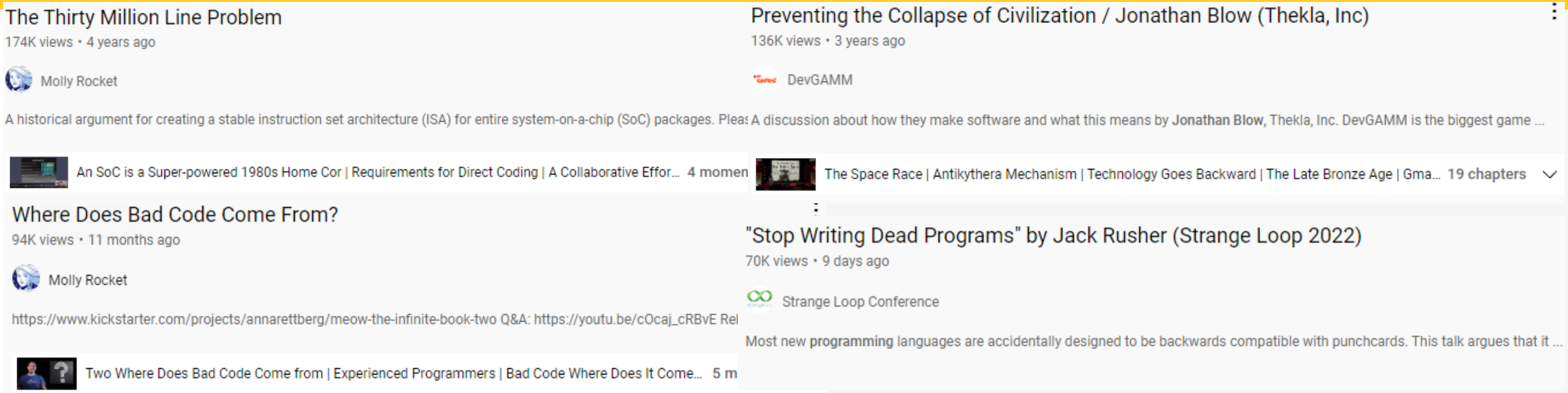
Non-linguistic Thinking as an Effective Tool for Innovation in Education

Leon Brânzan, Technical University of Moldova, https://fcim.utm.md/

ECCO INTERNATIONAL CONFERENCE
Chisinau, Moldova
October 20-21 2022

UNIVERSITATEA TEHNICĂ A MOLDOVEI
FCIM
· 1967 ·

Is software getting worse?



> “We are in the process of digging ourselves into an anachronism by preserving practices that have no rational basis beyond their historical roots in an earlier period of technological and theoretical development.”
>
> Seymour Papert, 1980

Is software getting worse?

- Are programmers getting worse?
- Are the tools getting worse?
- Are we teaching programming the wrong way?

How programming is taught


```
#include <stdio.h>

void main(int argc, char *argv[], char *envp[])
{
    printf("hello, world!");
}
```

```
.LC0:
  .string "hello, world!"
main:
pushq %rbp
movq %rsp, %rbp
subq $32, %rsp
movl %edi, -4(%rbp)
movq %rsi, -16(%rbp)
movq %rdx, -24(%rbp)
movl $.LC0, %edi
movl $0, %eax
call printf
nop
leave
ret
```

Why learning how to program is hard

- Programming languages should come last
- Text should be used less
- Specific skills and intuition should be developed early

Programming languages

$$3 + x = 5$$

Programming languages

Python → C → Assembly → Machine code

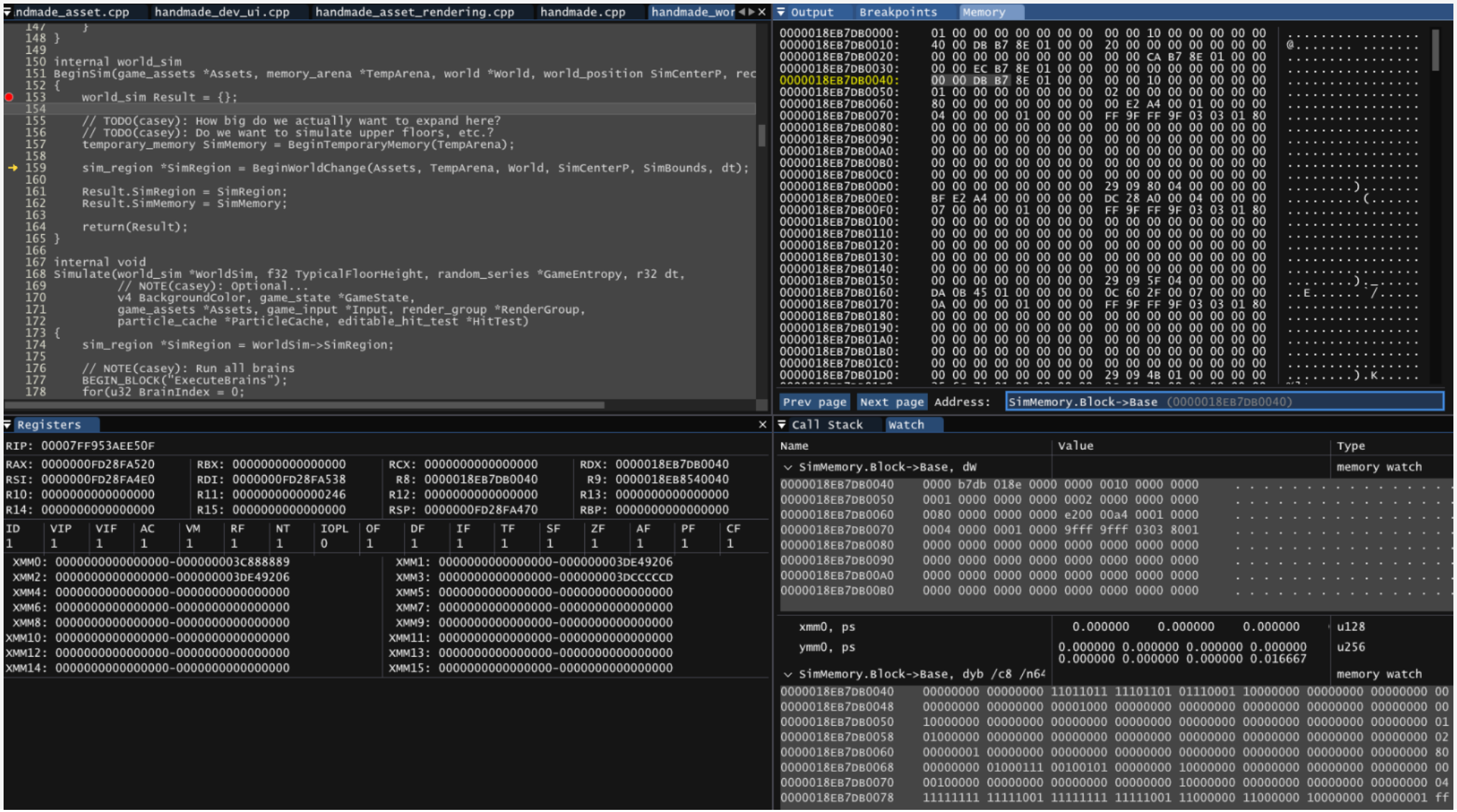
Data-oriented > Object-oriented > Functional > Structured

```
x = j if a > b else k
x = a > b → j, k
int x = (a > b) ? j : k;
```

Textual form

- Formal language =/= natural language
- Cognitive load theory
- Visual information processing vs. textual information processing
- Reasoning and language processing are done in different parts of the brain

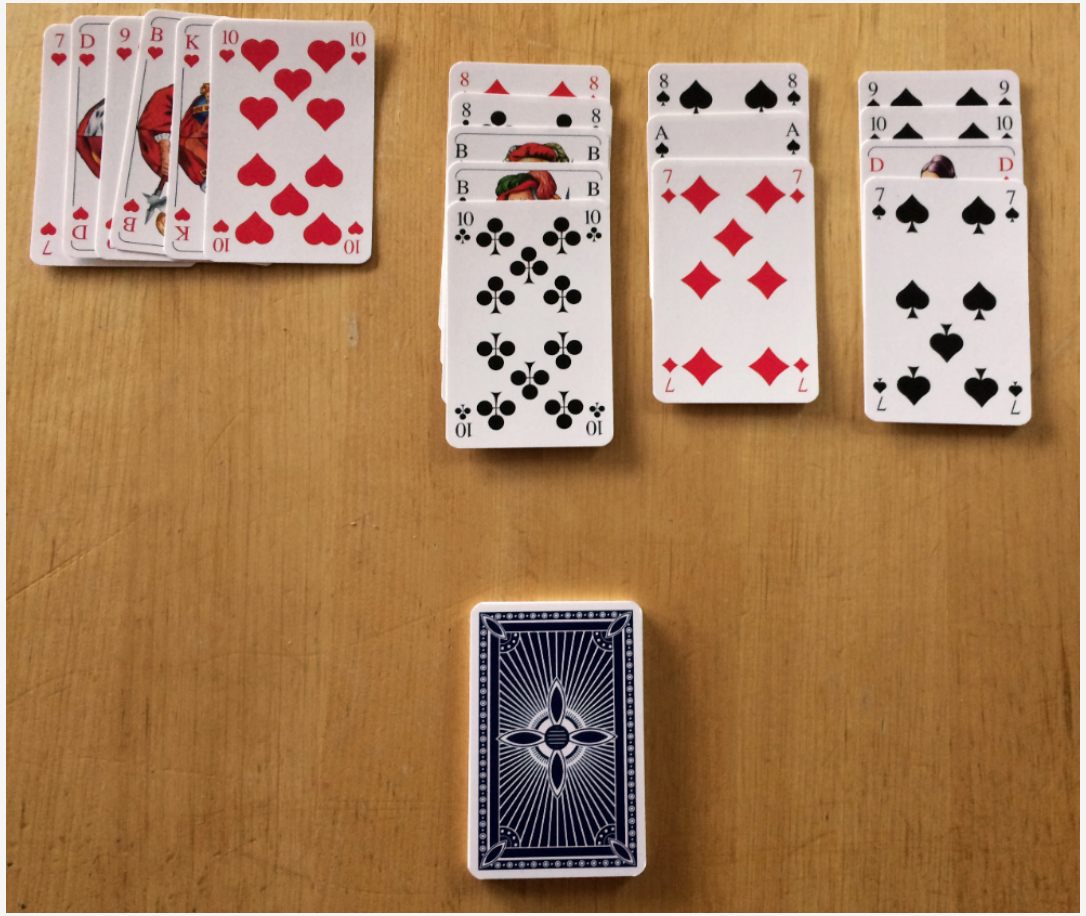
Specific set of skills



Solutions

- Guiding the learning process with visual tools to promote inference and reflection
- Choosing the right computational model
- Developing engineering thinking early by teaching fundamental concepts first

Inference and reflection

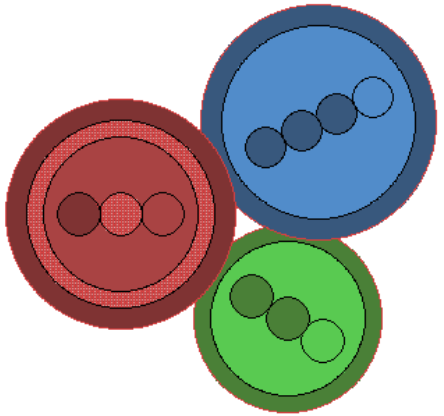


Computational models

Turing machine

Symbol	Write	Move
Blank	None	None
0	1	Left
1	0	Left

λ-calculus



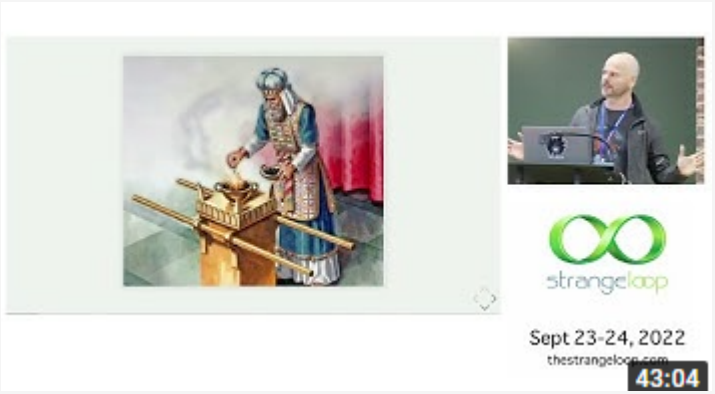
Fundamental concepts

Food for thought

Rethinking visual programming, Ivan Danyliuk

Video games and the future of education, Jonathan Blow

Programming with nothing, Tom Stuart



Stop writing dead programs, Jack Rusher

Presentation, links and references

https://bit.ly/case-for-non-linguistic-teaching