Learning to code

nature COMMUNICATIONS

Article | Open

26 October 2018

sequence-defined
for chemical data

, Bart Devreese, Peter Dawyndt & Filip Du Prez

WETENSCHAP

Vergeet de USB-stick of de harde schijf: zetten we onze vakantiefoto's en -filmpjes binnenkort op poeder?

Wetenschappers van de universiteit van Gent hebben een manier ontwikkeld om informatie - denk aan teksten, foto's en filmpjes - op te slaan in de vorm van poeder. Dat moet een alternatief bieden voor de USB-sticks, harde schijven en servers die we nu gebruiken en in de toekomst wellicht niet meer zullen volstaan. "QR-codes omzetten in poeder, dat lukt al. Voor YouTube-video's is het nog wat te vroeg."

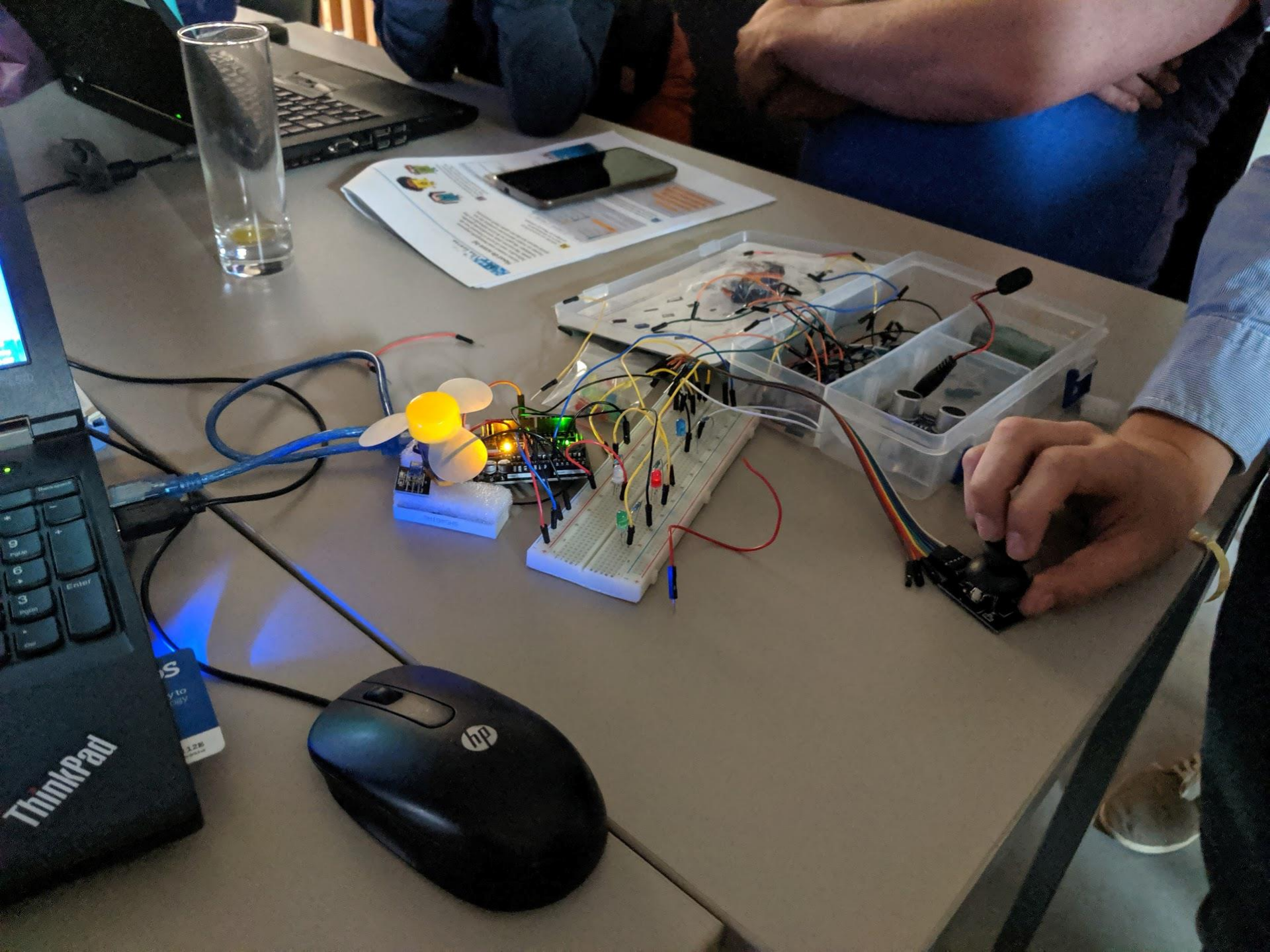
ning to code

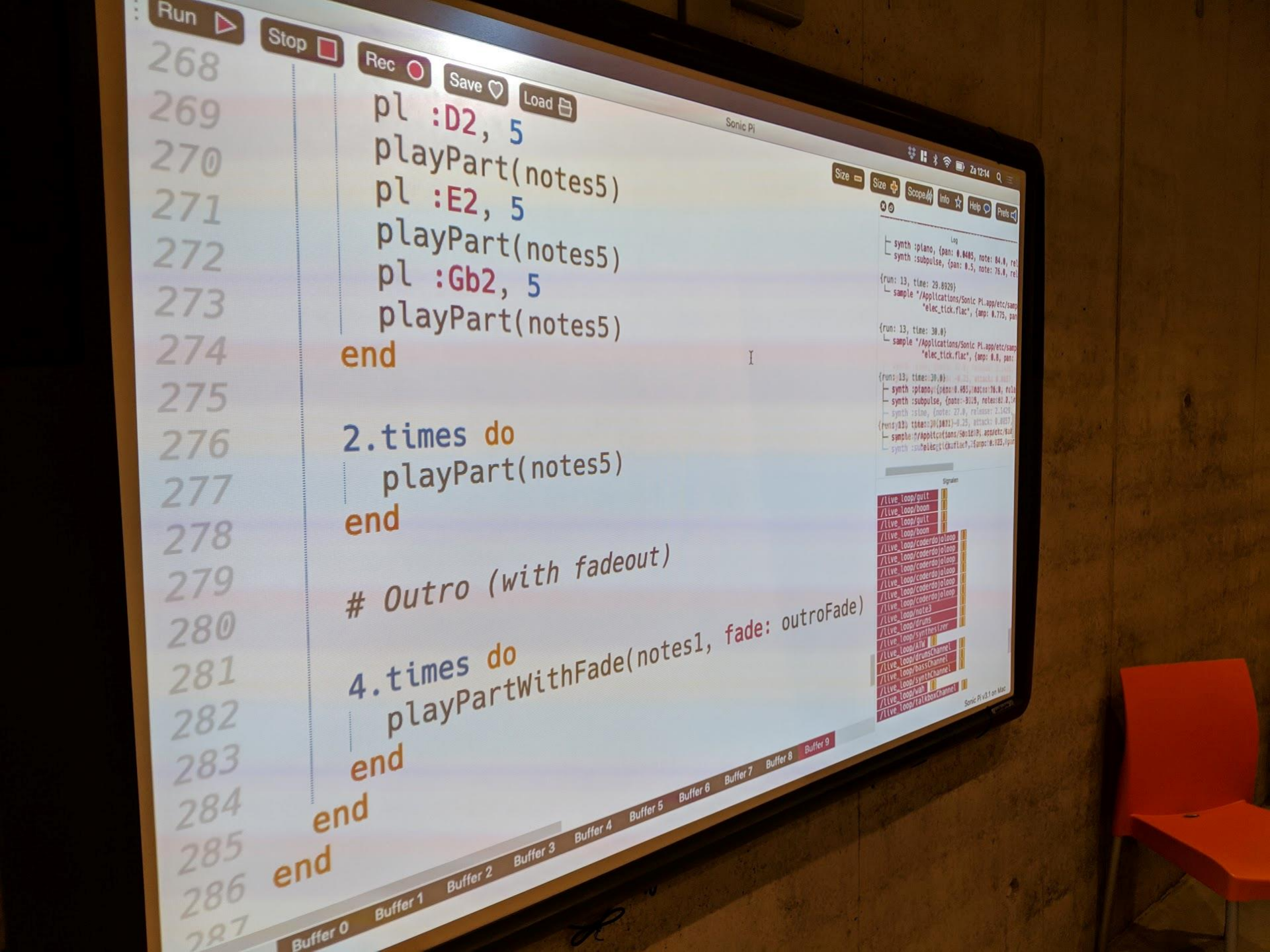
Learning to code

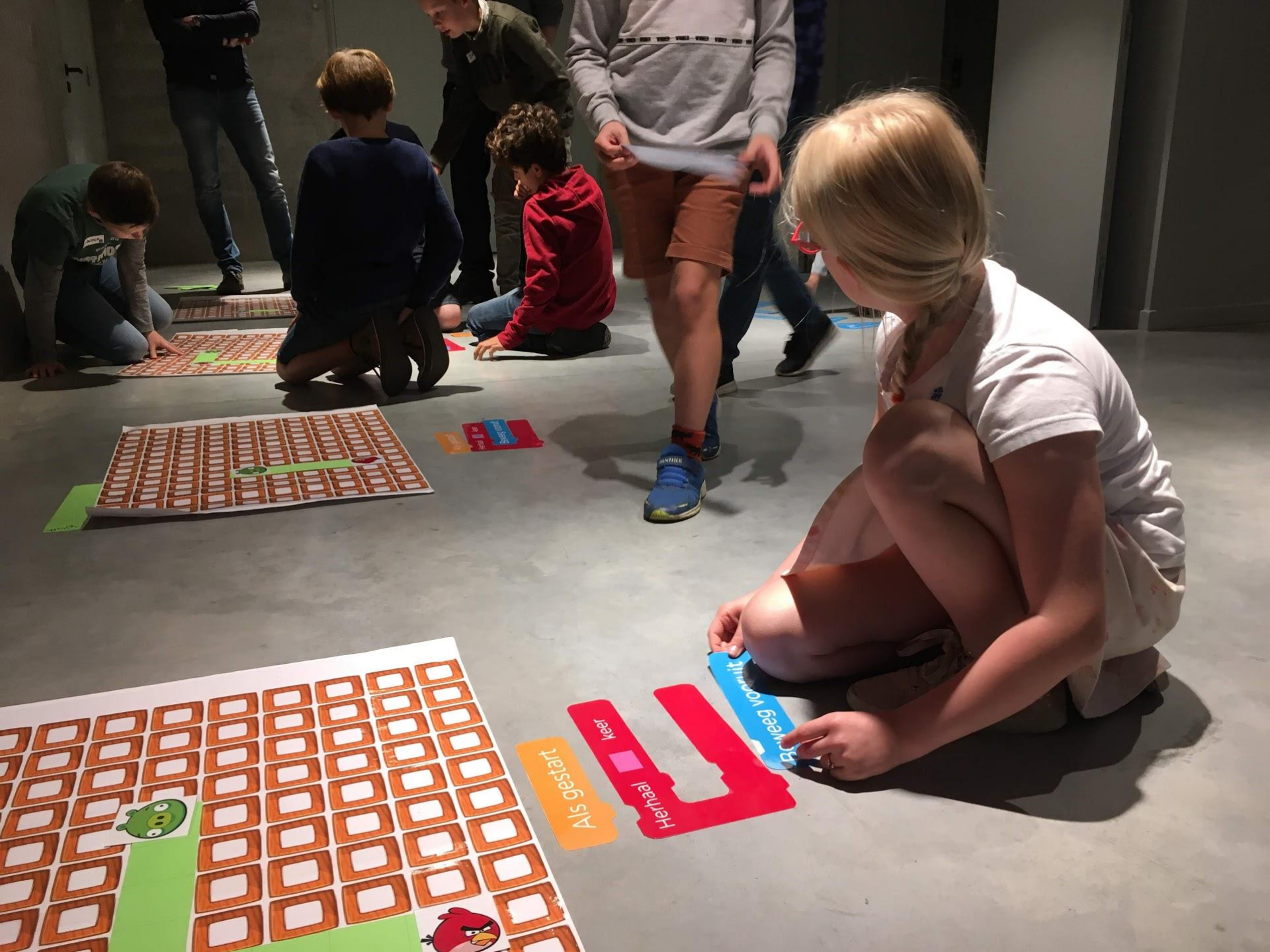
in a world full of paradoxes


CoderDojo
Welkom ninja's!


Ouders


ThinkPad
hp


Run
Stop
Rec
Save
Load
Sonic Pi
Size
Scope
Info
Help
Prefs
pl :D2, 5
playPart(notes5)
pl :E2, 5
playPart(notes5)
pl :Gb2, 5
playPart(notes5)
end
2.times do
playPart(notes5)
end
Outro (with fadeout)
4.times do
playPartWithFade(notes1, fade: outroFade)
end
end
end
Buffer 0
Buffer 1
Buffer 2
Buffer 3
Buffer 4
Buffer 5
Buffer 6
Buffer 7
Buffer 8
Buffer 9


Als gestart
keer
Herhaal


Coderdojo Belgium
Have all. Be cool!


Chrome
Archief
Bewerken
Weergave
Geschiedenis
Bladwijzers
Personen
Venster
Help

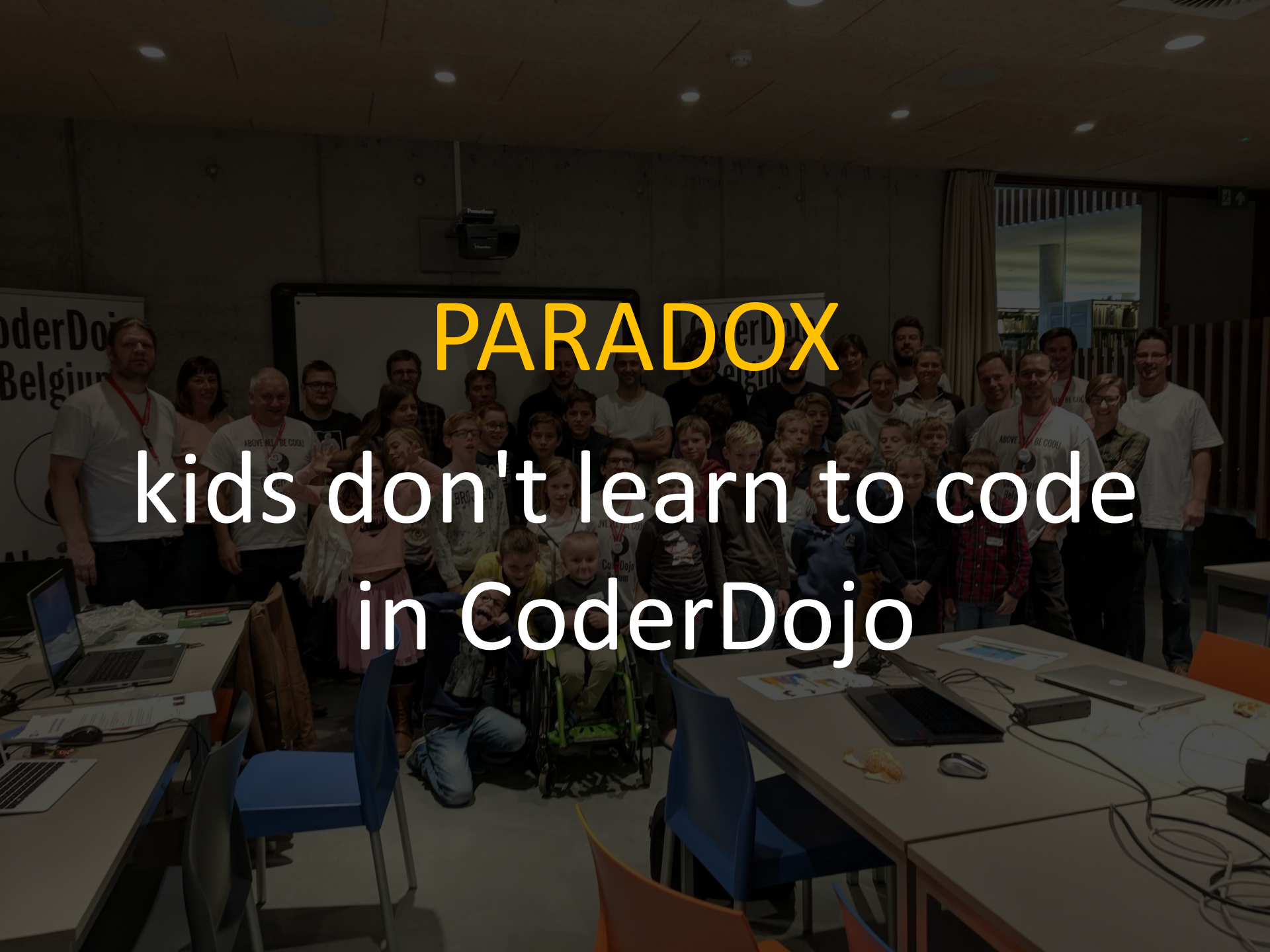

mBot

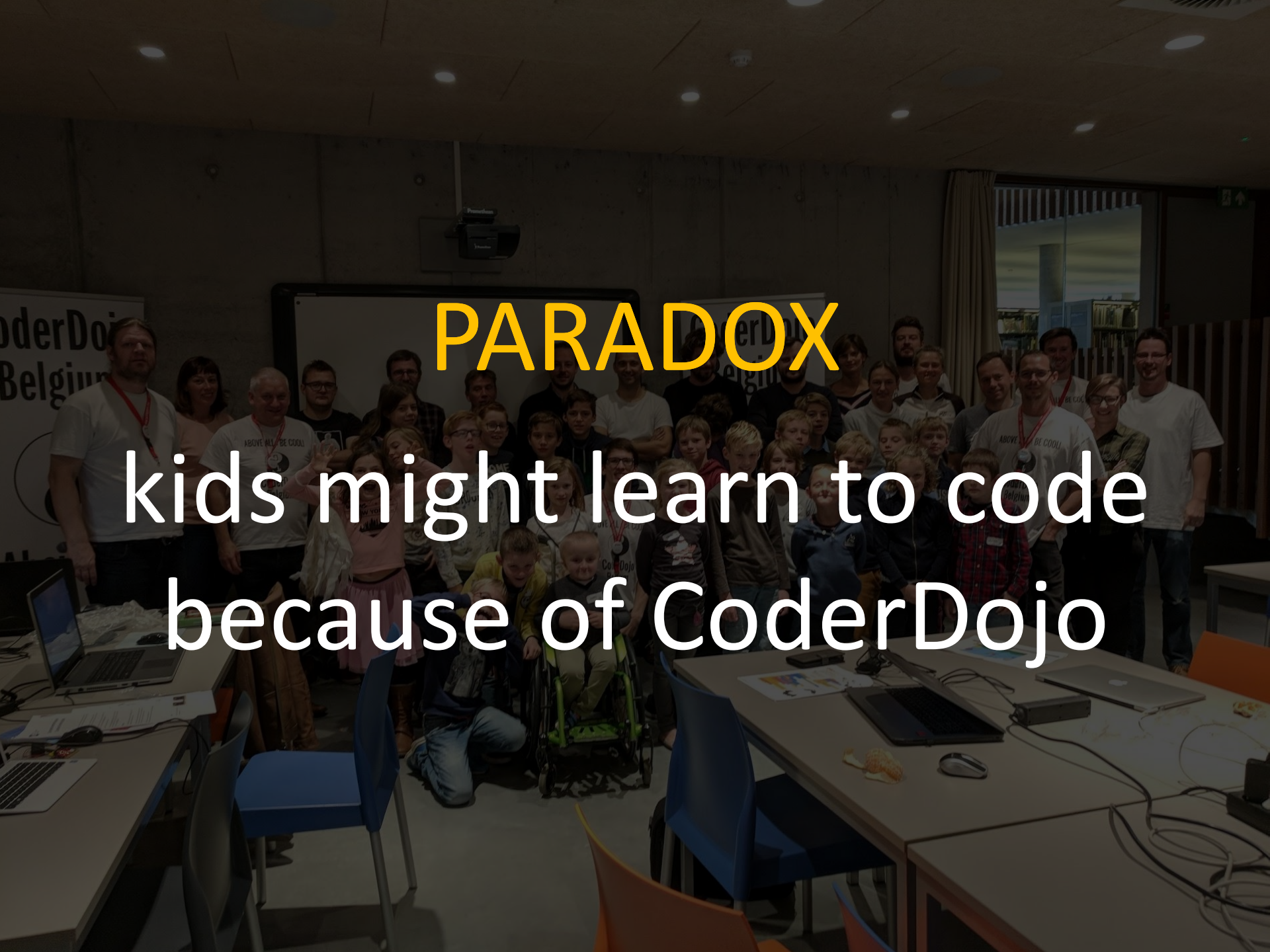

CoderDojo
4 Divas
#codingforgirls


COOLEST
2019
PROJECTS

CliniCoders

DojoCon 2018


PARADOX
kids don't learn to code
in CoderDojo


PARADOX
kids might learn to code
because of CoderDojo


challenges for learning
to code in schools

Programming is hard

Teach Yourself Programming in Ten Years

Peter Norvig

Why is everyone in such a rush?

Walk into any bookstore, and you'll see how to *Teach Yourself Java in 24 Hours* alongside endless variations offering to teach C, SQL, Ruby, Algorithms, and so on in a few days or hours. The Amazon advanced search for [title: teach, yourself, hours, since: 2000 and found 512 such books. Of the top ten, nine are programming books (the other is about bookkeeping). Similar results come from replacing "teach yourself" with "learn" or "hours" with "days."

The conclusion is that either people are in a big rush to learn about programming, or that programming is somehow fabulously easier to learn than anything else. Felleisen *et al.* give a nod to this trend in their book *How to Design Programs*, when they say "Bad programming is easy. *Idiots* can learn it in *21 days*, even if they are *dummies*." The Abtruse Goose comic also had their take.

Outliers

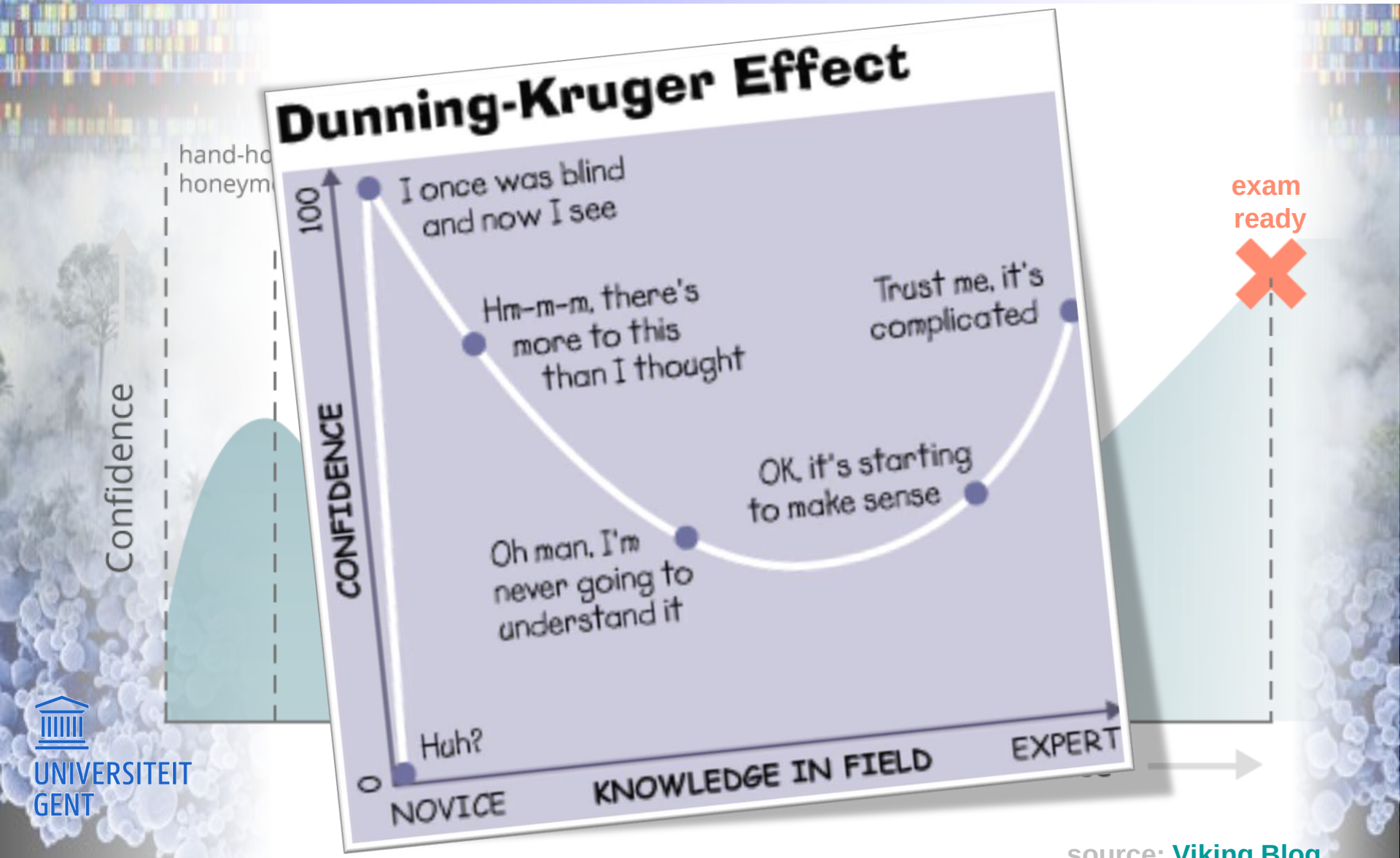
The Story of Success

Malcom
Gladwell

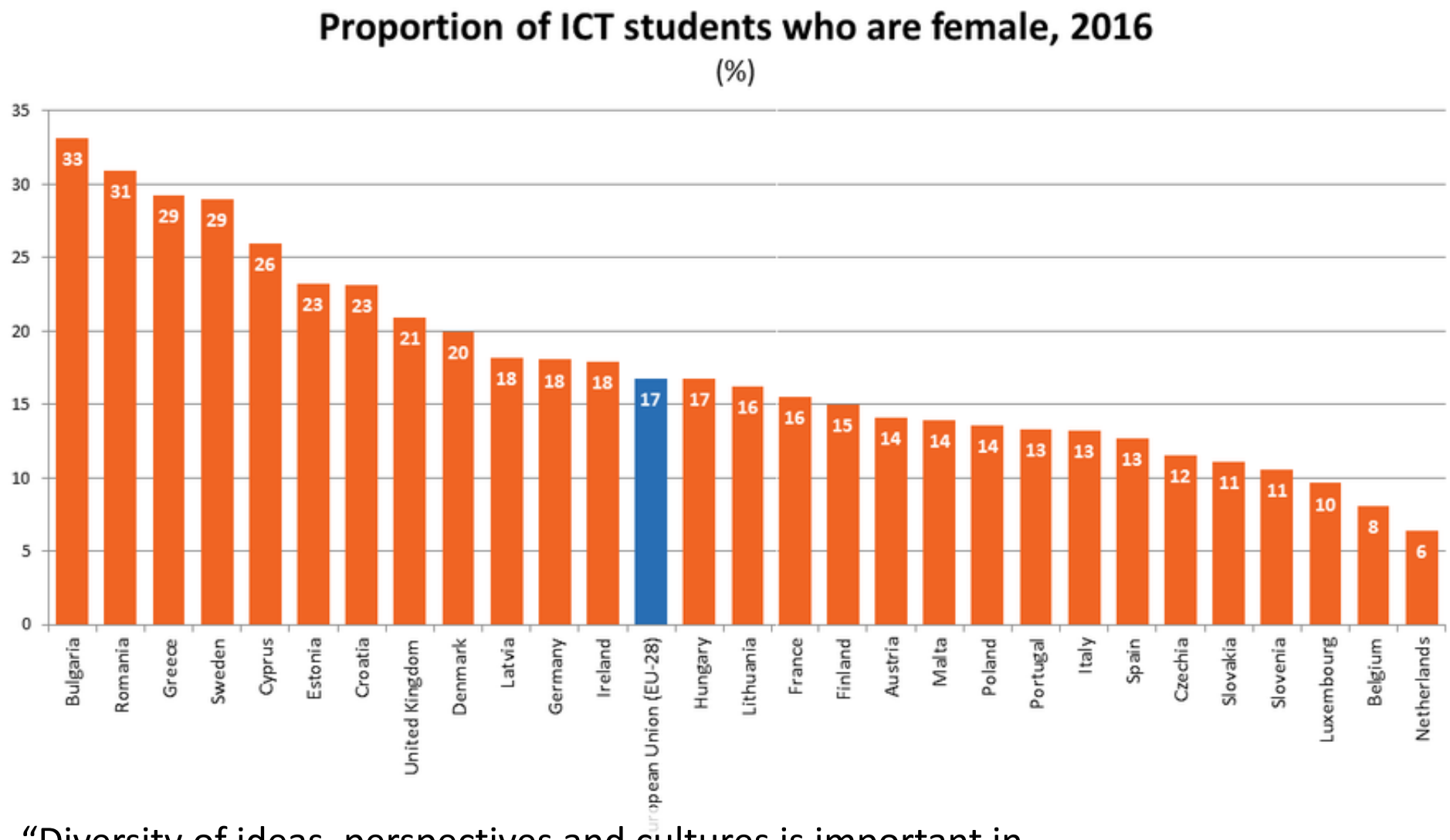
#1 bestselling author of *The Tipping Point* and *Blink*

Little, Brown and Company

Why learning to code is so damn hard

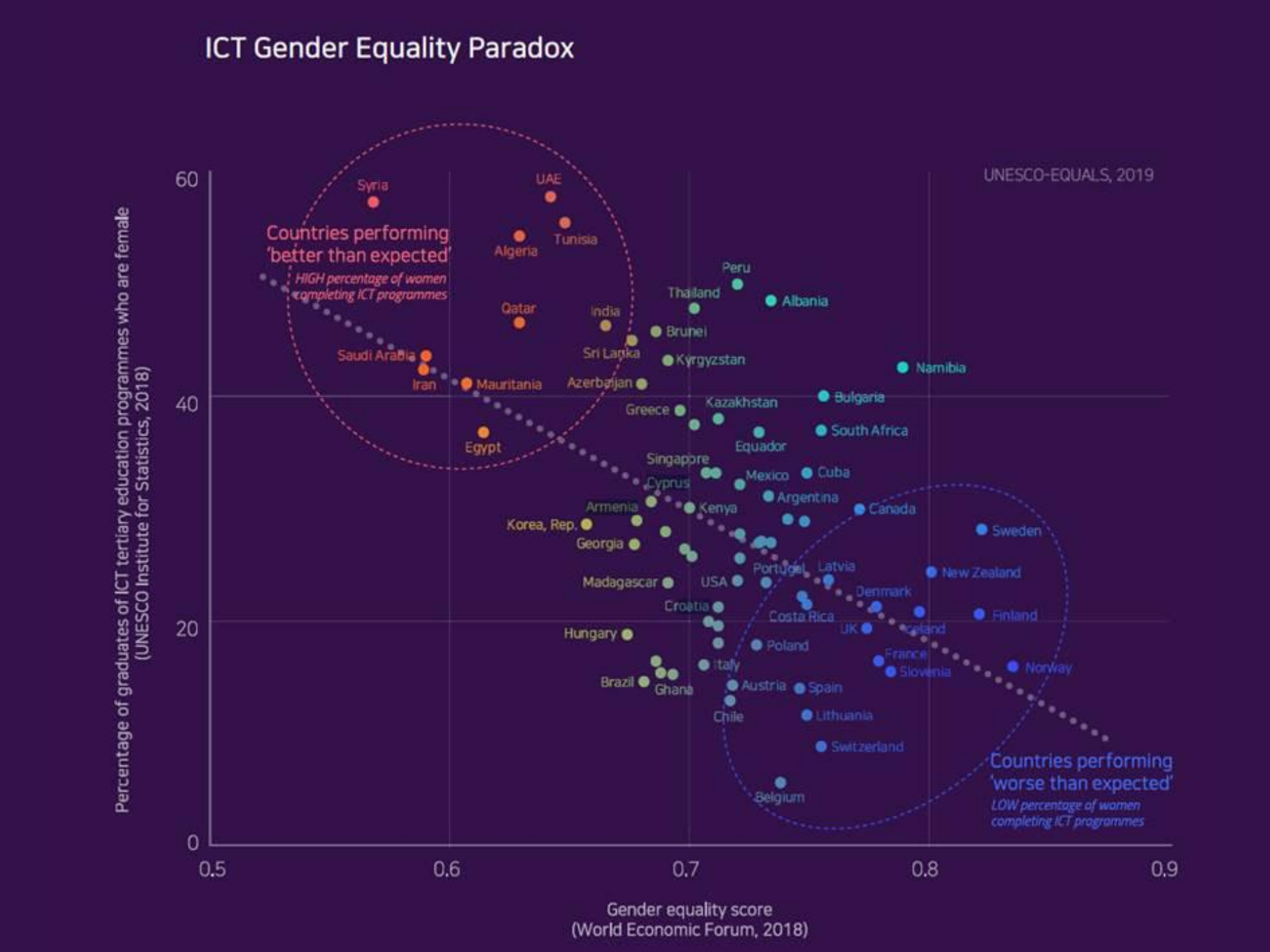


source: Viking Blog



"Diversity of ideas, perspectives and cultures is important in every domain, not just in computer science. If you ask me, computer science is too important to be left to men alone."

ec.europa.eu/eurostat


ICT Gender Equality Paradox
UNESCO-EQUALS, 2019
Percentage of graduates of ICT tertiary education programmes who are female
(UNESCO Institute for Statistics, 2018)
Gender equality score
(World Economic Forum, 2018)
60
40
20
0
0.5
0.6
0.7
0.8
0.9
Countries performing
'better than expected'
HIGH percentage of women
completing ICT programmes
Countries performing
'worse than expected'
LOW percentage of women
completing ICT programmes
Syria
UAE
Tunisia
Algeria
Qatar
India
Saudi Arabia
Iran
Mauritania
Egypt
Sri Lanka
Azerbaijan
Peru
Thailand
Albania
Brunei
Kyrgyzstan
Namibia
Bulgaria
Greece
Kazakhstan
South Africa
Equador
Singapore
Mexico
Cuba
Cyprus
Argentina
Armenia
Kenya
Canada
Korea, Rep.
Georgia
Sweden
Portugal
Latvia
New Zealand
Madagascar
USA
Denmark
Croatia
Costa Rica
Finland
UK
Hungary
Poland
France
Slovenia
Norway
Brazil
Ghana
Austria
Spain
Chile
Lithuania
Switzerland
Belgium

Design cycle

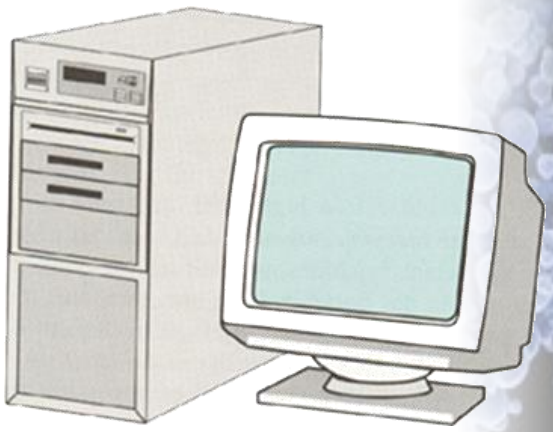
problem solving

natural language
(English, French, Arab, …)

programming language

programming language
(Python, Java, C++, …)

machine language
(Pentium, Xeon, AMD, …)



UNIVERSITEIT GENT

Design cycle

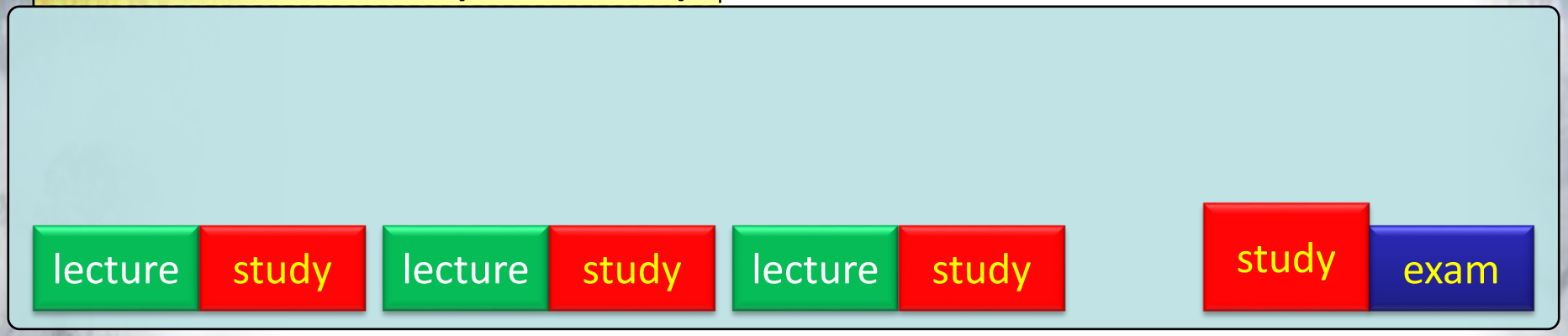
1. describe and analyze the problem
   - *what should the program do ?*
   - *what is the input, and what is the expected output ?*
2. design an *algorithm* (*pseudocode*)
   - *how to achieve the result ?*
3. convert algorithm into *source code*
   - following syntax rules of chosen programming language
4. compile source code into *machine language*
5. execute the program
6. trace potentional errors (*debugging*)

UNIVERSITEIT GENT


Ooooh! The last bug!

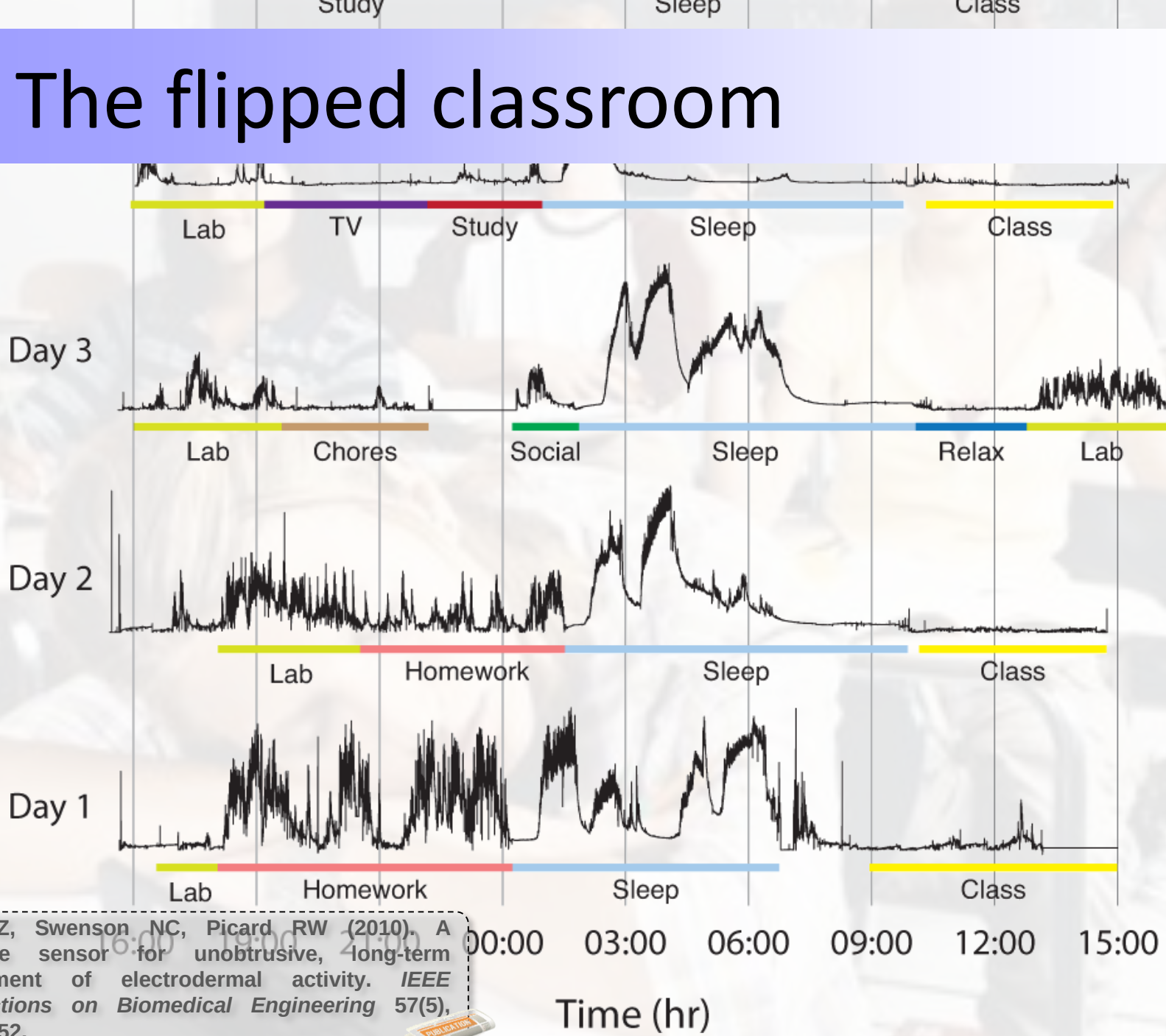
The flipped classroom

traditional classroom (ideal world)



UNIVERSITEIT GENT

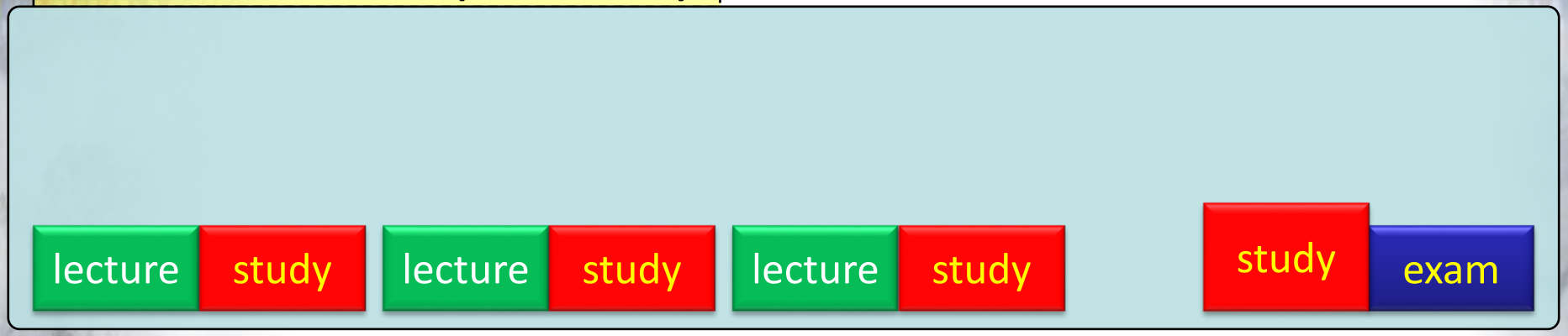
The flipped classroom



Poh MZ, Swenson NC, Picard RW (2010). A wearable sensor for unobtrusive, long-term assessment of electrodermal activity. *IEEE Transactions on Biomedical Engineering* 57(5), 1243-1252.

The flipped classroom

traditional classroom (ideal world)



UNIVERSITEIT GENT

The flipped classroom

traditional classroom (real world)

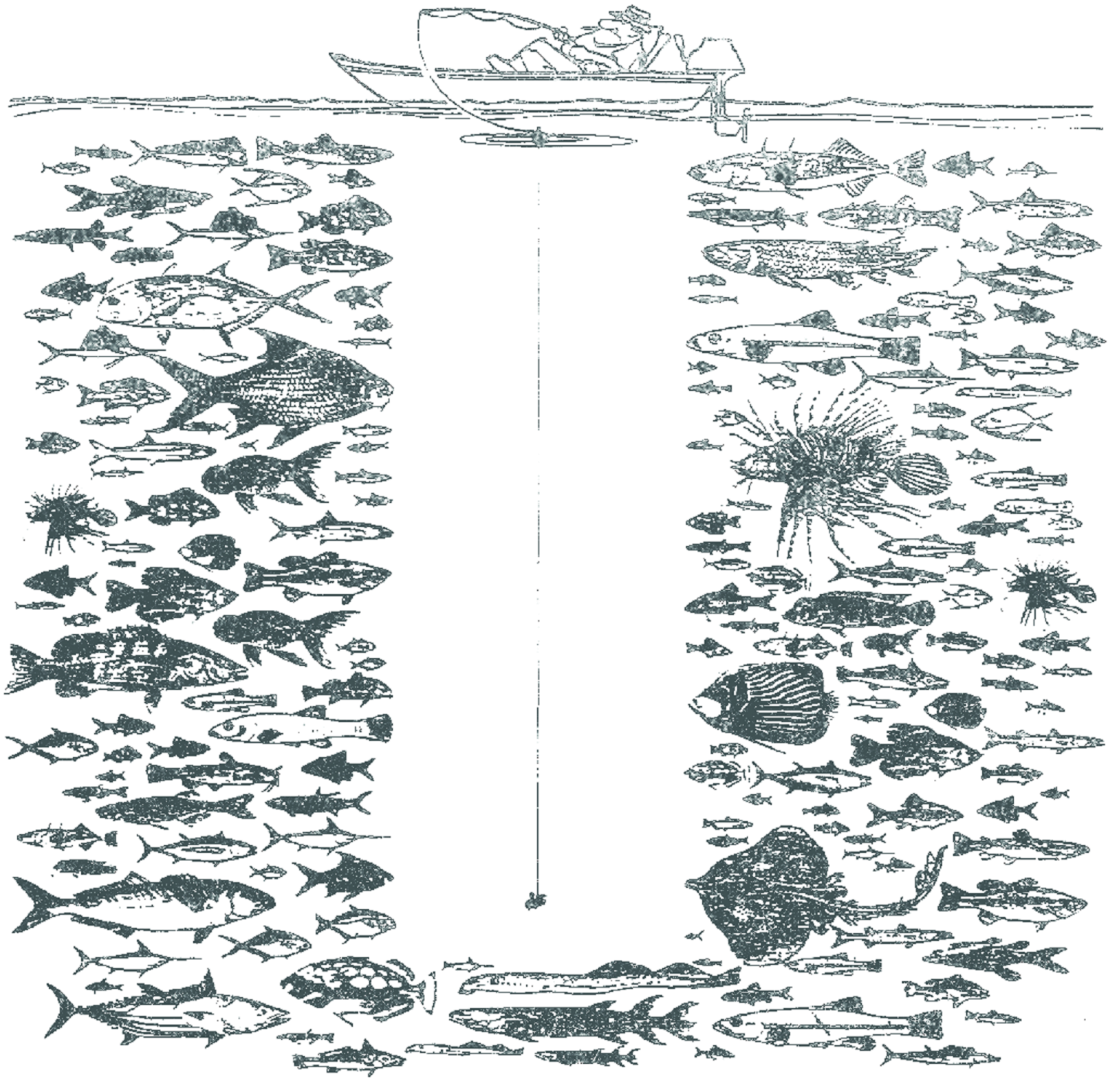
lecture | lecture | lecture | study | exam

flipped classroom

study | lecture | study | lecture | study | lecture | study | exam

Practice, practice, practice

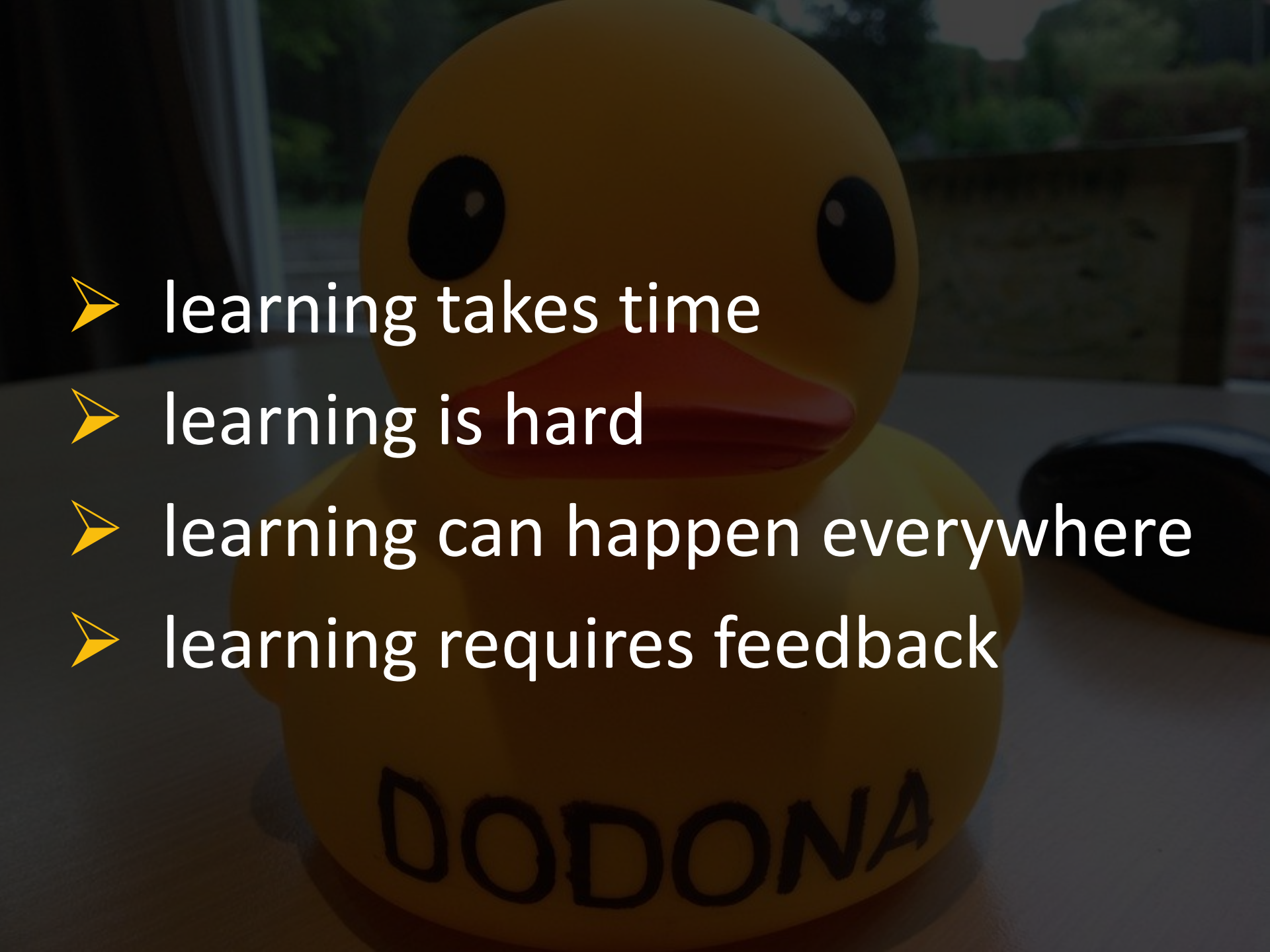
Programming = testing software



UNIVERSITEIT GENT

Feedback overload


dodona.ugent.be
@DodonaEdu
DODONA



- learning takes time
- learning is hard
- learning can happen everywhere
- learning requires feedback