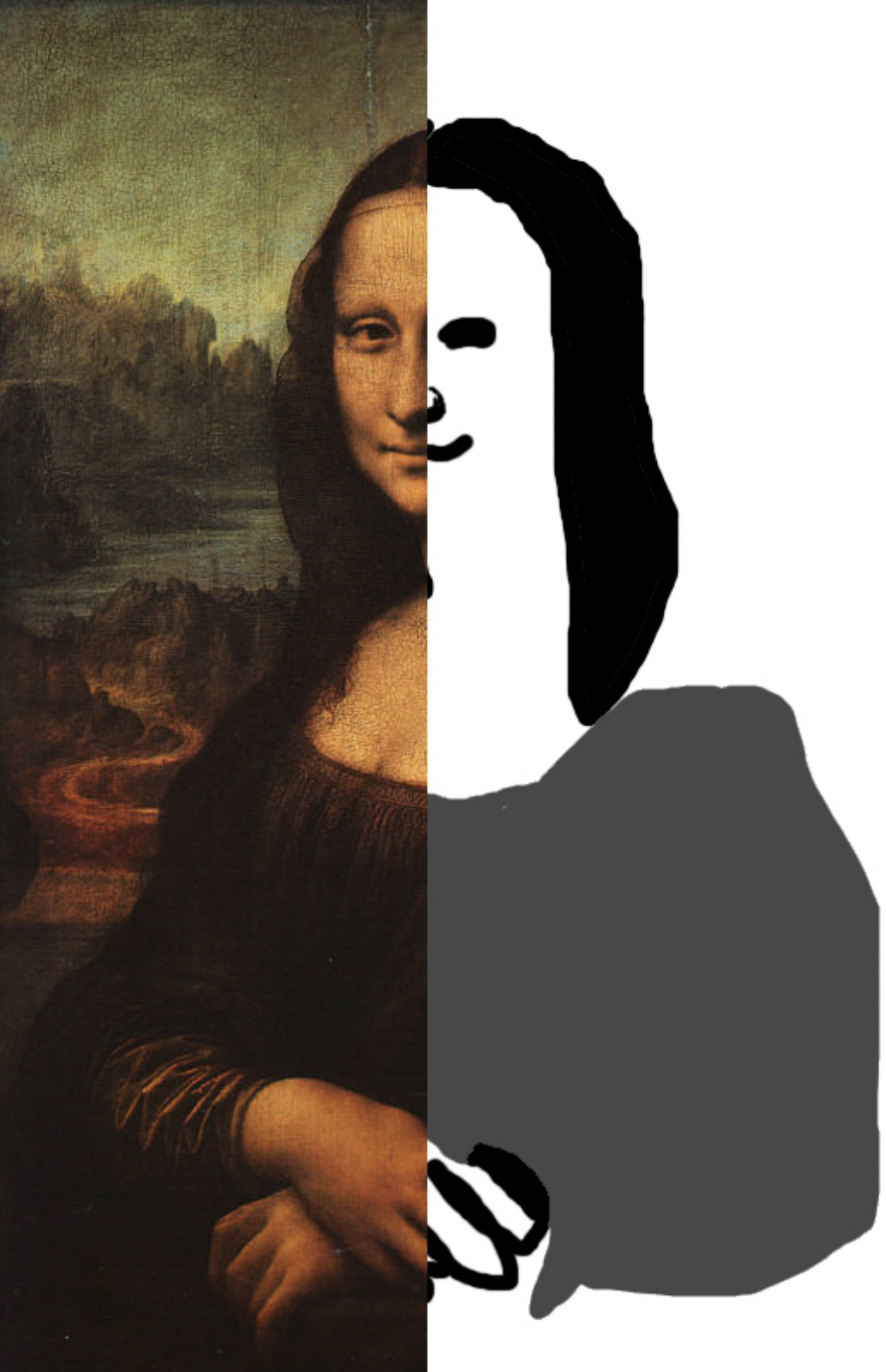
workshops in creative coding

Dr. Theo Papatheodorou

introduction



Computer Vision:

A set of algorithms that allow computers to understand images.

3D face recognition by machines

Imperial College London

Athens School Of Fine Arts

random quark

creative technology studio

http://randomquark.com

class purpose

- provide a fundamental understanding of code and modern computer literacy.
- introduce you to a range of techniques for creating interactive audiovisual software in C++
- develop your artistic practice
- assist you in delivering two large software projects

OpenFrameworks

class structure

- 1st term
  - syllabus
  - assessment
- 2nd term
  - syllabus
  - assessment

Mixing different abilities in the classroom

http://learn.gold.ac.uk

bureaucratic

- Email: theo@gold.ac.uk
- Workshops in Creative Coding website on learn.gold:
  https://learn.gold.ac.uk/course/view.php?id=6852
  enrollment key: **linux**
- MA/MFA Computational Arts website on learn.gold:
  https://learn.gold.ac.uk/course/view.php?id=8973
- absence policy: http://registermate.doc.gold.ac.uk
- Assessment:
  - lab+home assignments: 10%
  - end of term project: 90%
  - see class website for details

Pop-up exhibition
at the end of term

- We’ll book the church
- You’ll have it for 2 days
  - 1 day to prepare, 1 day for exhibition
- Great way to practice
- Showcase your work
- All modules on MA involved

Communication

- Learn.gold – forum of class:
  - code problems
  - technical issues
  - syllabus questions
  - things you didn’t understand in class
- Forum on MA/MFA learn.gold page:
  - events / announcements
  - cool links
  - Exhibitions
  - MA/MFA related questions

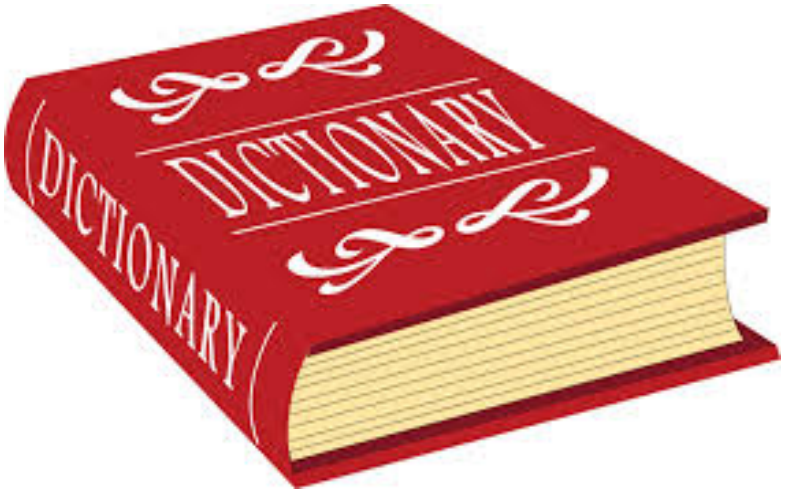
what is computational art?

“Could the artwork have been made without the use of a computer?”

“Does it take advantage of the computer’s unique capabilities?”

Roger Malina

SIGGRAPH '89, Leonardo Special Edition

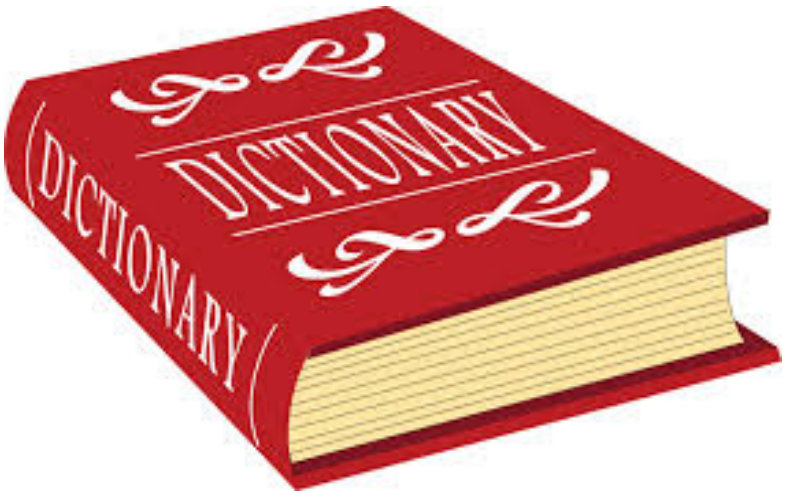


what is computational art?

“The computer’s utility in speeding up artistic operations can be justified as an artistic act.”

“Does your work with the computer affect the direction of your results?”

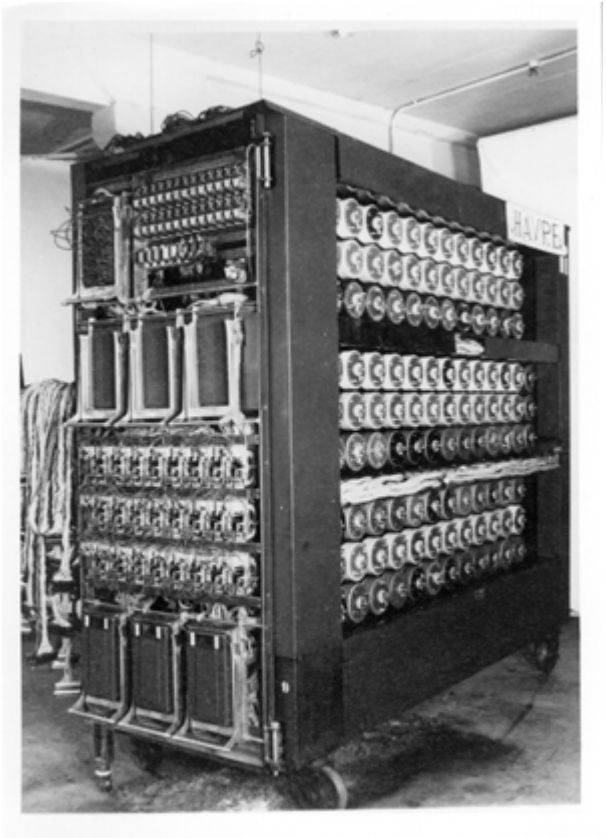
Aldo Giorgini – 1975
source



40s

code + art

- programming: a difficult task
- designers + artists have different requirements than engineers and scientists
- C++/Java take years to master



“The world was at war. Complex calculations had to be done under time pressures not normally felt by mathematicians. It's unlikely that the gave even a passing thought to making computers user-friendly to people with softer styles than theirs.”

- Seymour Papert

50-60s

code + art

- artists experimented with themes related to software and system aesthetics
  - Cybernetic Serendipidy exhibition (ICA, 1968)
  - Hans Haacke's Visitor's Profile (1971)

Cybernetic Serendipity

Serendipity

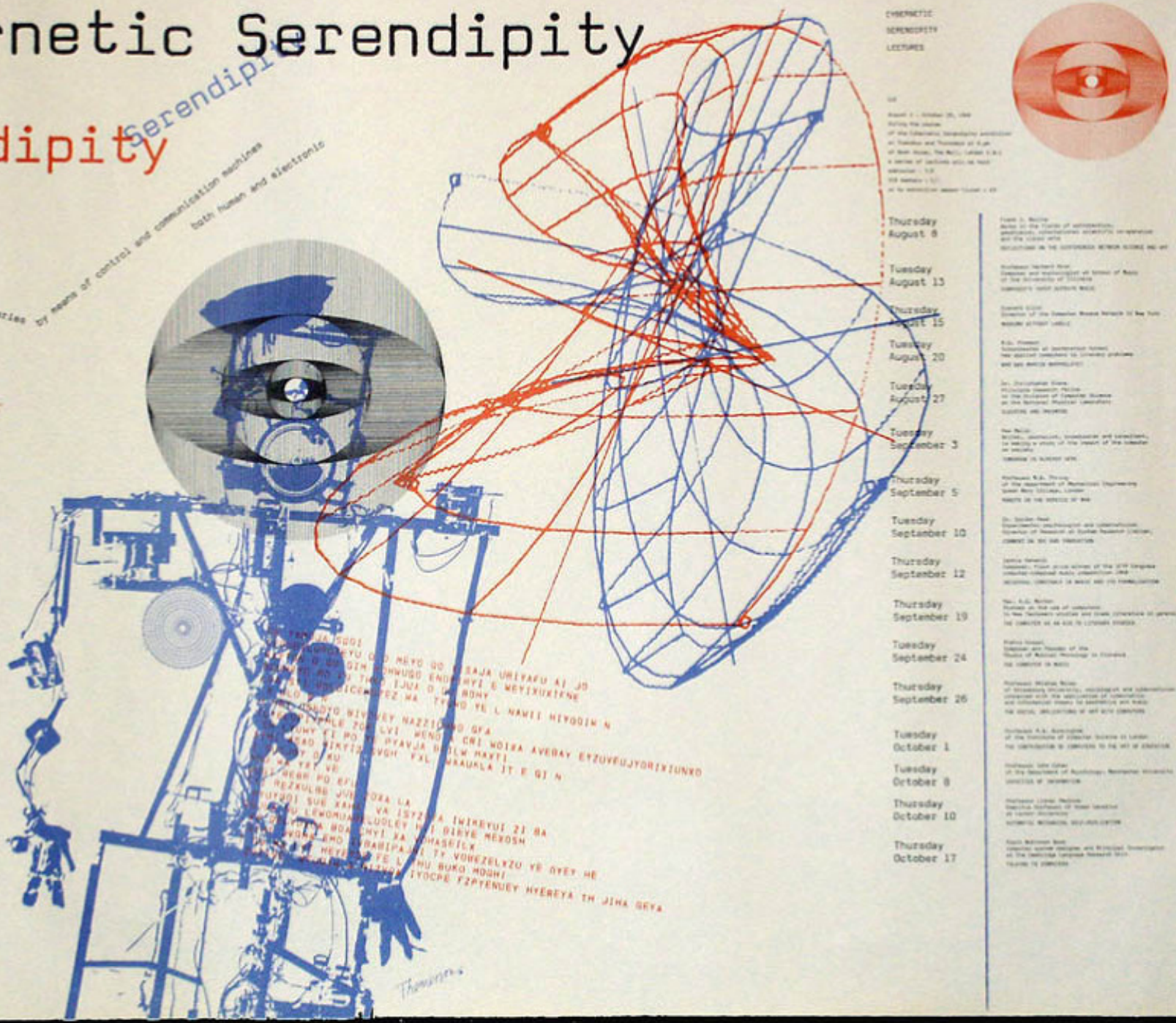
serendipity

the faculty of making happy chance discoveries

by means of control and communication machines both human and electronic

An exhibition

and other serendipitous manifestations



Institute of Contemporary Arts

August 2 – October 20

CYBERNETIC SERENDIPITY LECTURES

Thursday August 6

Tuesday August 13

Thursday August 15

Tuesday August 20

Tuesday August 27

Tuesday September 3

Thursday September 5

Tuesday September 10

Thursday September 12

Thursday September 19

Tuesday September 24

Thursday September 26

Tuesday October 1

Tuesday October 8

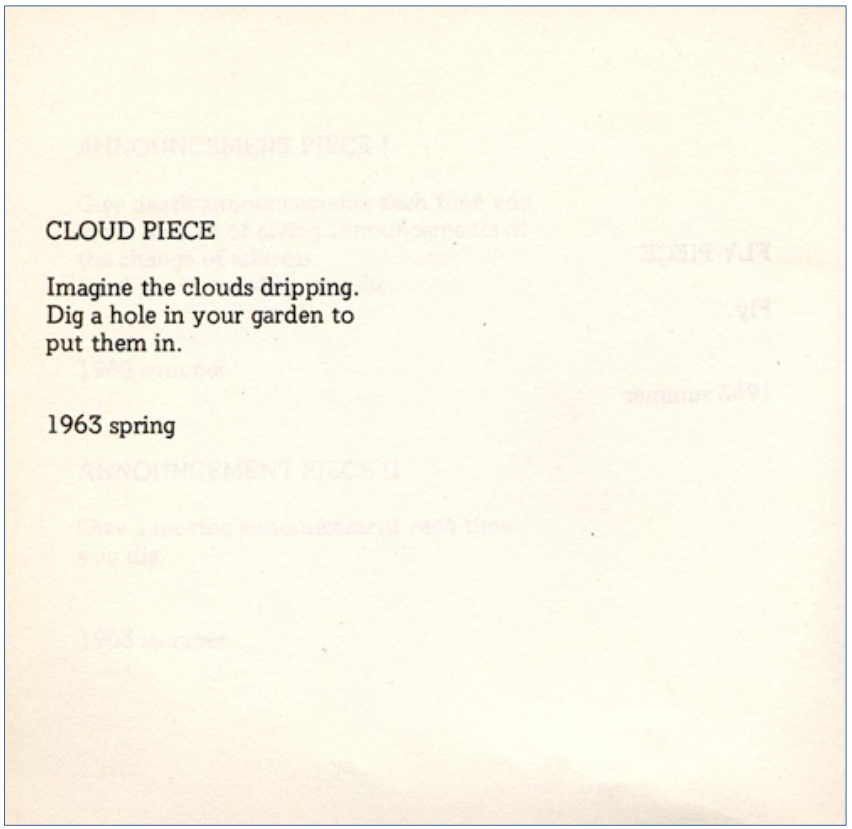
Thursday October 10

Thursday October 17

50-60s cont.

code + art

- process-based art
  - instructions and diagrams as a form of art

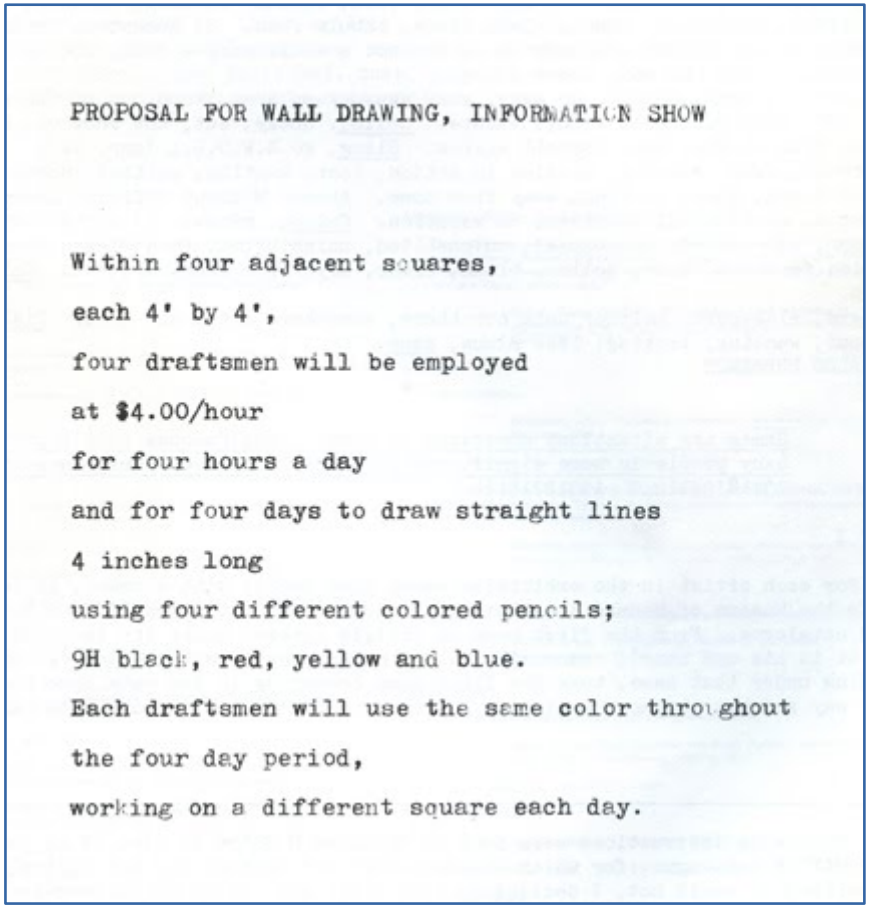


CLOUD PIECE

Imagine the clouds dripping.
Dig a hole in your garden to
put them in.

1963 spring

Yoko Ono (1963)



PROPOSAL FOR WALL DRAWING, INFORMATION SHOW

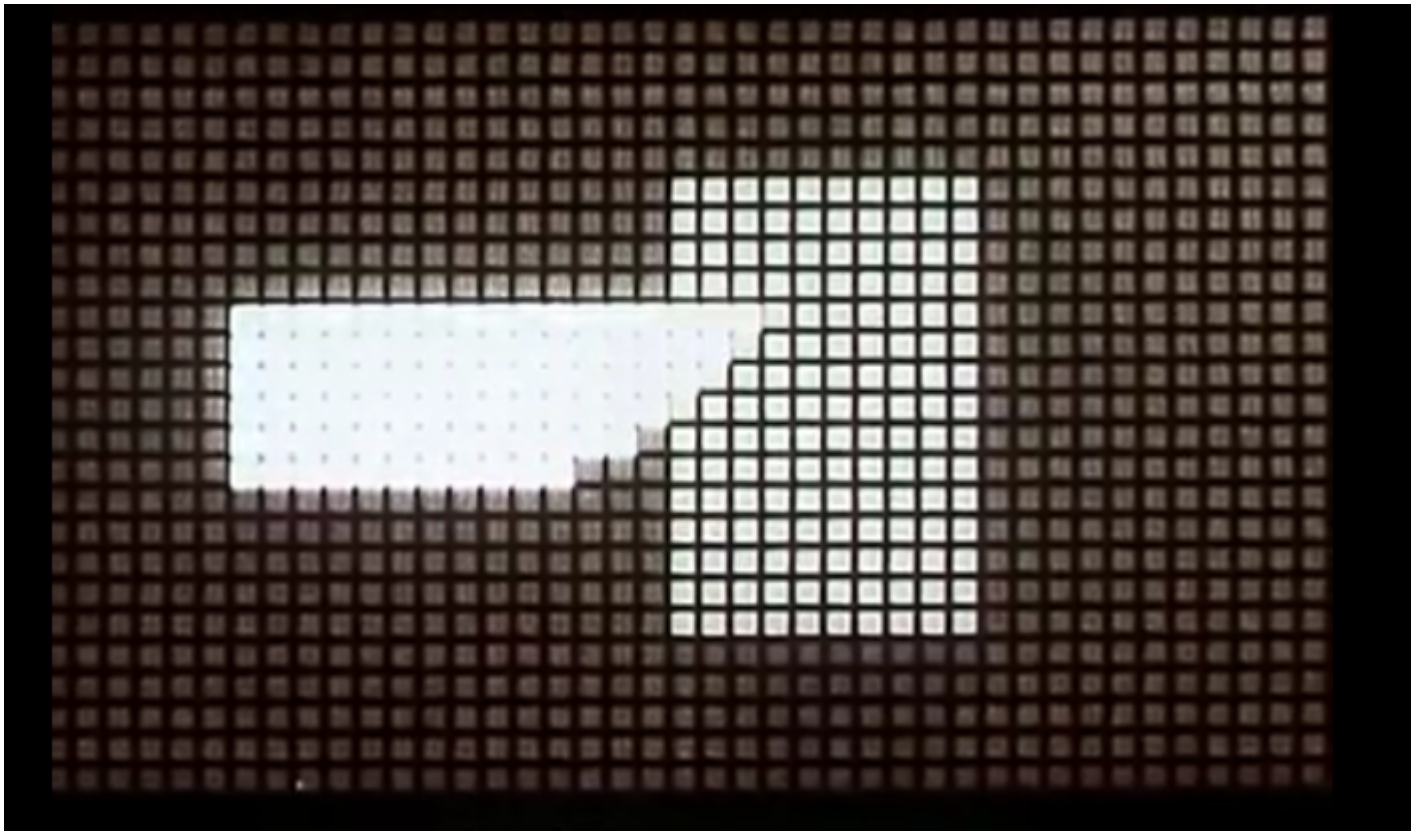
Within four adjacent squares,
each 4' by 4',
four draftsmen will be employed
at $4.00/hour
for four hours a day
and for four days to draw straight lines
4 inches long
using four different colored pencils;
9H black, red, yellow and blue.
Each draftsmen will use the same color throughout
the four day period,
working on a different square each day.

Sol LeWitt (1970)

50-60s cont.

meanwhile the engineers...

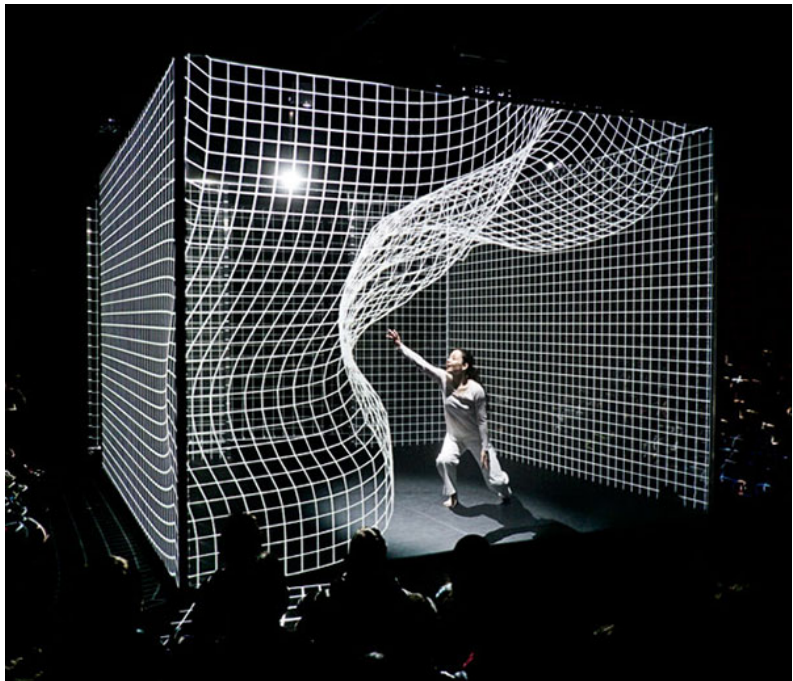
BEFLIX (Bell Labs) – early computer graphics



80s - today

code + art

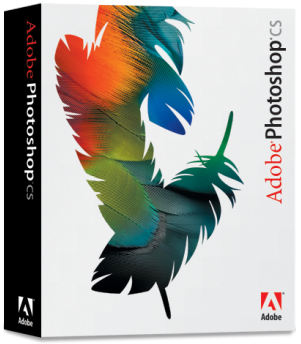
- proliferation of the personal computer
- programming reaches wider audience
- Lingo was developed
  - first language used by designers and artists up to early 90's
- code is used today not just for controlling pixels
  - controls motor parts of installations
  - architecture
  - decision making + smart interaction
  - design



why code?

Tradition of artists creating their own tools as old as art itself.

Software revolution and the “Photoshop effect”



From paint to code

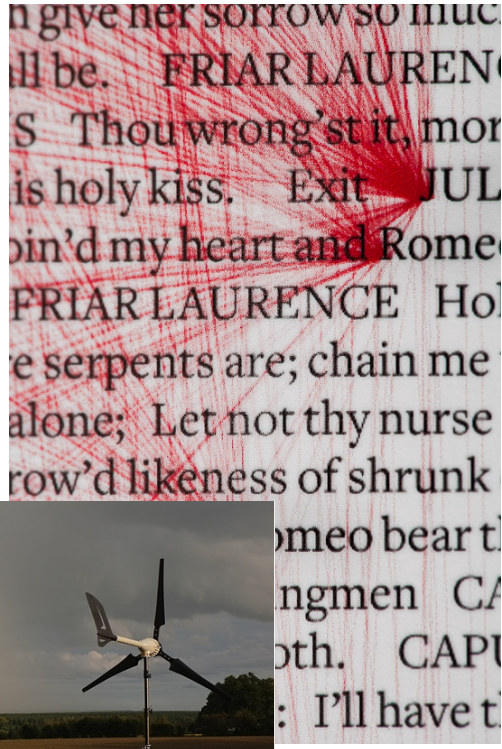
code + art

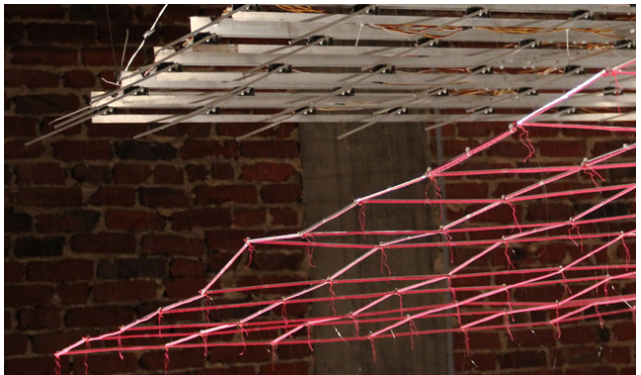
conclusions

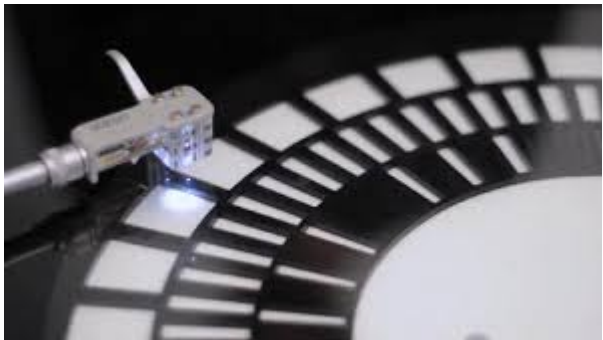
- observer vs. observed
- new modes of expression
- non-linear narrative
- New ways of I/O
- exceed limits of proprietary packages
- gateway to new media
- computer becomes a medium, not a tool

programming + art

examples

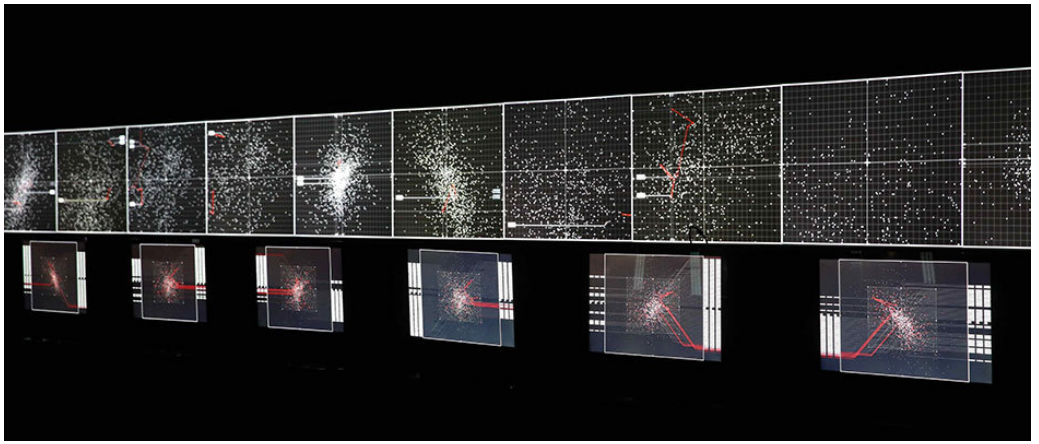






programming + art

examples (cont)



Your examples:

https://goo.gl/tQm475

Enroll to class now!

https://goo.gl/Z6jR5o
enrollment key: **linux**

Lab activity: Meet & Greet

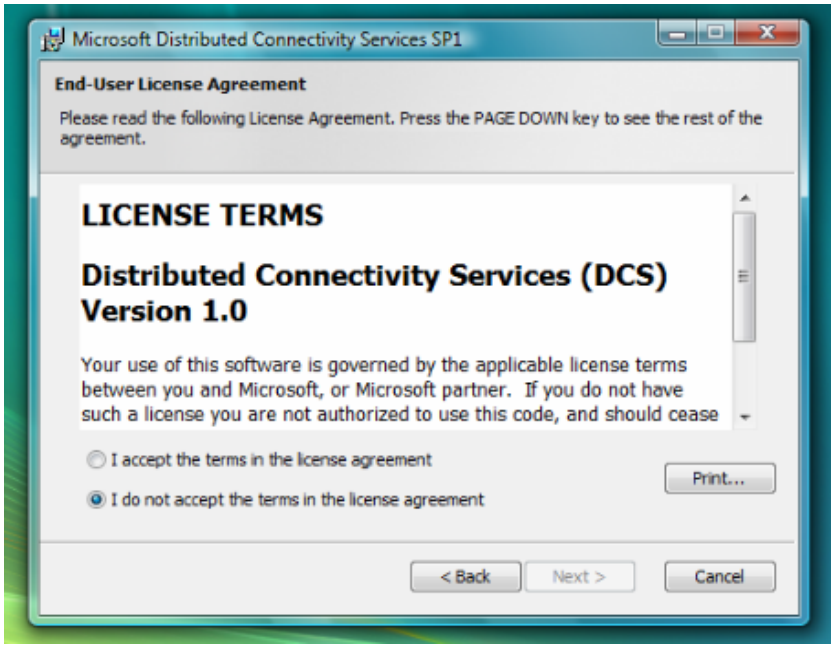
a bit more history...

...a bit more history

- Up until the late 70's users were able to study the source code of software, copy it and exchange it with each other
- But things were beginning to change since the end of the 60's:
  - increase in software costs
  - many users were asking for support
  - many users wanted to buy software separately
  - the software industry started competing with the hardware industry

a change...

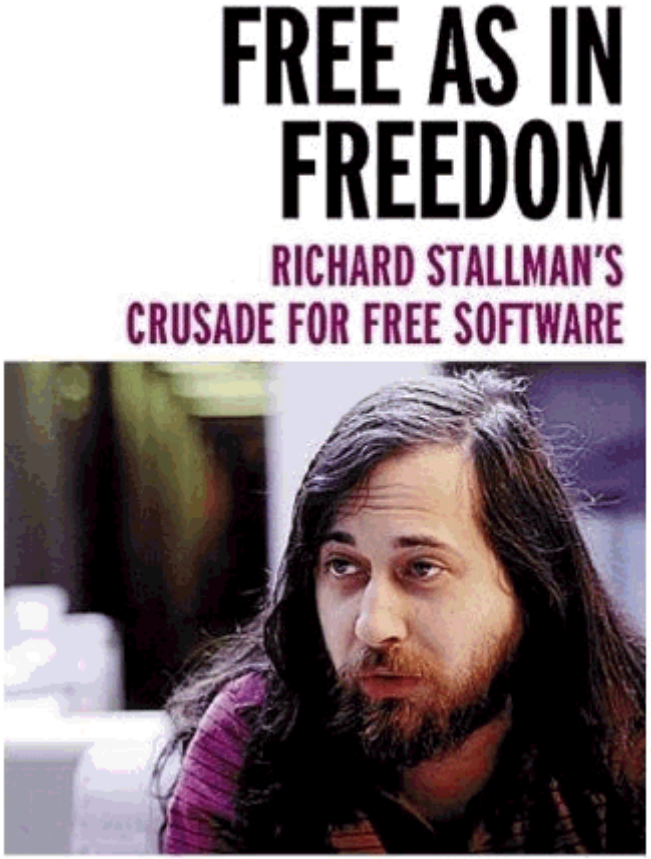
- In the late 70's companies started distributing binary copies of their software
- Laws on intellectual property were extended to software
  - a software was protected just like a song or novel
  - they started selling software licenses and not the actual programs




Richard Stallman

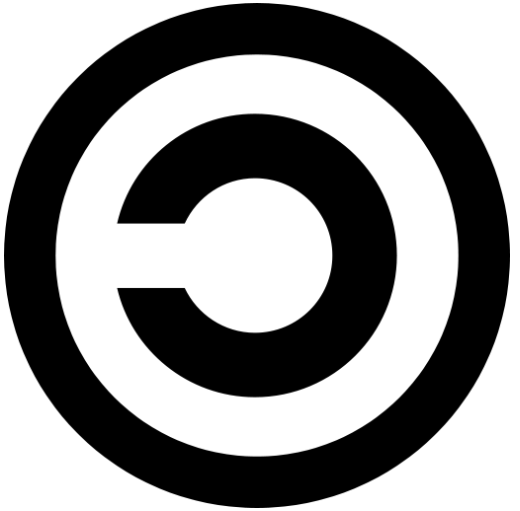
free software
the basic freedoms

- **Freedom 0**: The freedom to run the program, for any purpose.
- **Freedom 1:** The freedom to study how the program works, and change it so it does your computing as you wish. Access to the source code is a precondition for this.
- **Freedom 2:** The freedom to redistribute copies so you can help your neighbor
- **Freedom 3:** The freedom to distribute copies of your modified versions to others. By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.



GNU General Public License

- in 1989 publishes the GPL software license in which he also introduces *Copyleft*
- copyleft grants the freedom to use, modify and redistribute a intellectual creation...
- under the condition that every copy or modified work will be distributed with the same license (ie granting the same freedoms)
- Different from public domain



free vs. open source

- open source: way of developing software
- free software: social movement

Advantages of free software

- better code / better performace
- ability to use on a infinite number of machines
- free upgrades
- user not tied to a specific company / provider
- reduced risk for malware / viruses (see NASA)
- security (see NSA)
- ability to change the source code to fit your needs
- reduced costs

requirements

- Windows 8, hardware requirements

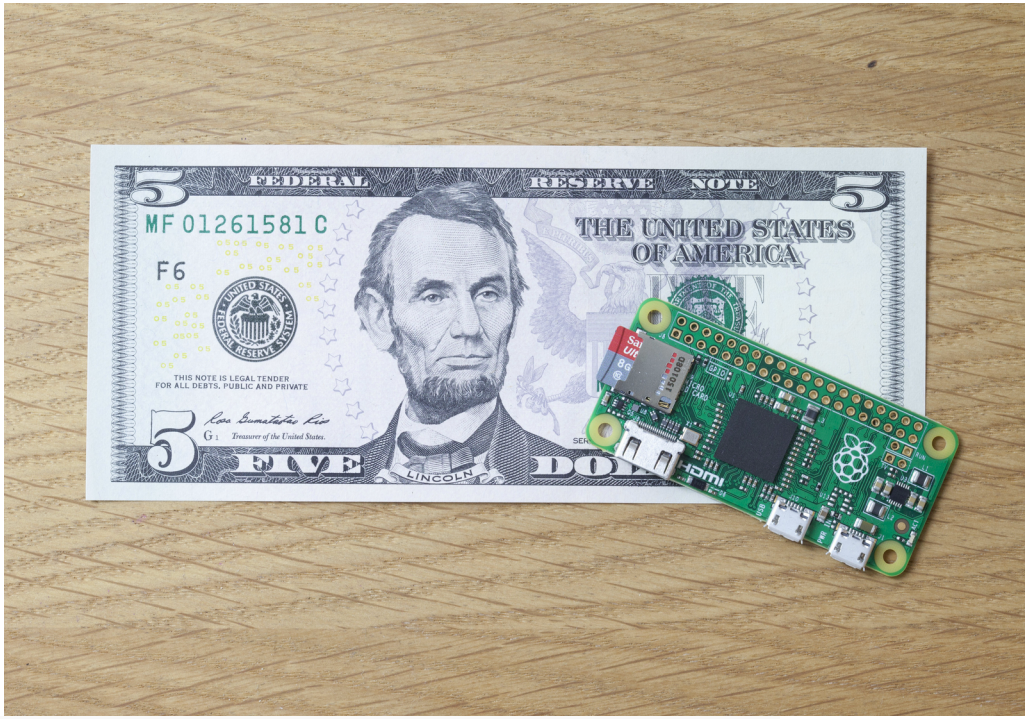
Architecture	32-bit	64-bit
Processor	1 GHz IA-32 processor	1 GHz x86-64 processor
Memory (RAM)	1 GB	2 GB
Graphics card	DirectX 9 graphics processor with WDDM driver model 1.0 (Not absolutely necessary; only required for Aero)	
HDD free space	16 GB of free disk space	20 GB of free disk space
Optical drive	DVD-ROM drive[121] (Only to install from DVD-ROM media)	

- Ubuntu 11.10, hardware requirements

Current Minimum Requirements[36][37][38][39]	Server	Desktop
Processor (x86) with the i686 instruction set[40]	300 MHz	700 MHz
Memory (RAM)	128 MiB	384 MiB
Hard Drive (free space)	1 GB	5 GB
Monitor Resolution	640×480	1024×768

...it can run on very cheap hardware

- raspberry pi (raspberrypi.org) – cost £4



Raspberry Pi

What GNU/Linux should I install?

- Get https://linuxmint.com (cinnamon edition)
- It can be installed parallel to Windows (dual boot)
- It will make your machine a lot faster
- You'll have a lot more tools available
- You need to burn the ISO on a usb stick or DVD and boot from it. Then follow screen instructions

Learning the GNU way

- We share code (with appropriate citation)
- We study other people's code!
- We work in pairs in class
- We teach each other how to use tools
- Classes are arranged for collaboration
- Hatchlab vs. studio space

Sharing
but with conditions

OpenFrameworks

Free Software
Free Society

Lab assignment:

- openFrameworks troubleshooting
- compile the emptyExmple

A final note...

The impostor syndrome