

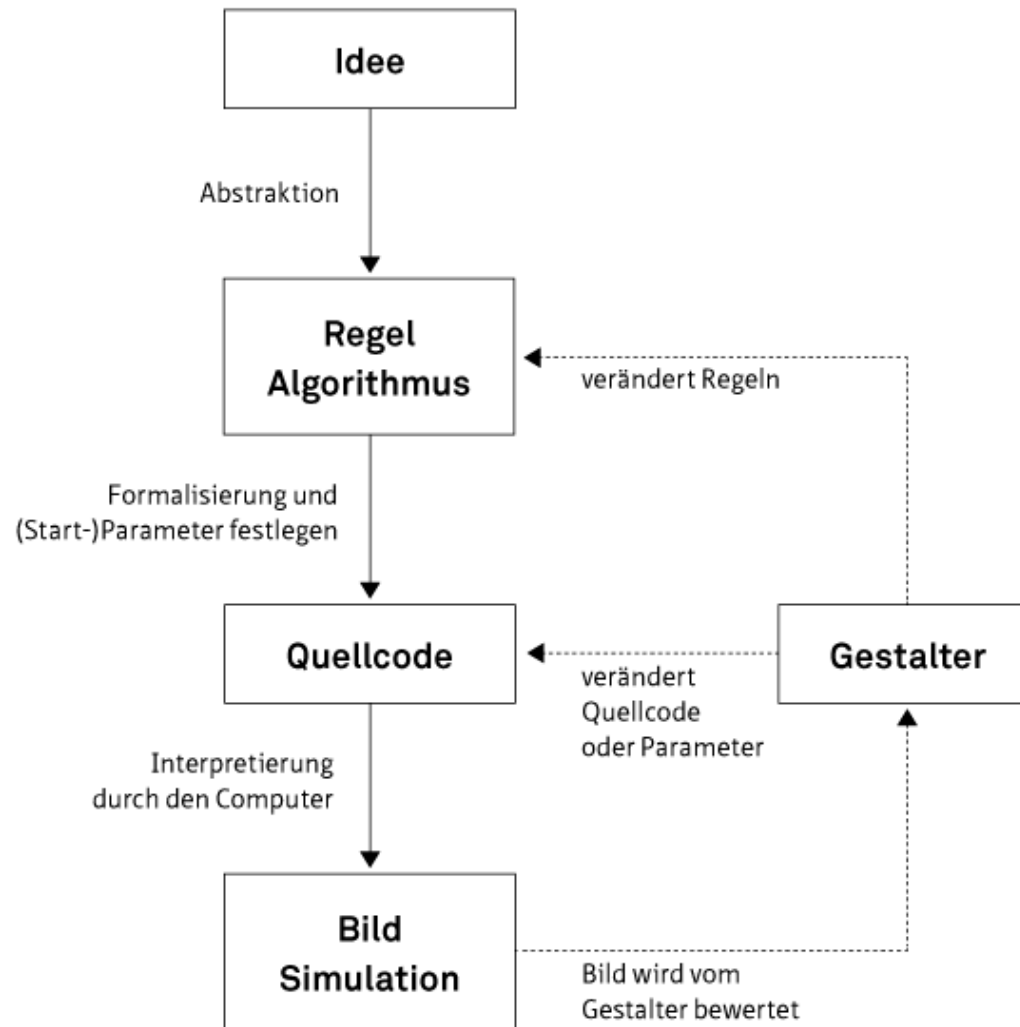
Leitgedanken

Universalität

Spezifische Anwendungen

Programmierung

Abstraktions- und Formalisierungsprozesse



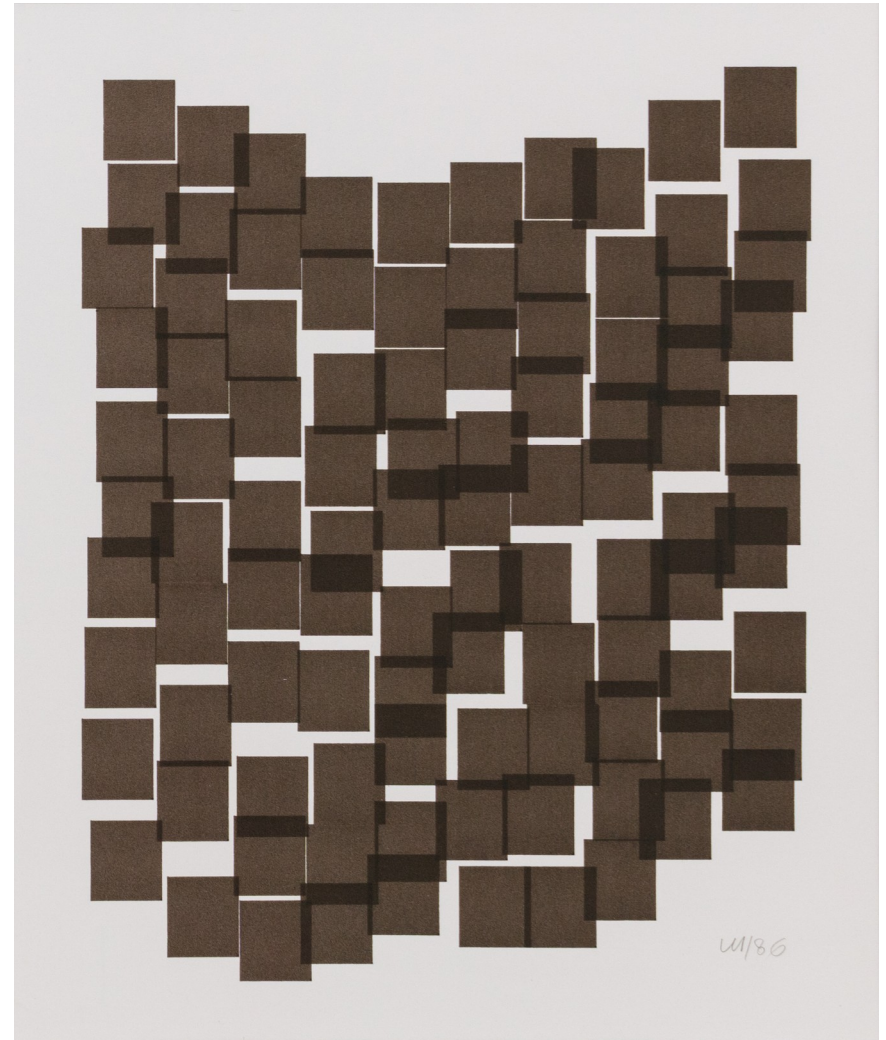
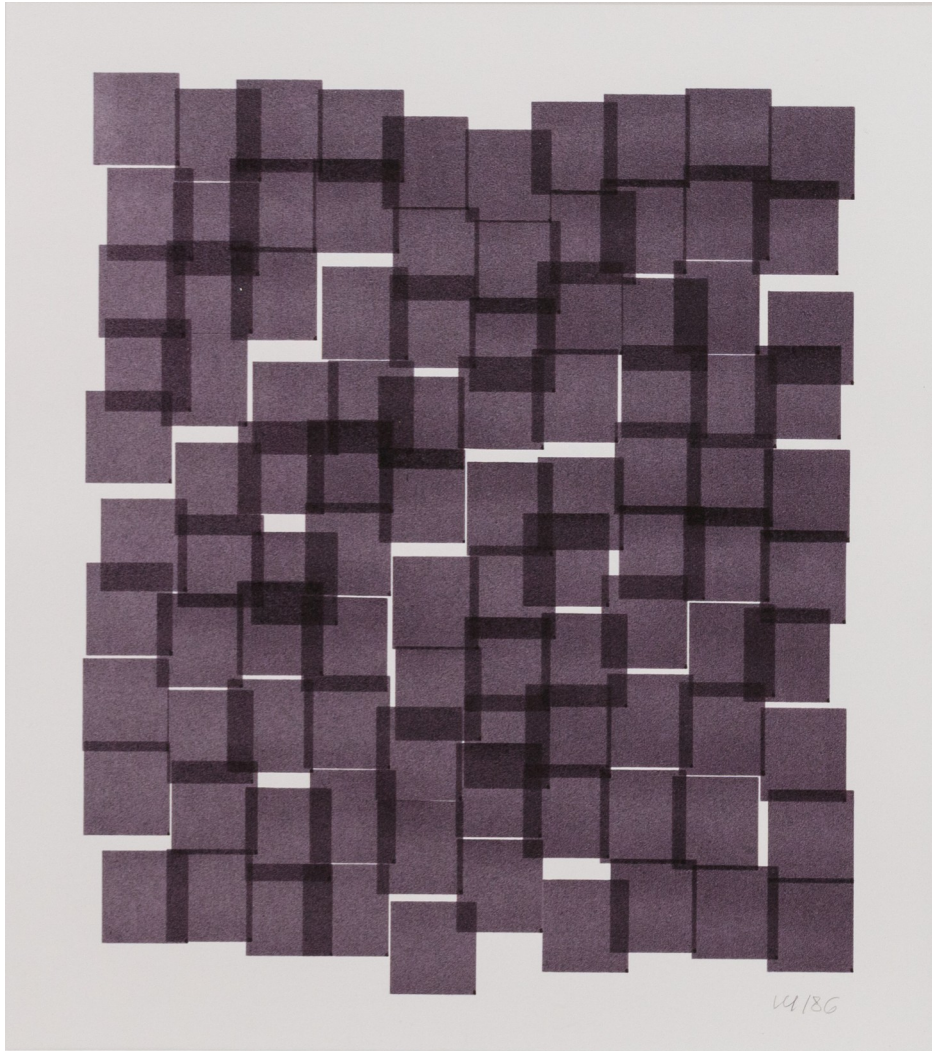
Copyright Hartmut Bohnacker, Julia Laub, Benedikt Groß, Claudius Lazzaroni (2009)
Buch „Generative Gestaltung“, www.generative-gestaltung.de

Programmierung und Poesie

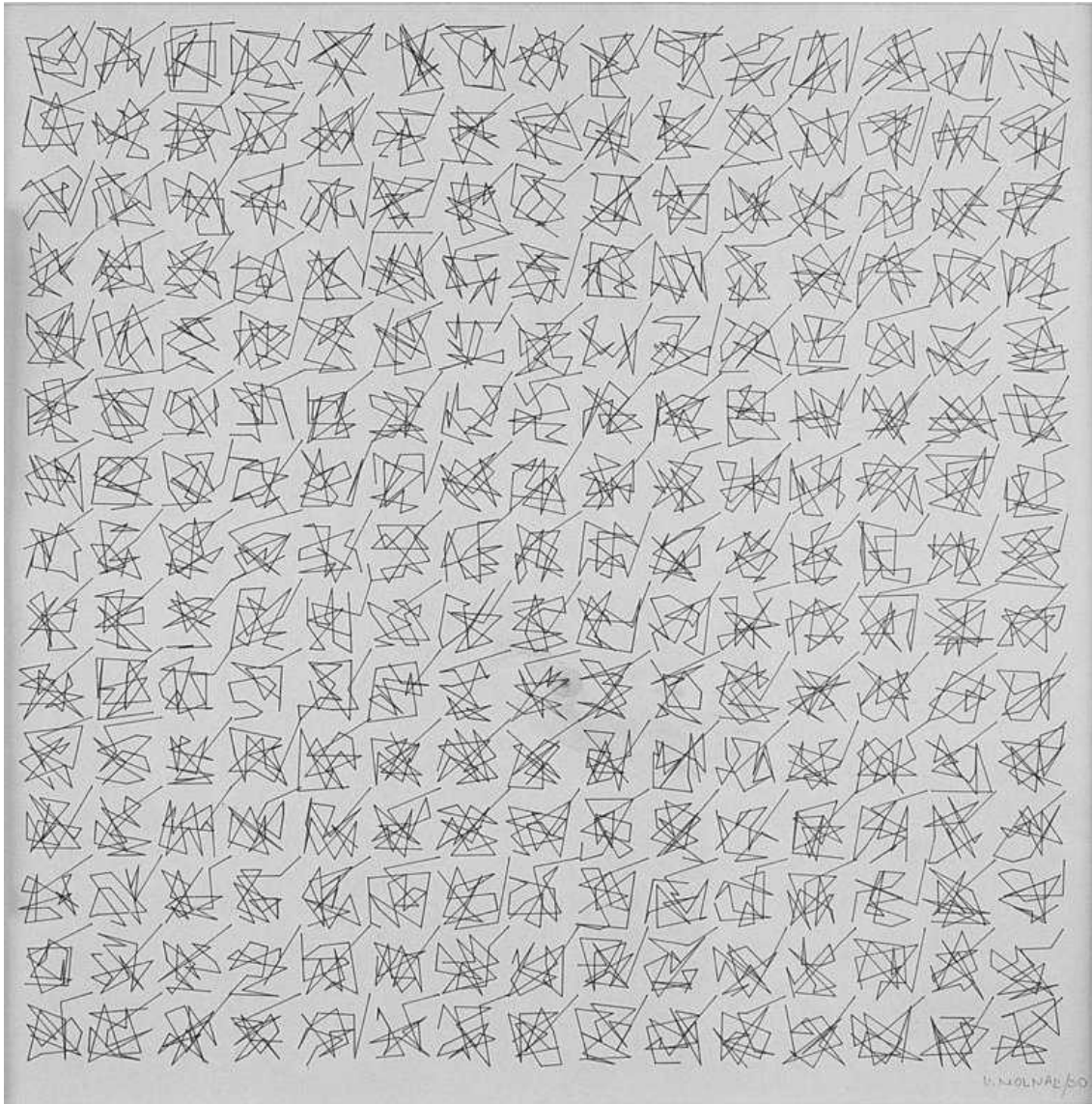


Lara Celman, Enamored, 2004-2005

<http://www.mikolaskova.cz/nop/streifzuege>

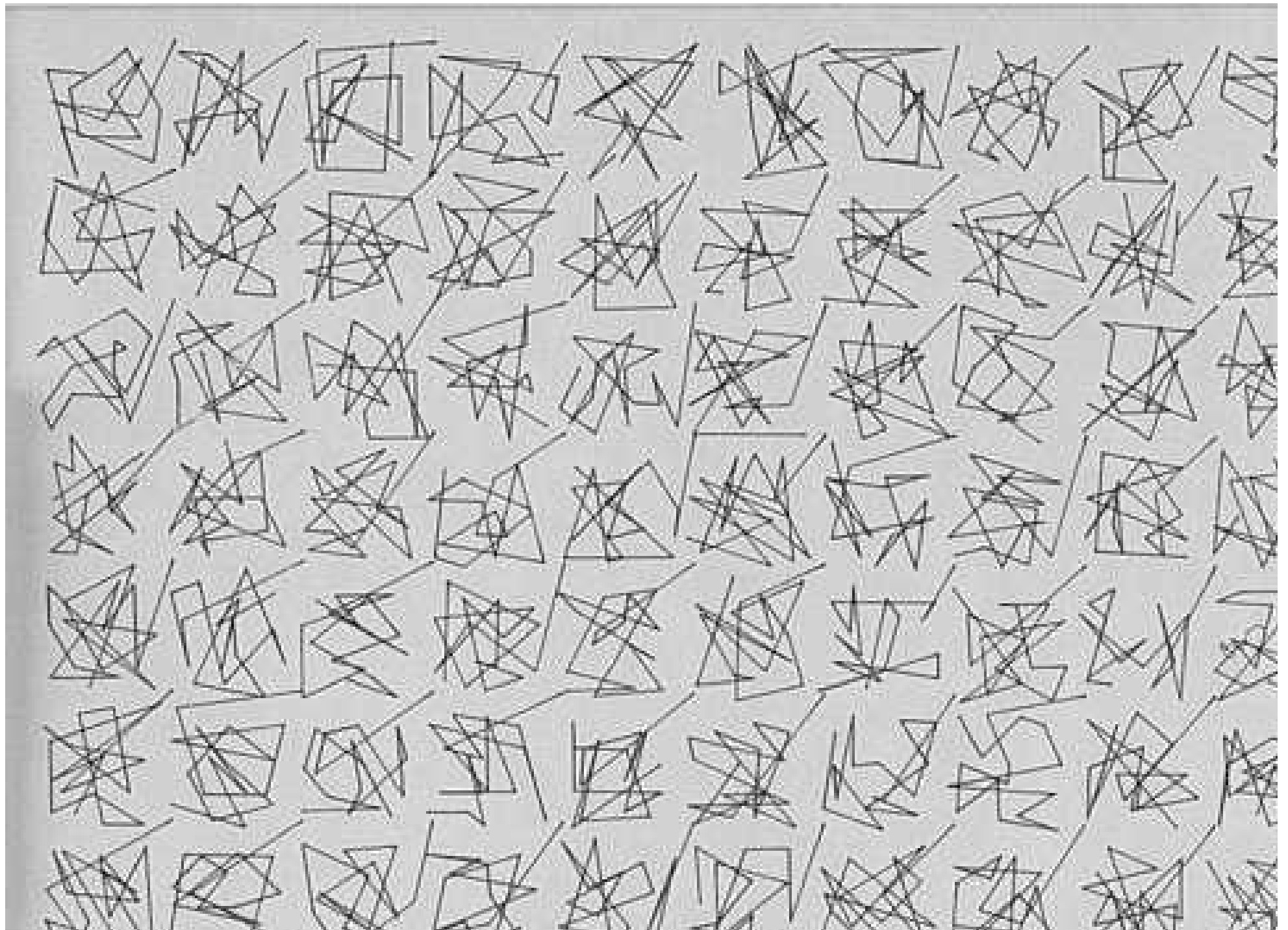


Vera Molnar, 100 Carrés, 1986, Plotterzeichnung, 25 x 22 cm



Vera Molnar,
Hommage à Dürer, 225 variations
aléatoires, direction chaos
plotter drawing, open series,
1990, +/- 30 x 30 cm





Vera Molnar, Hommage à Dürer, 225 variations aléatoires, direction chaos plotter drawing, open series, 1990, +/- 30 x 30 cm

<http://www.mikolaskova.cz/nop/streifzuege>

Einführung des Gymnasialfachs Informatik - Bildungspolitisches

Erstens brauchen wir mehr kritisches Denken an unseren Schulen. Kritisches Denken bedeutet, nicht nur darauf zu fokussieren, Fakten, Modelle oder Methoden zu erlernen, um am Schluss das Produkt «Wissen» zu haben. Vielmehr geht es um die Prozesse, wie Wissen erzeugt wird und wie man eigene Ideen entwickelt. Im Vordergrund stehen für mich das Sammeln von Erfahrungen, die Versuche, Probleme zu lösen und Sachinhalte zu verstehen. **Es geht darum, Hypothesen zu formulieren und zu überprüfen; um die kreative Arbeit an Ideen und darum zu testen, ob sie funktionieren.***

**aus Informatikunterricht – Quo vadis*

Juraj Hromkovic- Professor für Informationstechnologie und Ausbildung an der ETH Zürich

<https://www.ethz.ch/de/news-und-veranstaltungen/eth-news/news/2018/07/blog-hromkovic-informatikunterricht.html>

Thrilled by...

School for poetic computation



Lara Celman, Enamored, 2004-2005

School for poetic computation

School for Poetic Computation

Mission
People
Classes
Press

Participate
Code Words
Spring 2019
FAQ

Twitter 
Blog
Finances 
Instagram 

The school for poetic computation is organized around exploring the creative and expressive nature of computational approaches to art and design. The school approaches writing code like creative writing — focusing on the mechanics of programming, the demystification of tools, and hacking the conventions of art-making with computation.

We value the craft necessary to realize an idea, recognizing that every writer needs space and time to hone their trade. Our school aims to provide a safe haven for you to get acquainted with the craft of coding at your own pace, make it your own, and investigate the space between creative process and craft. This takes conversations with colleagues and the right push at the right time.

The school aims to be more than a technical bootcamp. It is an opportunity to work intensively with a small group of students, faculty, and artists to explore questions about the poetics of computation. For us, computation is poetic when technology is used for critical thinking and aesthetic inquiry — a space where logic meets

electricity (hardware), math meets language (software) and analytical thinking meets creative experimentation.

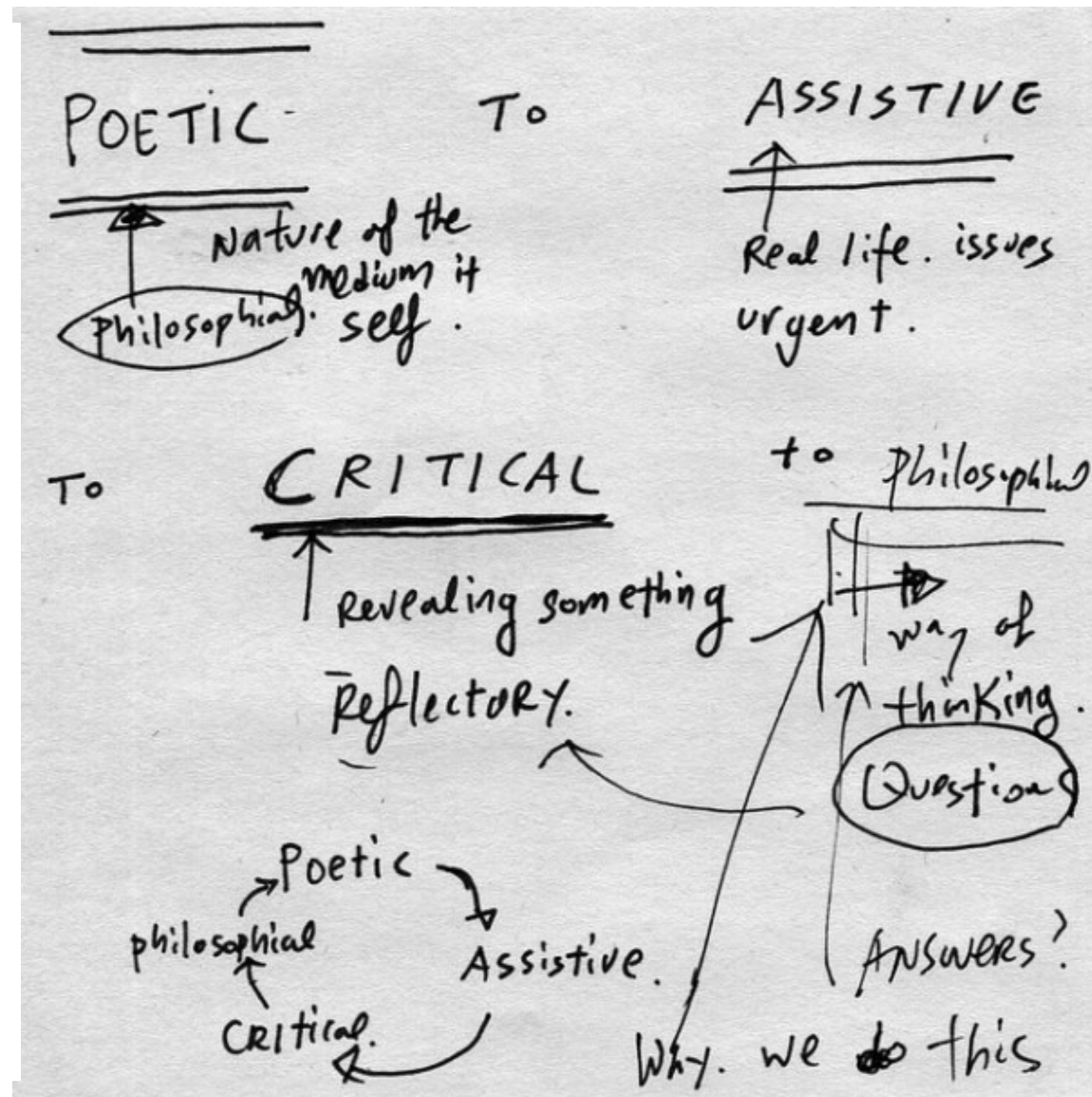
This is also a school for teaching. Every student who comes here will be asked to share their expertise with their classmates in the form of workshops and outreach.

The goal of the school is to promote completely strange, whimsical, and beautiful work — not the sorts of things useful for building a portfolio for finding a job, but the sort of things that will surprise and delight people and help you to keep creating without a job. However, employers tell us they appreciate this kind of work as well.

This is not a program to get a degree, there are large programs for that. This is not a program to go for vocational skills, there are programs for that. This is a program for self-initiated learners who want to explore new possibilities. This is a program for thinkers in search of a community to realize greater dreams.

Join our mailing list for more info:

Subscribe



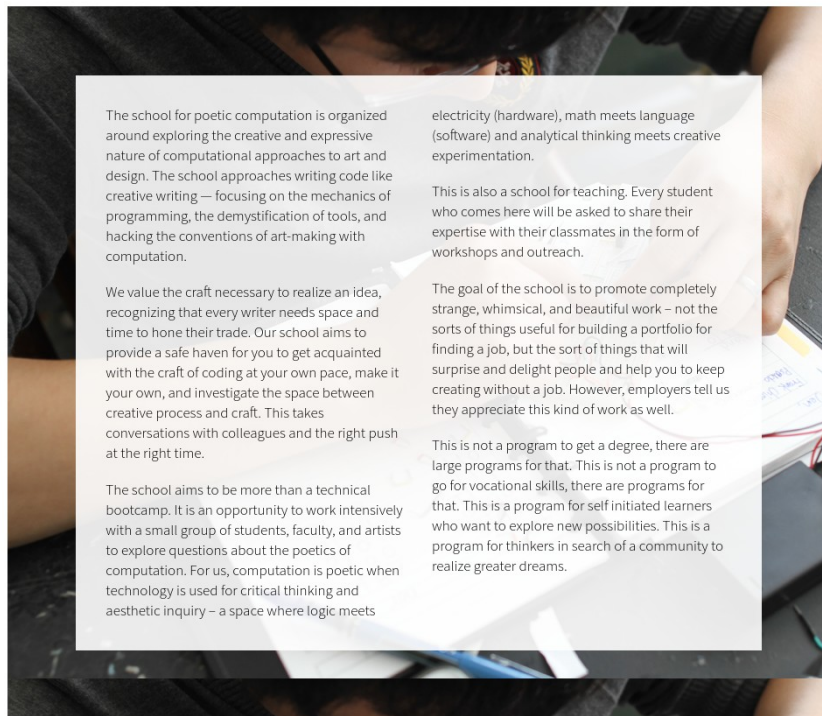
School for poetic computation

School for Poetic Computation

Mission
People
Classes
Press

Participate
Code Words
Spring 2019
FAQ

Twitter 
Blog
Finances 
Instagram 



The school for poetic computation is organized around exploring the creative and expressive nature of computational approaches to art and design. The school approaches writing code like creative writing — focusing on the mechanics of programming, the demystification of tools, and hacking the conventions of art-making with computation.

We value the craft necessary to realize an idea, recognizing that every writer needs space and time to hone their trade. Our school aims to provide a safe haven for you to get acquainted with the craft of coding at your own pace, make it your own, and investigate the space between creative process and craft. This takes conversations with colleagues and the right push at the right time.

The school aims to be more than a technical bootcamp. It is an opportunity to work intensively with a small group of students, faculty, and artists to explore questions about the poetics of computation. **For us, computation is poetic when technology is used for critical thinking and aesthetic inquiry – a space where logic meets electricity (hardware), math meets language (software) and analytical thinking meets creative experimentation.**

This is also a school for teaching. Every student who comes here will be asked to share their expertise with their classmates in the form of workshops and outreach.

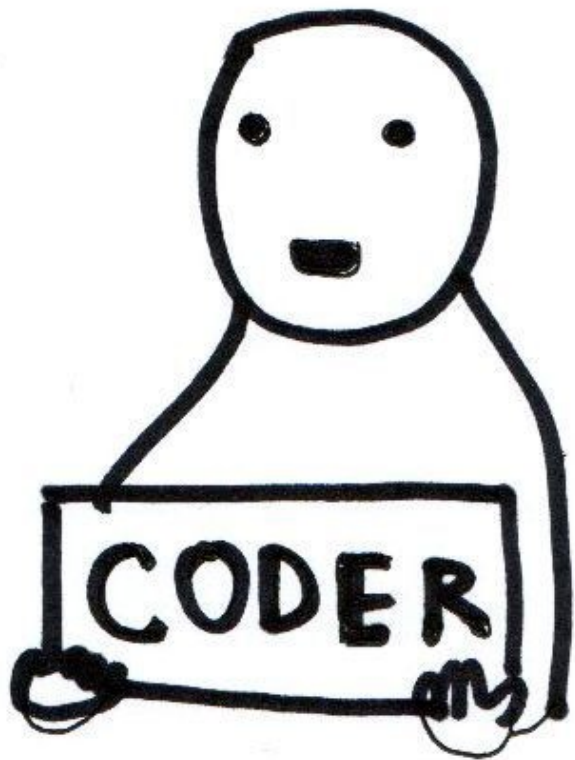
The goal of the school is to promote completely strange, whimsical, and beautiful work – not the sorts of things useful for building a portfolio for finding a job, but the sort of things that will surprise and delight people and help you to keep creating without a job. However, employers tell us they appreciate this kind of work as well.

This is not a program to get a degree, there are large programs for that. This is not a program to go for vocational skills, there are programs for that. This is a program for self initiated learners who want to explore new possibilities. This is a program for thinkers in search of a community to realize greater dreams.

Join our mailing list for more info:

Subscribe

School for poetic computation



Zeichnung von Taeyoon Chon (Mitglied SFPC)

<https://thecreativeindependent.com/people/taeyoon-choi-on-drawing-teaching-disability-and-the-difference-between-work-and-project/>

<http://www.mikolaskova.cz/nop/streifzuege>

School for poetic computation

Something that is poetic is very beautiful and expresses emotions in a sensitive or moving way

Aisthesis

„Wahrnehmung“, „Empfindung“

Poesis

Der Begriff Poietik ([griechisch](#) ποιητική poietike, zu **poiein ‚machen‘**) bezeichnet im ursprünglichen Sinn eine Wissenschaft oder

Lehre des Schaffens und Gestaltens

poetic

Being poetic is turning little things that no one would see right away and turning it into a wonderful meaning. To be poetic you must have a feeling of knowledge of things around you, taking things of your dreams and making it reality (UrbanDictionary)

po|e|tisch [po'e:tɪʃ] <Adj.>:

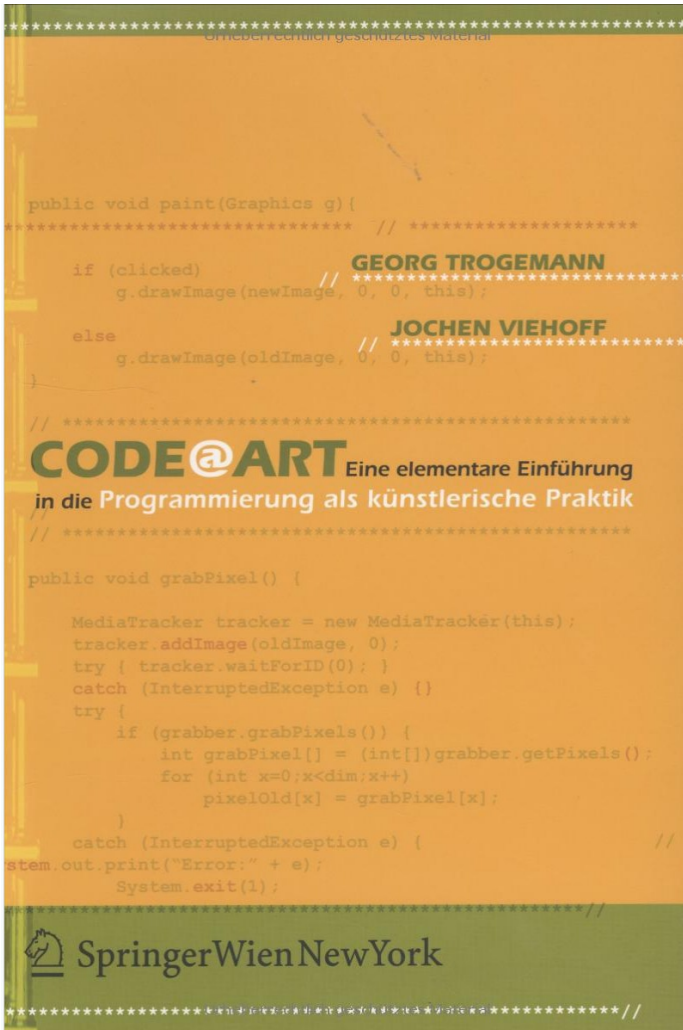
1. *die Poesie betreffend, zu ihr gehörend*: eine poetische Veranlagung haben.
2. *von dichterischer Ausdruckskraft*: eine poetische Schilderung, Sprache; ein sehr poetischer Film.

Code@Art

Eine elementare Einführung in die
Programmierung als künstlerische Praktik

Trogemann, Georg und Viehoff, Jochen

2004

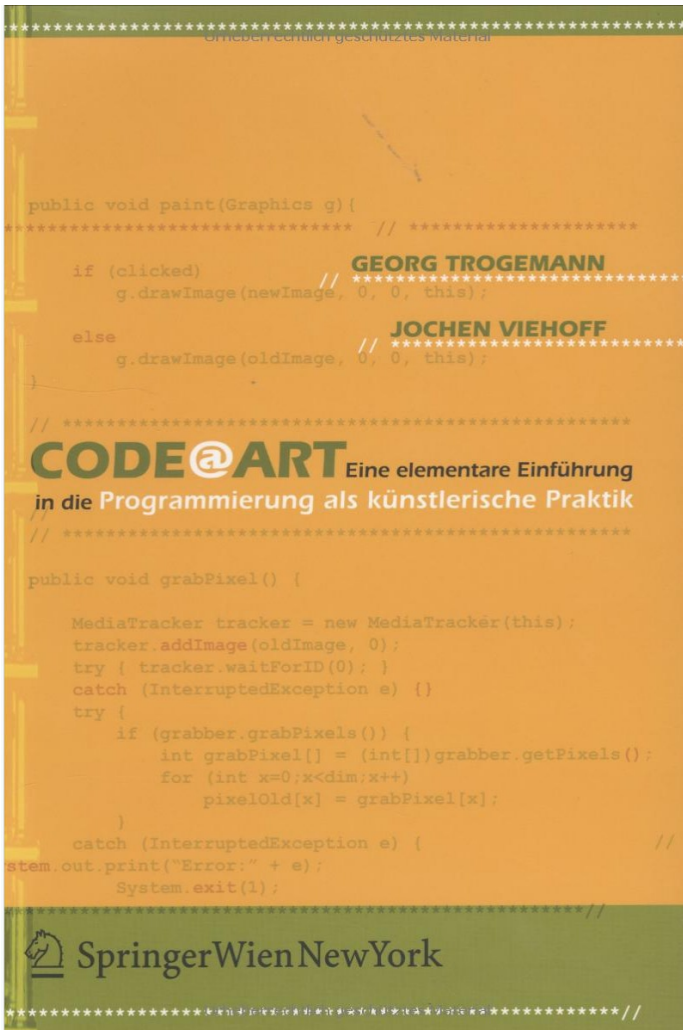


Code@Art

Eine elementare Einführung in die **Programmierung als künstlerische Praktik**

Trogemann, Georg und Viehoff, Jochen, 2004

Code Art versteht Programmierung als elementare Kulturtechnik und spürt den vielschichtigen **historischen Wurzeln und methodischen Hintergründen der programmierbaren Maschinen** nach. Neben einer einführenden Darstellung algorithmischer Grundstrukturen anhand der objektorientierten Sprache Java erhält der Leser einen Überblick über aktuelle Praktiken der Programmierung in künstlerischen und medienzentrierten Anwendungsfeldern. Darüber hinaus bietet das Werk in kompakten Zwischenspielen **spannende Einblicke in die Kultur- und Technikgeschichte der Programmierung.**



<http://georgtrogemann.de/code-art-interview-conducted-by-adriana-mikolaskova-nautsch/>

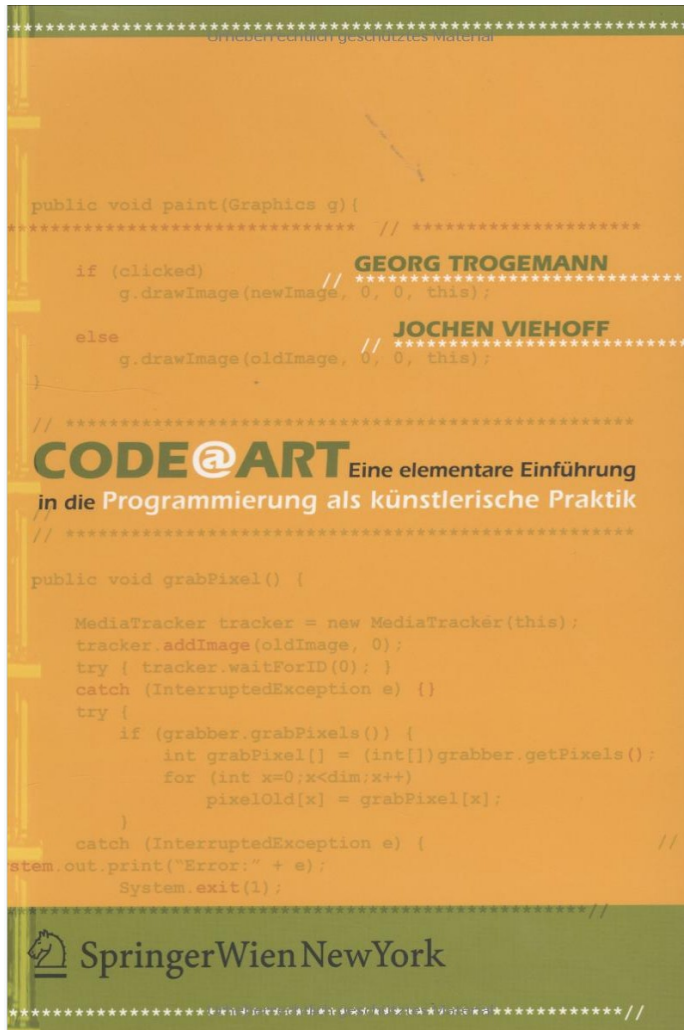
<http://www.mikolaskova.cz/nop/streifzuege>

Code@Art

Eine elementare Einführung in die
Programmierung als künstlerische Praktik

Trogemann, Georg und Viehoff, Jochen

2004



<http://georgtrogemann.de/code-art-interview-conducted-by-adriana-mikolaskova-nautsch/>

Siehe Heft 11, “programmiert”