den Trixeln



Vertikal: einfacher Weg , sieht doof aus

Zum Graphen in den Trixeln



30°: wie horizontal, also alle 60° sind ok.

Demo

Achtung!

Alles Turisten und Nonteknischen Lookenpeepers!
Das Komputermaschine ist nicht für
der Gefingerpoken und Mittengraben!
Oderwise ist easy to Schnappen der Springenwerk,
Blowenfusen und Poppencorken mit Spitzensparken.
Ist nicht für Gewerken bei Dummkopfen.
Der Rubbernecken Sightseeren keepen
das Cottonpicken Händer in das Pockets muss.
Zo Relaxen und Watschen der Blinkenlichten.