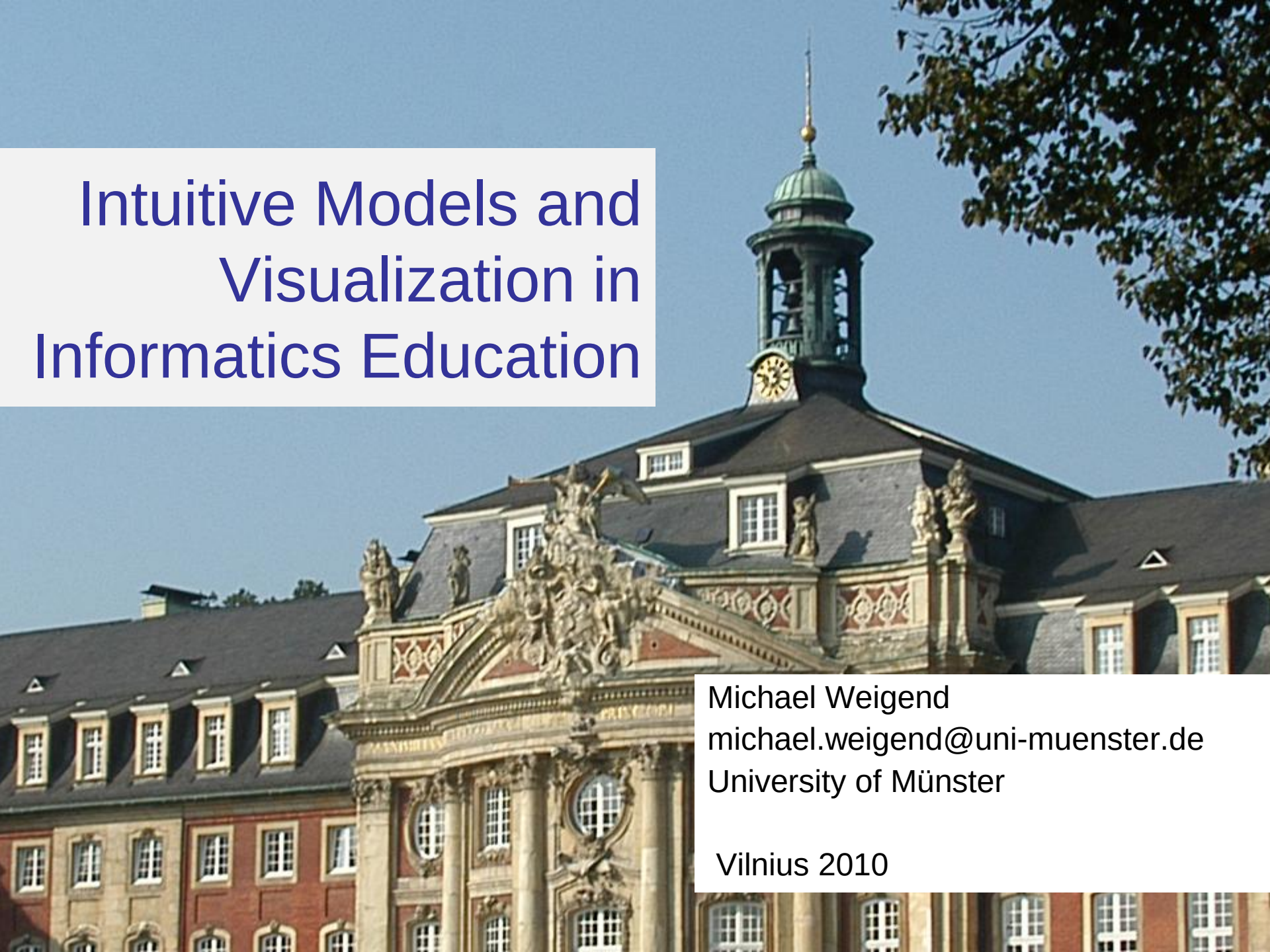
Intuitive Models and Visualization in Informatics Education

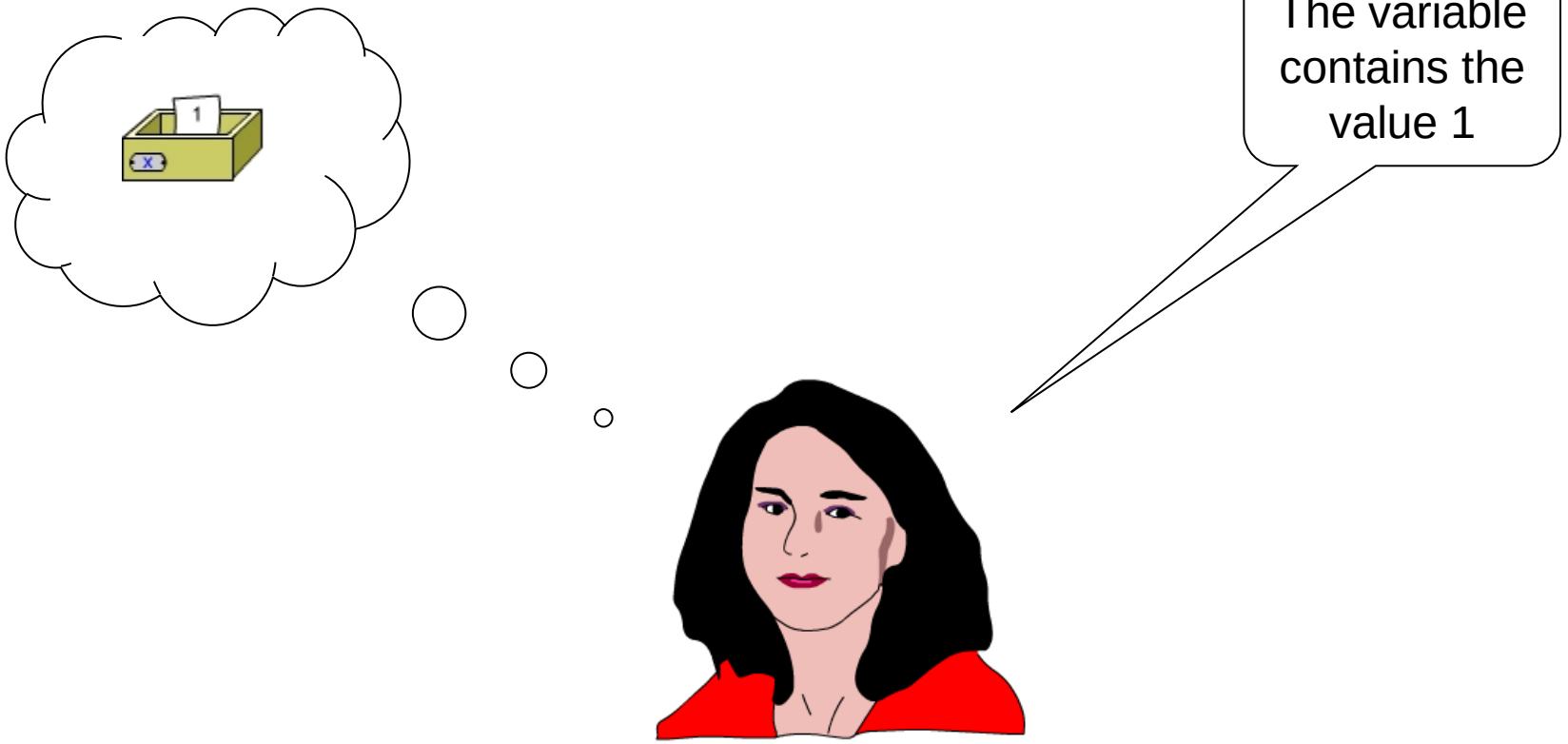


Michael Weigend
michael.weigend@uni-muenster.de
University of Münster

Vilnius 2010

Intuitive Model

```
x = 1
```



Outline

1 Intuitive model

2 Research methods

– Exercise –

3 Some basic Intuitive models in informatics

4 Intuitive Modeling (understanding programs by visualizing them)

5 Visualizations and classroom activities

1 Intuitive Model

Fischbein (1987): Intuition

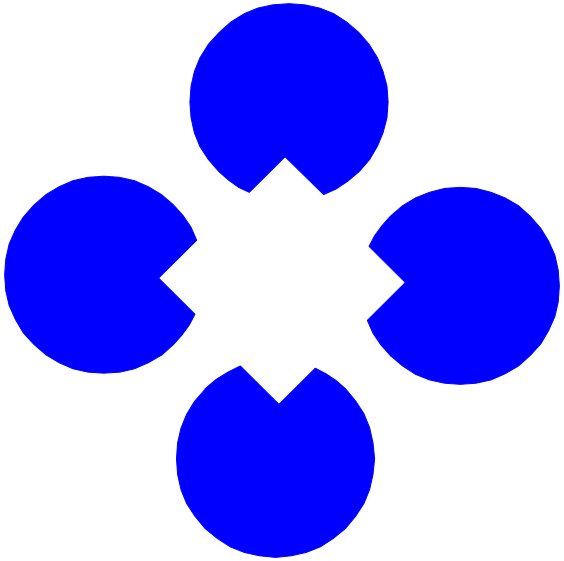
diSessa (2001): Phenomenological primitive (p-prim)

- Declarative knowledge
- Subjective certainty and self evidence
- Gestalt
- Persistency

Subjective Certainty

… based on experience in well known domains

Gestalt



Gestalt = a perceived hole entity

Wertheimer

"… this program works like hive of bees, going out for pollen and bringing it back to the hive"

Ron Jeffries: What is Extreme Programming? 2001

Certain knowledge about containers

You can put something into a container

The thing in the container still exists

When you search a thing search for the container, it is stored in

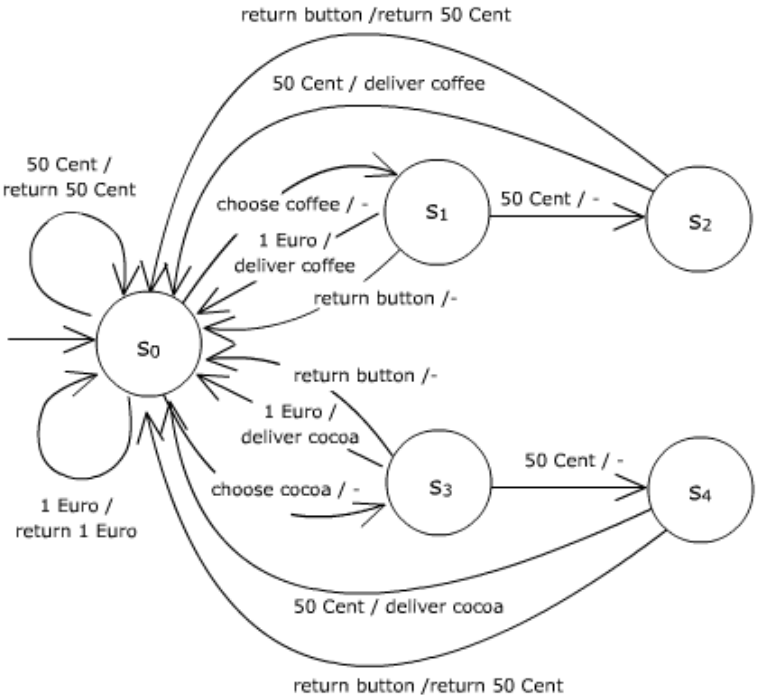
A thing remains in a container until someone takes it off

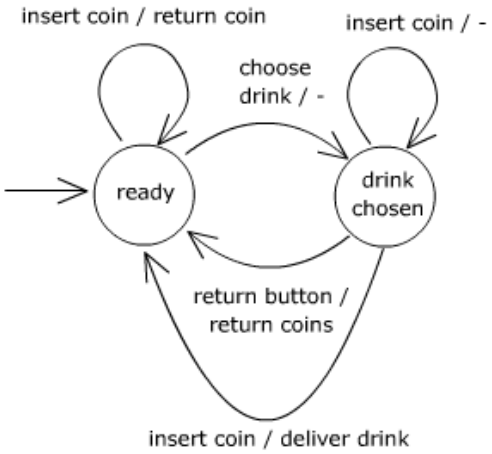
…

Piaget: object permanence

Simplicity and cognitive load

Was is the idea of a vending machine?





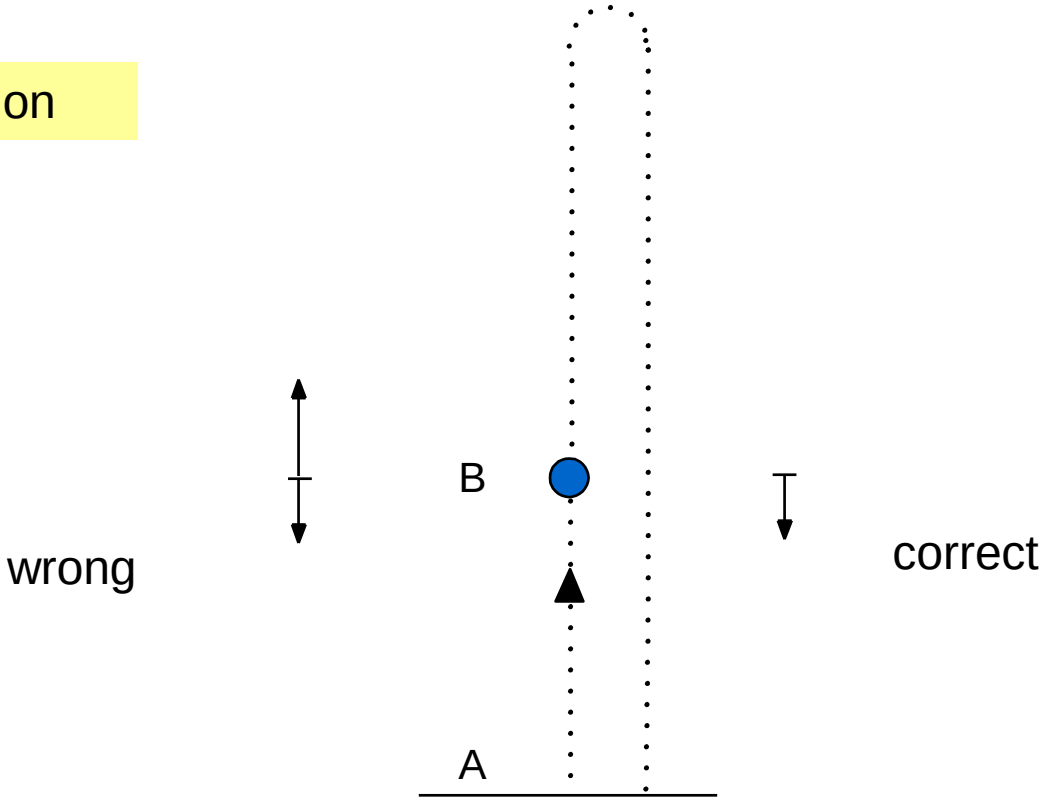
Only 5 chunks of information in working memory

Baddeley, Dehn 2008

Unconscious, compulsive

"Which forces are acting on the coin in point B"

Impetus intuition



Clement 1982

Persistency

Example:
Exhaustion of the container model for arrays (lists)

```
s = [1, 1, 1]
t = s
s[0] = 5
print t
```

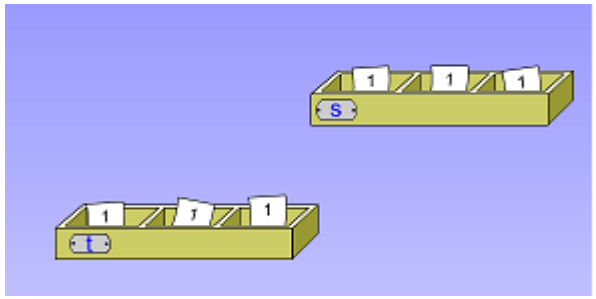
output

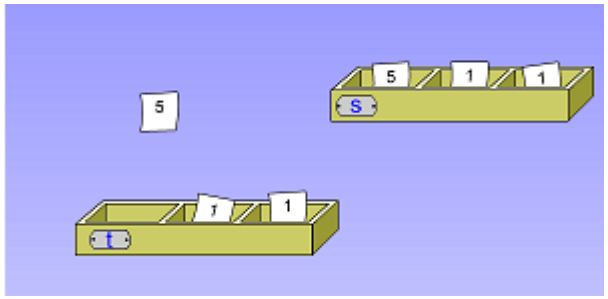
```
[5, 1, 1]
```

“Which model would you use to explain this?”

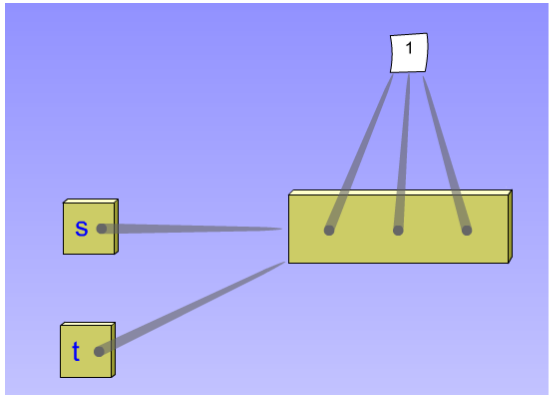
Exhaustion

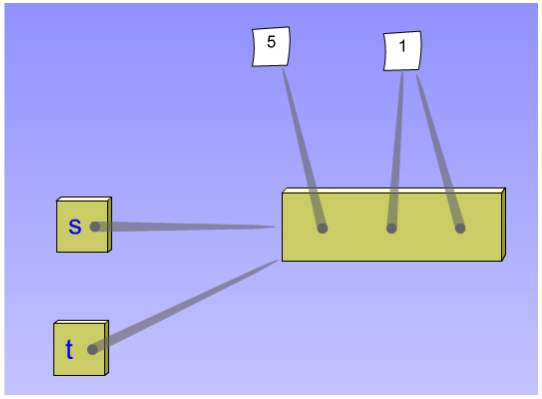
```
s = [1, 1, 1]
t = s
s[0] = 5
```





40 / 70





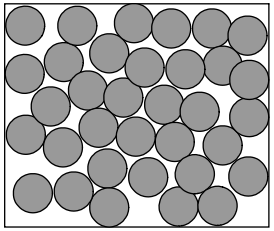
4 / 70

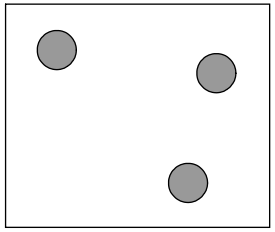
70 students (16 f 54 m), 2006

Modeling in natural science

reality

model



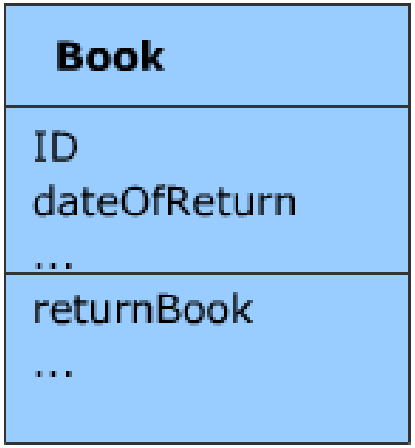


Goal: Structural feasability

Modeling in informatics (OOP)

Reality

Model



Goals: Efficiency, Controllability of complexity

Cultivated intuitive Models in Informatics

Stack

Queue

Dictionary

Scan Line

Loop

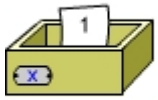
Switch (branch)

Data flow

“Message” in OOP

Three dimensions of representation

“contain”



The variable contains the value 1

Encoding

“carry”

```
[1, 2, 3]
[1, [2, 3], "a"]
gruppe = ["Tina", "Max"]
```

Metaphors

Exemplification

Conceptional Metaphors

- Not just rhetoric improvement of speech
- represent conceptional knowledge

2 + 3

Source domain ⟶ Target domain

“Arithmetic is moving objects”

Lakoff & Núnez (1997)

2 Research
How to find intuitive models?

- Student explains how a program works (“teach back”)
  van der Veer 1994
- Loud thinking while problem solving
- Interviews
- Analysis of programming mistakes

Teach back

“Explain …,

- How to search for a text document,
- How to display it on screen,
- How to save a copy
- How to print it.”

van der Veer 1994

Expressive means

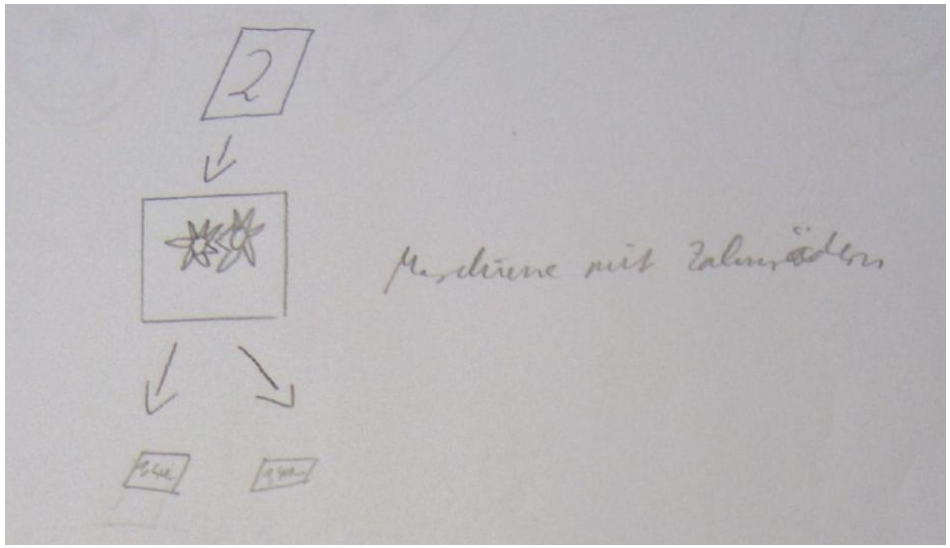
Text	91 %
pictures	25 %
Icons	28 %
rules	6 %
Programming code	34 %

607 student from three European countries

van der Veer 1994

Visualization exercises

```
double a;
a = Math.sqrt(2);
```



Students need to practise

Python Visual Sandbox (2006)

Idea:

Judge and use given animations
in different contexts

The Python Visual Sandbox

Idea:

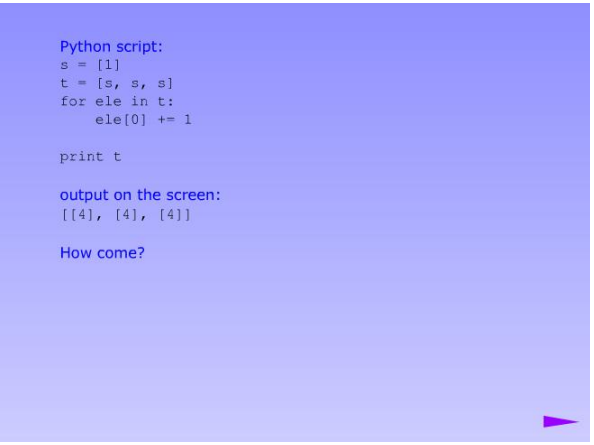
Judge and use
given animations
in different contexts

14 Workshops 2005 – 2006

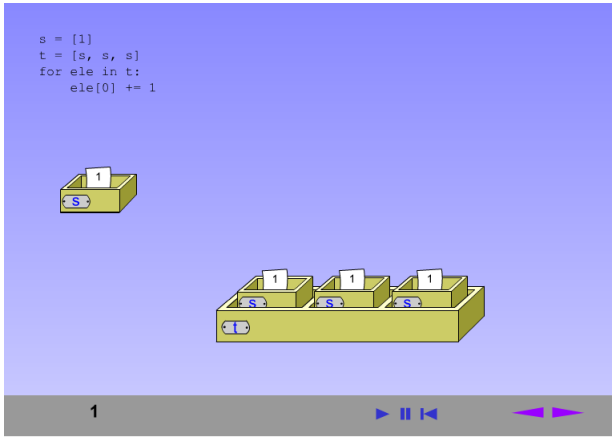
In Germany and
Hong Kong

International American Highschool Hong Kong

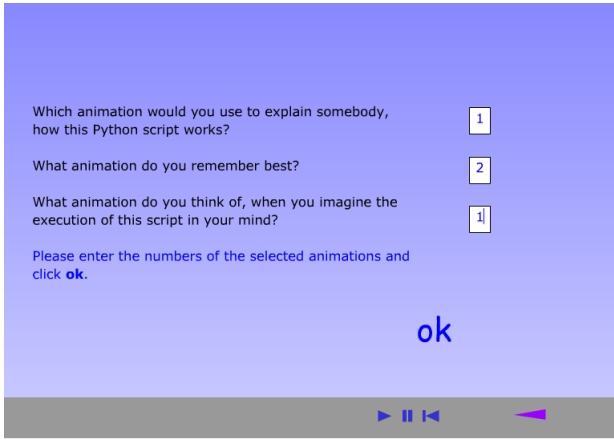
Python Visual



Problem



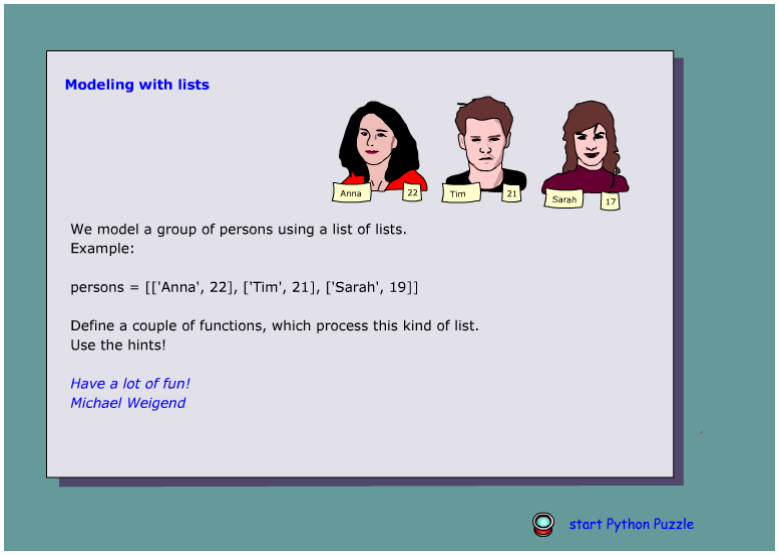
Four alternative models

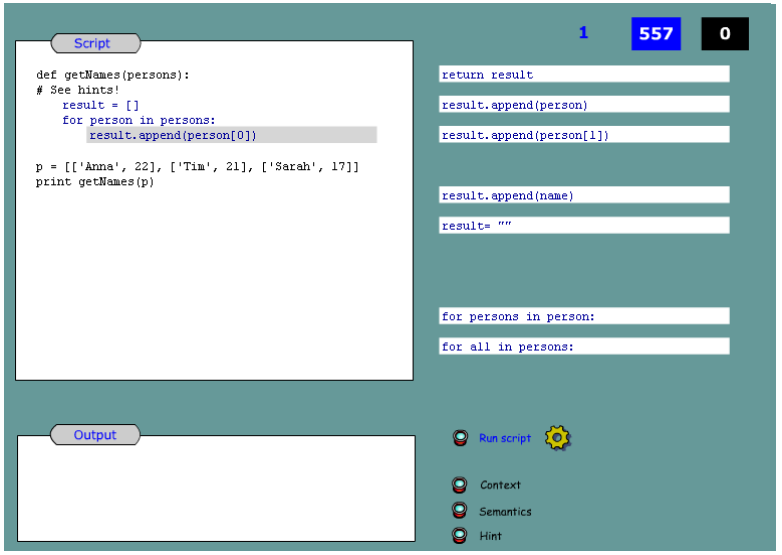


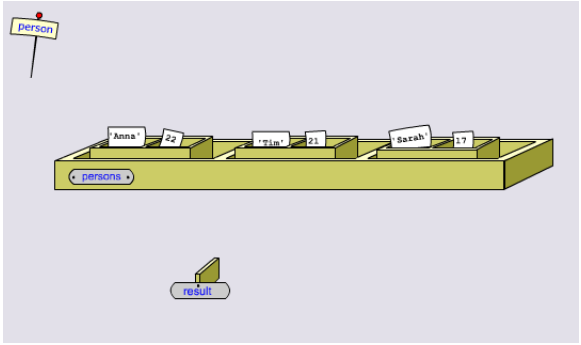
Three questions

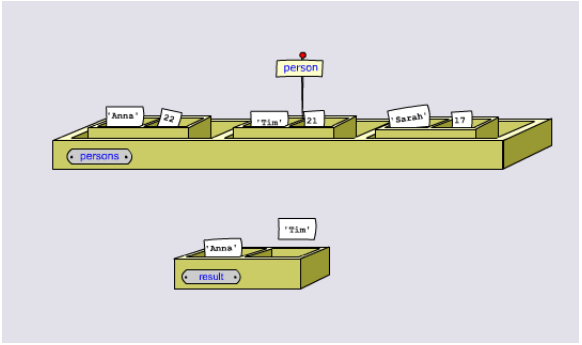
Observed and recorded are: decisions and decision times

Python Puzzle







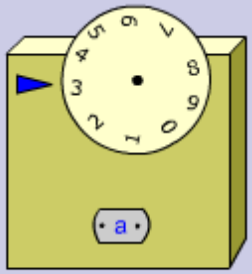


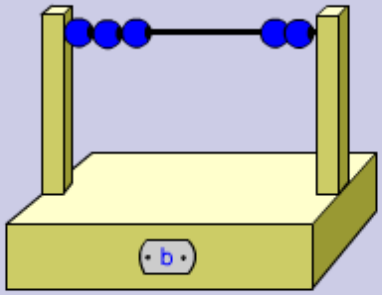
Observed and recorded: For how long did they watch visual help? How many test runs? Which correct and incorrect program blocks did they use?

Python Quiz

Aufgabe 1 | 20 | 70

```
a = 3
b = a
```





3

I bet ... 0 points 5 points 10 points

The model is .. good bad

Motivation through …

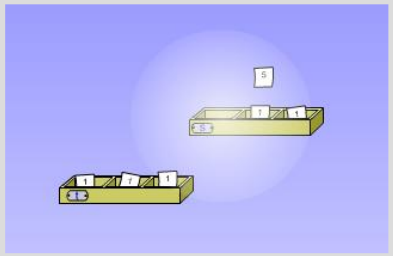
… workshop dramaturgy

Teaching with the Python Visual Sandbox

[home]

Evaluation of *Changing lists*

All sessions of group *PVS Seminar Diesel Gymnasium Augsburg* .
Number of evaluated sessions: 18

Model 1	Question 1 ... would you use to explain	Question 2 ... do you remember best	Question 3 ...when you imagine the execution of the script
 duration: 6 sec -- watched: 27 sec	11 percent	6 percent	11 percent

Group results of a “Python Visual”

Motivation through …

… Activity Reports

Python Visual Sandbox

Michael Weigend 17. 7. 2004 - 30. 5. 2005

[home]

Test_hans 's Activity Report

You have got **-12890** points.

Activities

Python Visual	**Time of first valid session**	**Points**
Recursive function that mirrors a string. *(1 valid session)*	Thu Aug 25 20:33:30 2005	20
Regular expressions *(2 valid sessions)*	Sat Aug 20 08:01:32 2005	20
What happens, when a function returns something? *(2 valid sessions)*	Tue Aug 2 08:18:25 2005	20
Recursive computation of factorial *(1 valid session)*	Wed Jan 4 14:48:54 2006	20
Analogies for iterations *(1 valid session)*	Mon Apr 18 23:07:21 2005	20
Multilists *(2 valid sessions)*	Sat Apr 23 10:11:59 2005	20
Python Puzzle	**Time of session**	**Points**
Modeling a group	Thu May 26 22:57:50 2005	-200
Modeling a group	Thu May 26 23:14:24 2005	60

… competition

The Python Visual Sandbox

[home]

Highscores

It is *Wednesday, 25. October 2006, 12:18:47.*
You did 3 Python Quizes and 6 Python Puzzles. Here are the highscores of these games.

Python Quiz

Modeling a group using a list of tuples.

1	420 points	jr	DEU	Thursday, 05. January 2006 11:11:27 (MET)
2	360 points	markus	DEU	Tuesday, 31. January 2006 16:38:18 (MET)
3	300 points	TE	DEU	Tuesday, 31. January 2006 16:40:40 (MET)
4	300 points	honey123	DEU	Thursday, 26. January 2006 10:13:44 (MET)
5	290 points	MeanMetal	CHN	Thursday, 05. January 2006 07:17:40 (MET)
6	285 points	asir	DEU	Saturday, 04. March 2006 14:14:32 (MET)
7	220 points	Unseensurvivor	CHN	Thursday, 05. January 2006 07:13:34 (MET)
8	220 points	vreni	DEU	Tuesday, 11. October 2005 11:59:56 (MET)
9	210 points	austrian_angela	AUT	Saturday, 04. March 2006 14:17:23 (MET)
10	210 points	Felix	DEU	Thursday, 26. January 2006 10:02:29 (MET)

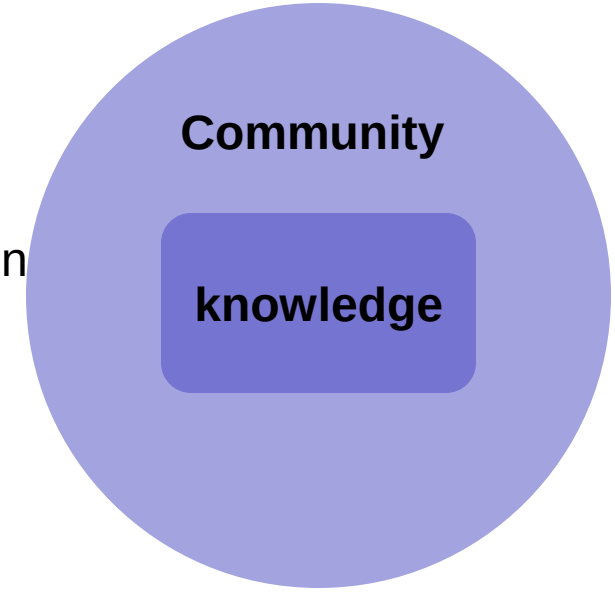
V-Quiz (since 2008)

Web 2.0 technology
Metaphor: Quiz Show

Roles

3 Some basic Intuitive models in informatics

Player (candidate)
Designer
Quiz Host
Administrator



For technology nerds …

Virtual server
Turbogears Rapid Web Development Framework
Python
Model-View-Controller Design:

- SQLObject for ORM
- MySQL data base

- KID: Python-based templating language (XHMTL templates)

- CherryPy HTTP-Server (OOP)

Let’s have a look …

3 Some basic Intuitive models in informatics

Web 2.0 technology
Metaphor: Quiz Show

Roles

Player (candidate)
Designer
Quiz Host
Administrator

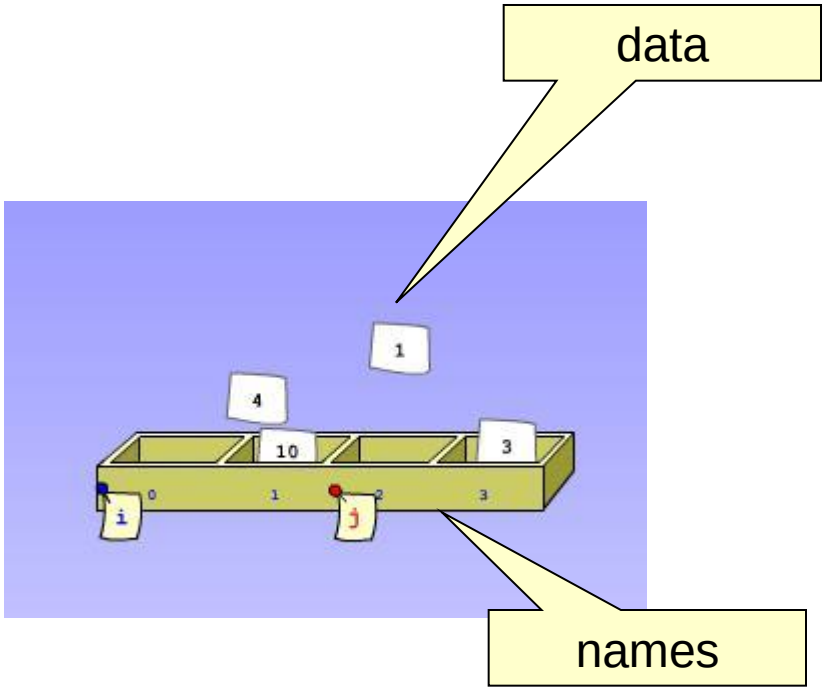
3 Some basic groups of intuitive Models

- actor models
- names
- data
- functions
- control
- processing
- classes
- objects

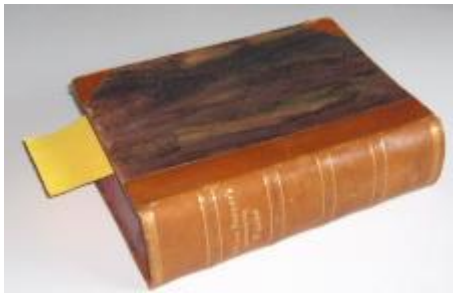
Actor models

One actor

Many actors



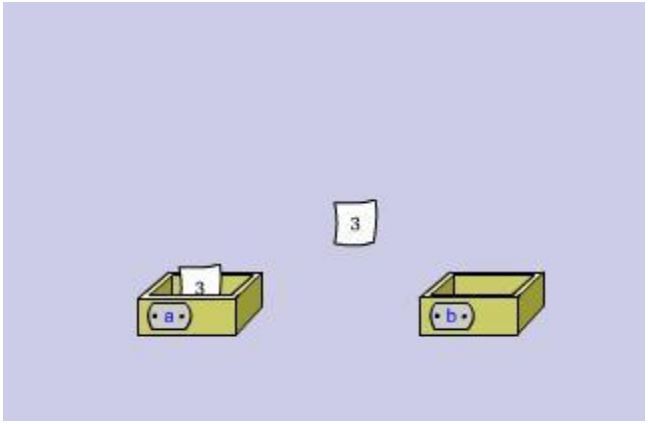
Names



Identifying, addressing

Container versus Reference

133/154

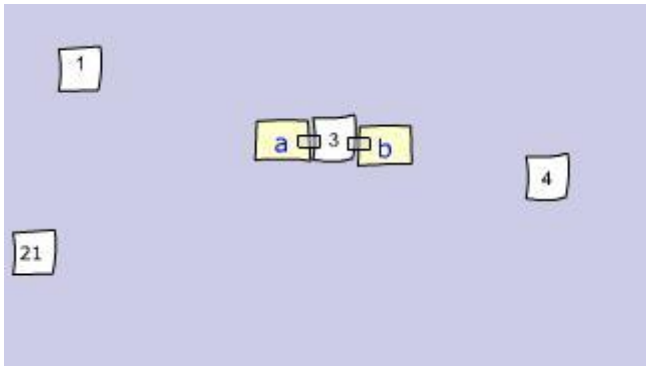



```
a = 3
b = a
```

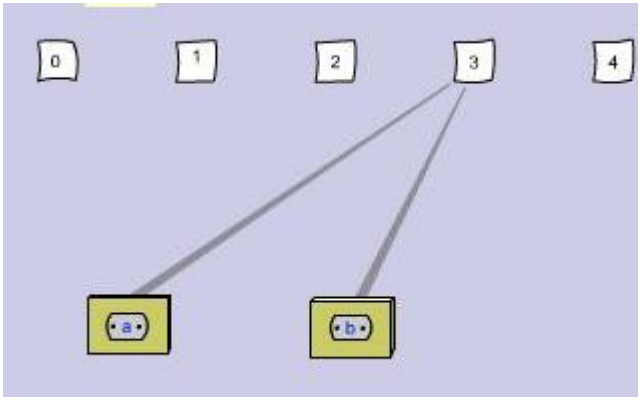
Name persistent

Data volatile

119/154



119/154



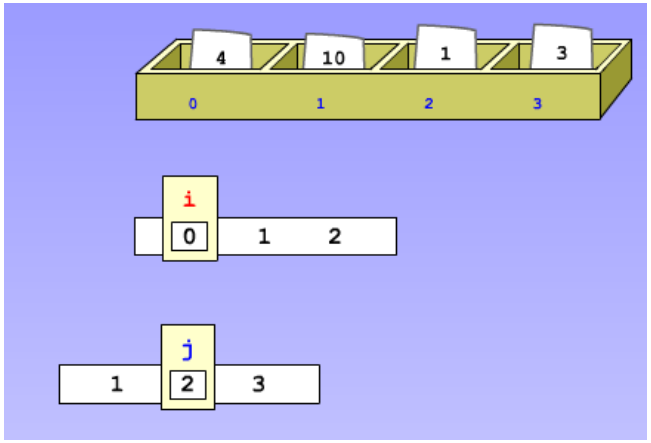
Data persistent

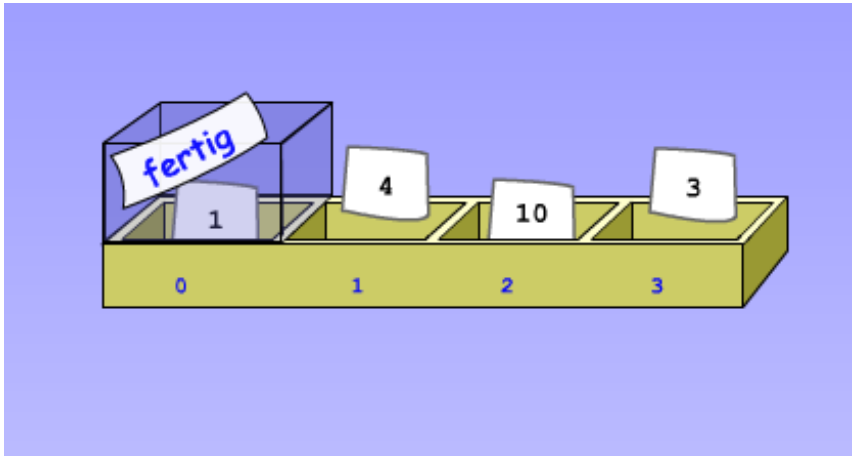
Name volatile

Explicit and implicit names

```
s = [10, 4, 1, 3]
for i in range(len(s)):
  for j in range(i+1, len(s)):
     if s [j] < s[i]:
        s[i], s[j] = s[j], s[i]
```

Explaining models





2 /16

"Stepper", Sajaniemi, 2002

9 /16

No split of attention
(Sweller & Chandler)

Indirect Names

Constructed from other names

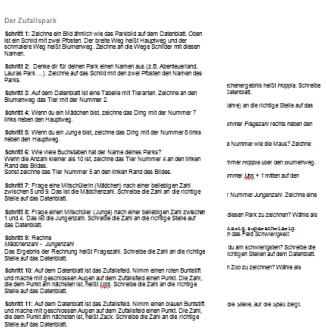
Everyday life:
„My brother Andi‘s oldest daughter‘s dog "

Informatics:

```
s[2]
object.attribute
sqrt(2)
```

How well do children understand naming concepts?

Study: The Random Park


Der Zufallspark

Schritt 1: Zeichne ein Bild ähnlich wie das Parkbild auf dem Datenblatt. Oben ist ein Schild mit zwei Pfaden. Der breite Weg heißt Hauptweg und der schmalere Weg heißt Blumenweg. Zeichne all die Wege Schilder mit diesen Namen.

Schritt 2: Denke dir für deinen Park einen Namen aus (z.B. Abenteuerland, Laurás Park ...). Zeichne auf das Schild mit den zwei Pfaden den Namen des Parks.

Schritt 3: Auf dem Datenblatt ist eine Tabelle mit Tierarten. Zeichne an den Blumenweg das Tier mit der Nummer 2.

Schritt 4: Wenn du ein Mädchen bist, zeichne das Ding mit der Nummer 7 links neben den Hauptweg.

Schritt 5: Wenn du ein Junge bist, zeichne das Ding mit der Nummer 8 links neben den Hauptweg.

Schritt 6: Wie viele Buchstaben hat der Name deines Parks?
Wenn die Anzahl kleiner als 10 ist, zeichne das Tier Nummer 4 an den linken Rand des Bildes.
Sonst zeichne das Tier Nummer 5 an den linken Rand des Bildes.

Schritt 7: Frage eine Mitschülerin (Mädchen) nach einer beliebigen Zahl zwischen 6 und 9. Das ist die Mädchenzahl. Schreibe die Zahl an die richtige Stelle auf das Datenblatt.

Schritt 8: Frage einen Mitschüler (Junge) nach einer beliebigen Zahl zwischen 1 und 4. Das ist die Jungenzahl. Schreibe die Zahl an die richtige Stelle auf das Datenblatt.

Schritt 9: Rechne
Mädchenzahl – Jungenzahl
Das Ergebnis der Rechnung heißt Fragezahl. Schreibe die Zahl an die richtige Stelle auf das Datenblatt.

Schritt 10: Auf dem Datenblatt ist das Zufallsfeld. Nimm einen roten Buntstift und mache mit geschlossen Augen auf dem Zufallsfeld einen Punkt. Die Zahl, die dem Punkt am nächsten ist, heißt Ups. Schreibe die Zahl an die richtige Stelle auf das Datenblatt.

Schritt 11: Auf dem Datenblatt ist das Zufallsfeld. Nimm einen blauen Buntstift und mache mit geschlossen Augen auf dem Zufallsfeld einen Punkt. Die Zahl, die dem Punkt am nächsten ist, heißt Zack. Schreibe die Zahl an die richtige Stelle auf das Datenblatt.

scheinergroms heißt Hoppla. Schreibe Datenblatt.

lahre) an die richtige Stelle auf das

immer Fragezahl rechts neben den

s Nummer wie die Maus? Zeichne

immer Hoppla über den Blumenweg.

immer Ups + 1 mitten auf den

r Nummer Jungenzahl. Zeichne eine

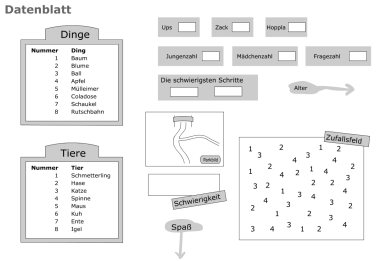
diesen Park zu zeichnen? Wähle als

6 6+6 6 8+8 8+8 8+8=16 16
n das Feld Schwierigkeit.

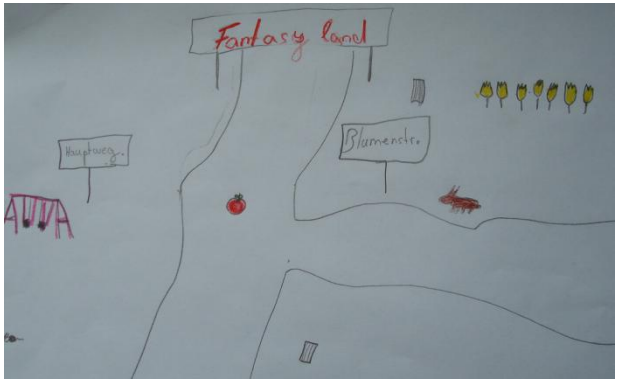
du am schwierigsten? Schreibe die richtigen Stellen auf dem Datenblatt.

n Zoo zu zeichnen? Wähle als

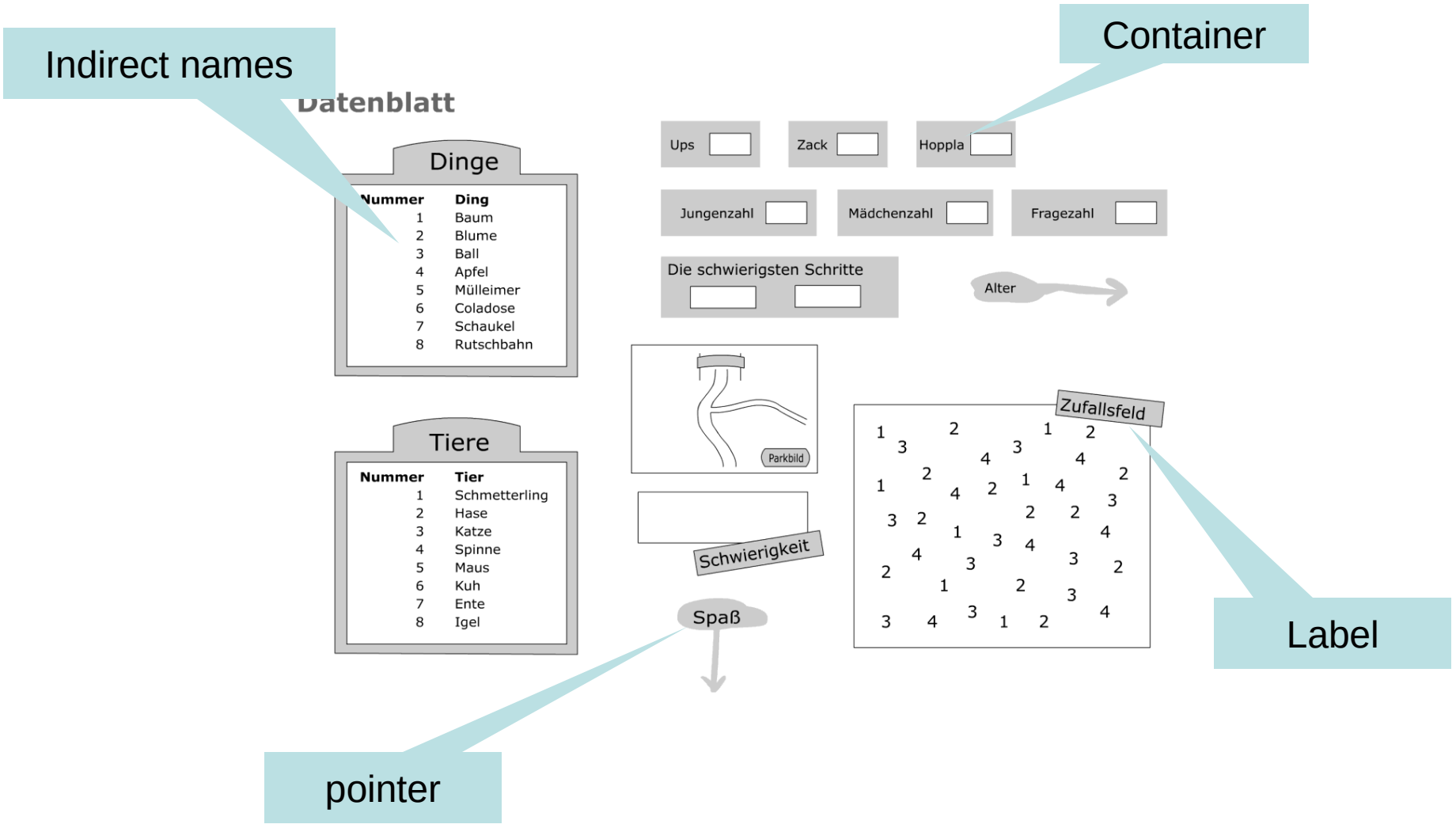
die Stelle, auf die sie zeigt.



149 children, 3rd und 4th grade,
Winter 2008/2009,
Münster andWitten



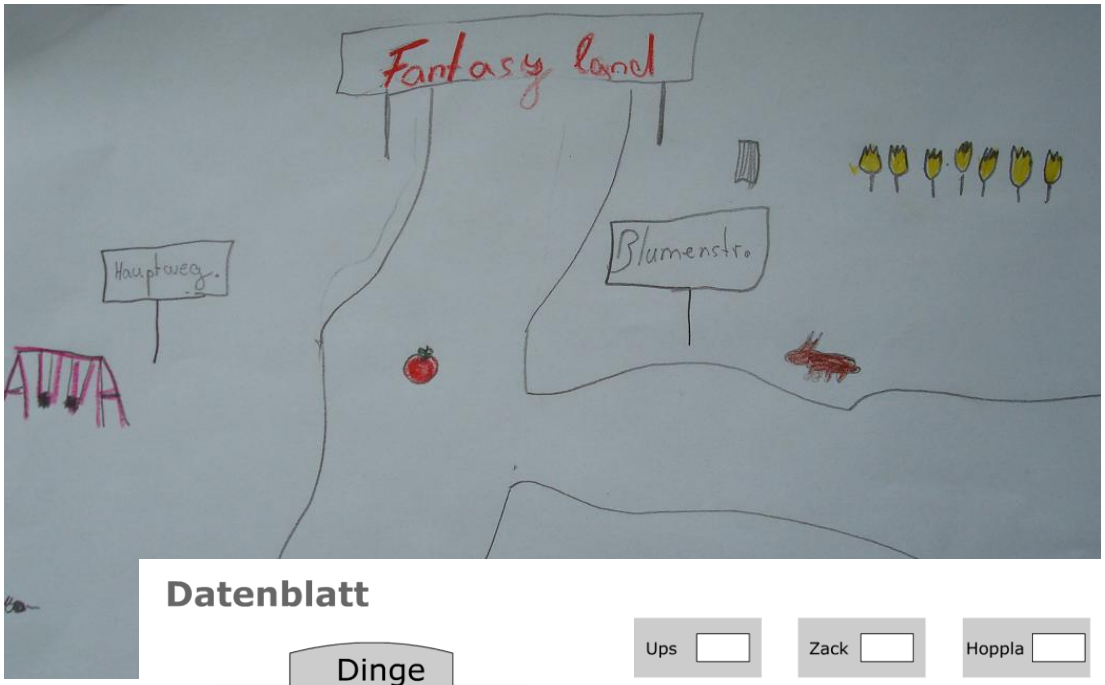
The Data sheet



Indirect names – simple access

Step 3: On the data sheet there is a table with animals. Draw animal number 2 at the Flower Street.

85% correct



Datenblatt

Ups | Zack | Hoppla

Jungenzahl | Mädchenzahl

Die schwierigsten Schritte

Alter

Dinge

Nummer	Ding
1	Baum
2	Blume
3	Ball
4	Apfel
5	Mülleimer
6	Coladose
7	Schaukel
8	Rutschbahn

Tiere

Nummer	Tier
1	Schmetterling
2	Hase
3	Katze
4	Spinne

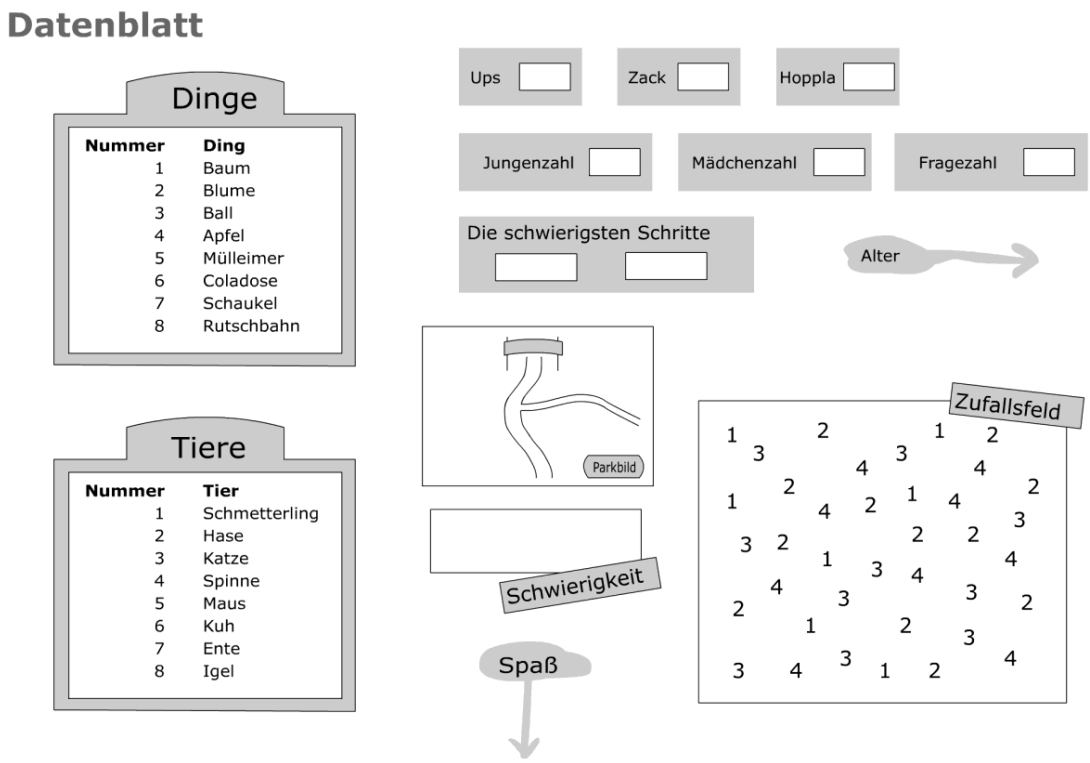
Parkbild

Schwierigkeit

Container

Step 7: Ask a girl for a random number between 5 and 9. This is the girl‘s number (Mädchenzahl). Write this number at the correct spot on the data sheet

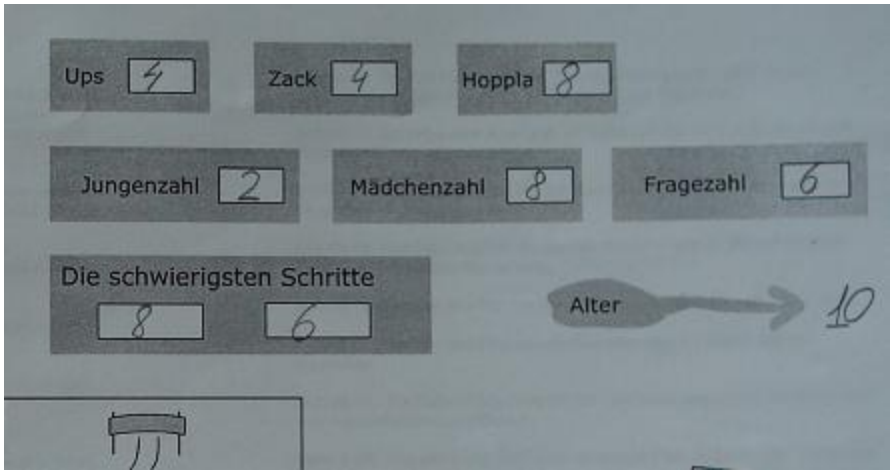
98% correct



Calculating with names

Step 12: Calculate Ups plus Zack. The result is called Hoppla. Write this number at the correct spot on the data

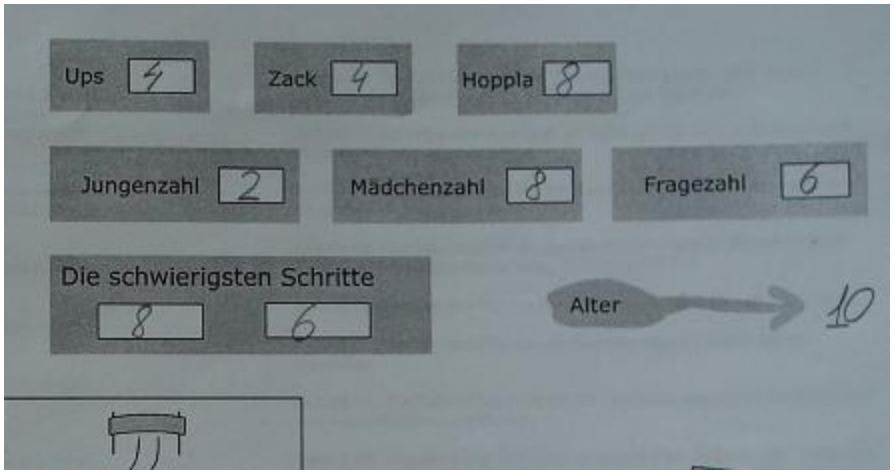
95% correct



Pointer

Step 13: Write your age at the correct spot on the data sheet.

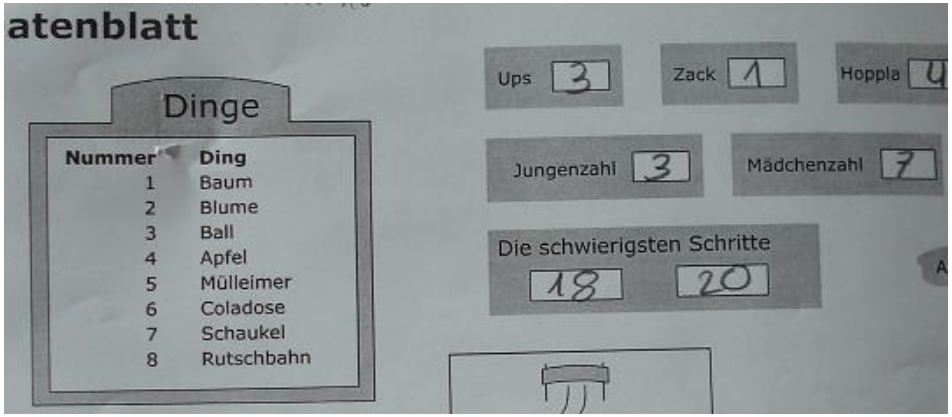
92% correkt



The Limit

Step 18: A Bubu is a thing with number Jungenzahl. Draw a sequence of Mädchenzahl Bubus.

16% correct

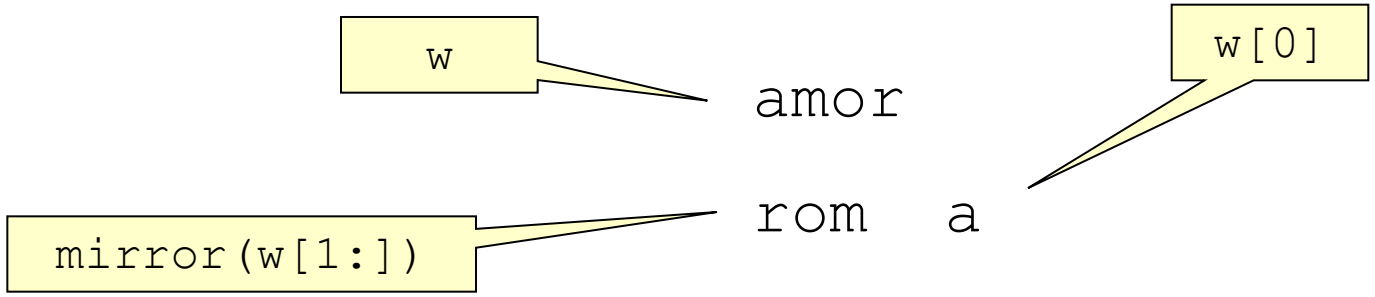


Function calls as indirect names

cognitive advantage when interpreting recursion

```
def mirror (w):
    if w == "": return w
    else: return mirror(w[1:]) + w[0]
```

Function call = name of a string



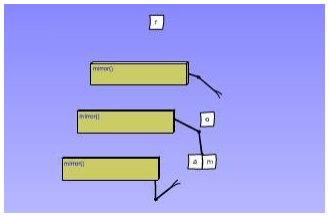
Is one step of recursion enough?

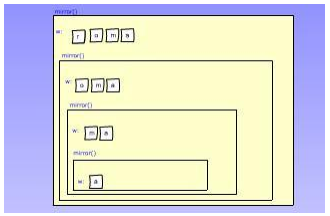
```
def mirror (w):
    if w == "": return w
    else: return mirror(w[1:]) + w[0]
```

Which model would you use to explain?

Full depth

12/30

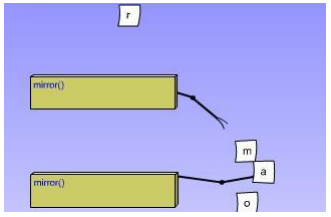


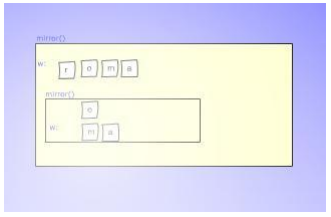


11/30

One step

3/30

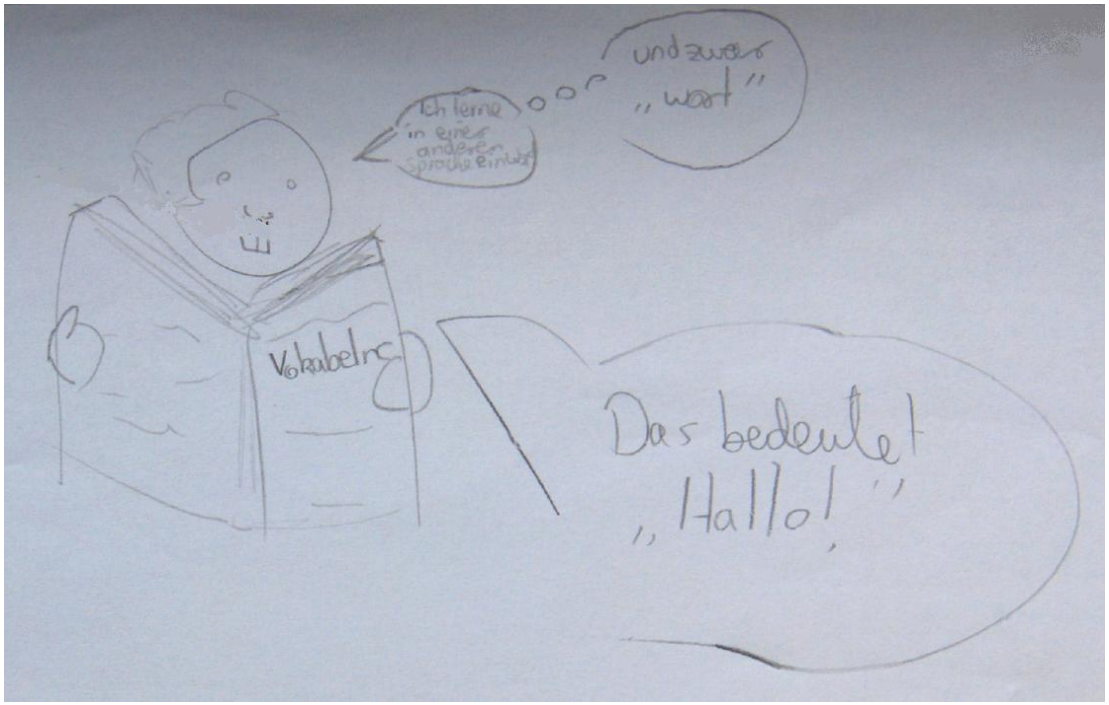




4/30

Naming as associating

```
string wort;
wort = "Hallo";
```



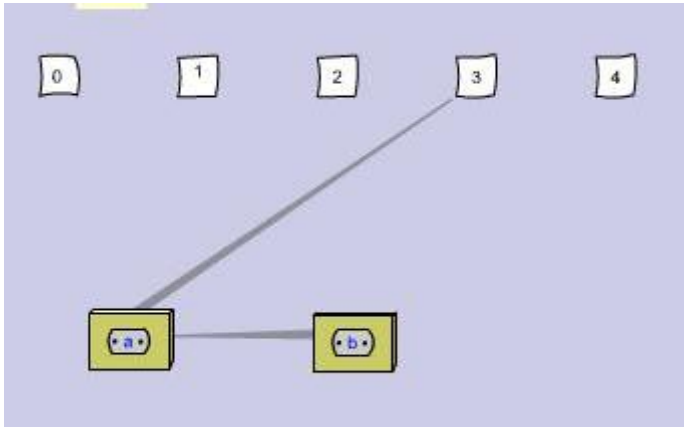
Visualization made by a student (17)

symmetric — Brot – bread

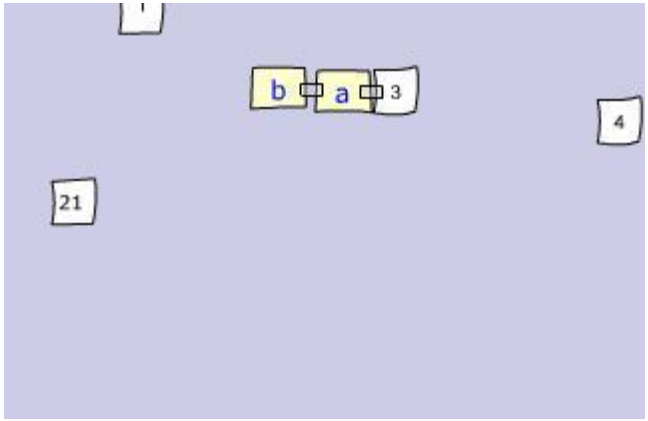
cumulativ — Brot – bread – du pain

Chains of association

```
a = 3
b = a
```



56%

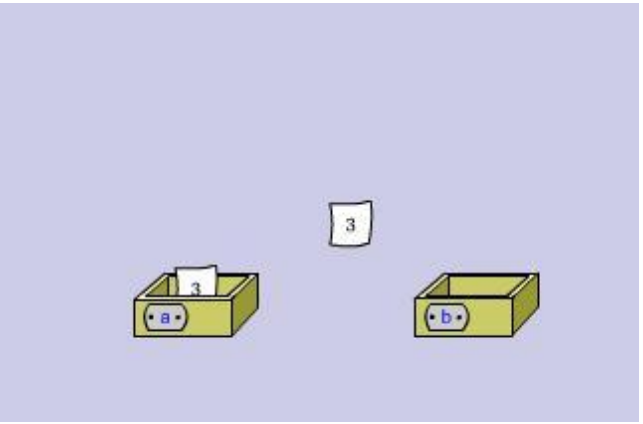


51%

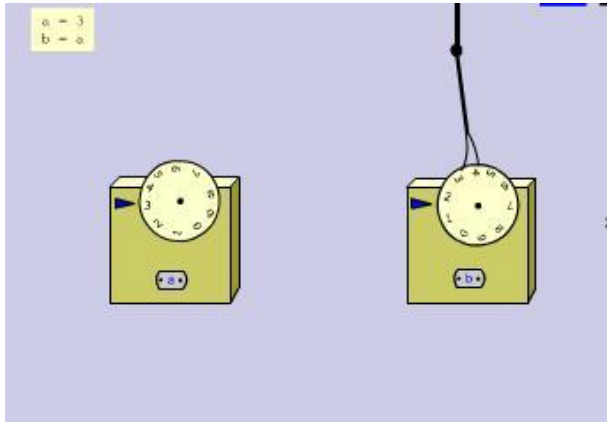
n = 154

Data – states or entities?

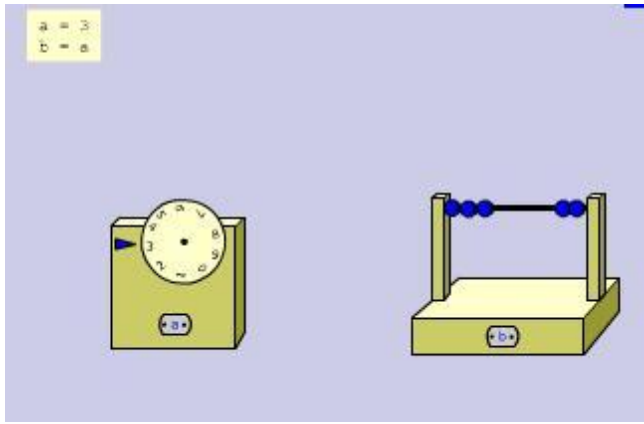
```
a = 3
b = a
```



86.4 %



64.9 %

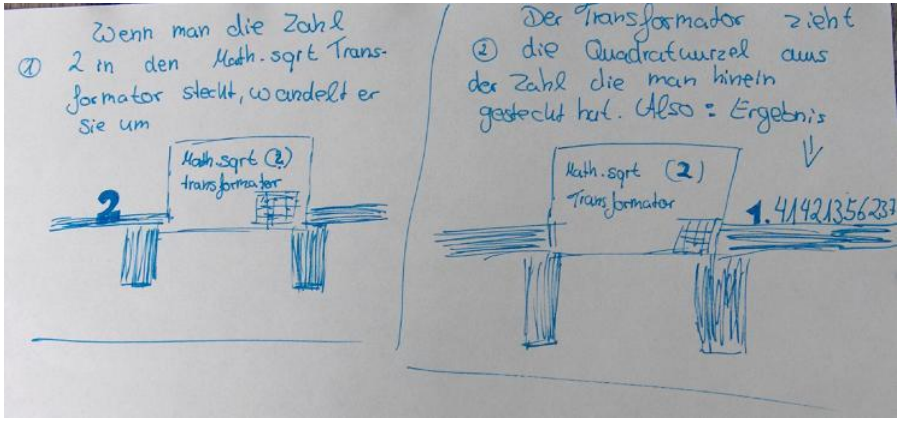


66,7 %

N = 154

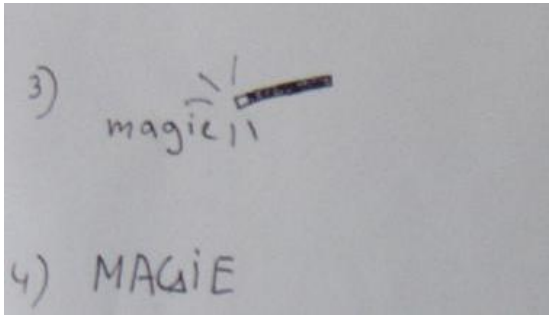
Functions – factories or tools?

Factory




```
Math.sqrt(2)
```

Tool




```
b = a.upper()
```

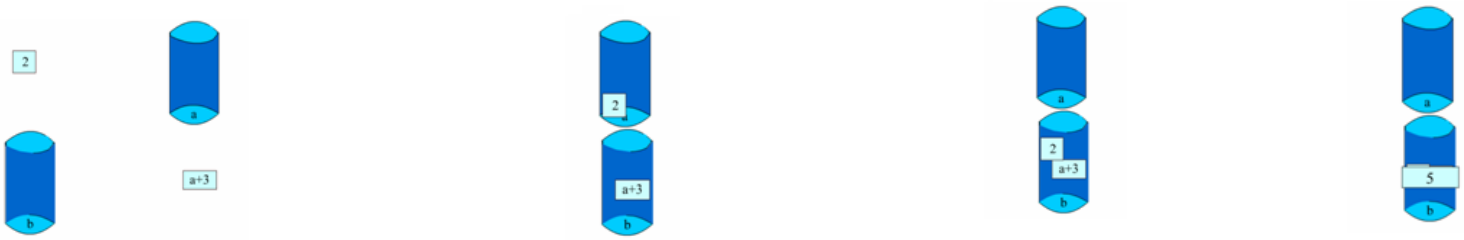
Control

- Sequence
- Data flow flow of control
- Branch
- Repetition
- Iteration
- Signal

Parallelizing a Sequence

Sequence perceived as a wholeness?

```
int a;
int b;
a = 2;
b = a + 3;
```



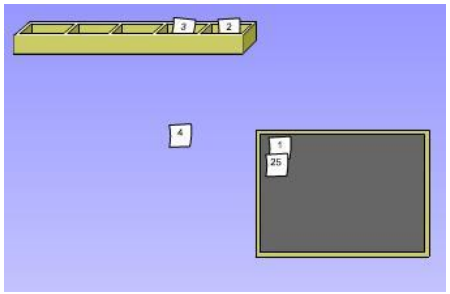
Flash-Film made by a 17-years-old student

Iteration

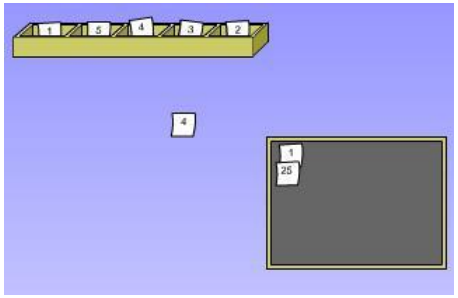
```
s = [1, 5, 4, 3, 2]
    for i in s:
        print i*i
```

Which model would you use to explain the program?

9%

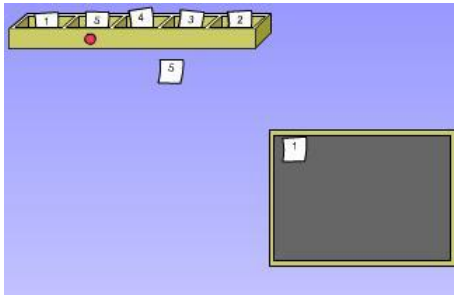


20%

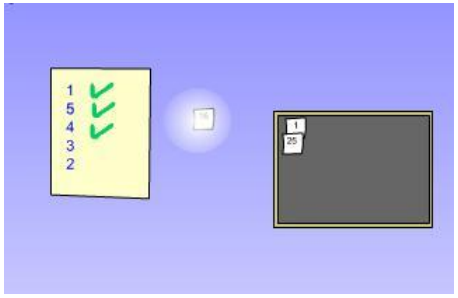


Focused model

45%



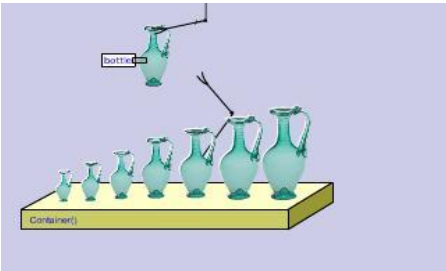
26%



Everyday life

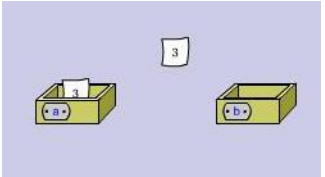
n = 66

Data processing



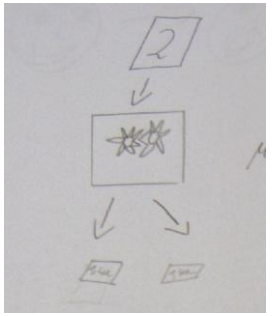
Begin to exist
- generate
- choose
- produce, construct

move
- names
- data



Annihilate
- destroy
- mark as invalid
- loose

Metamorphosis
- names
- data

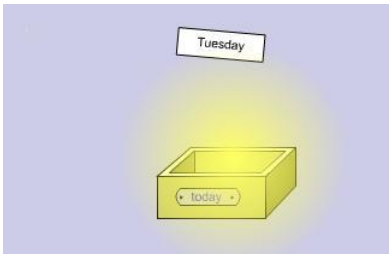


Implicit destruction

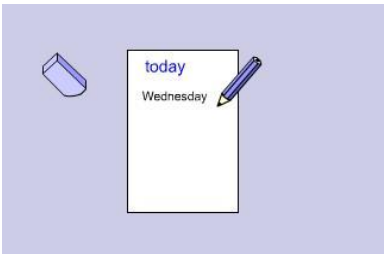
```
today = "Monday"
today = "Tuesday"
today = "Wednesday"
```

n = 154

destroy

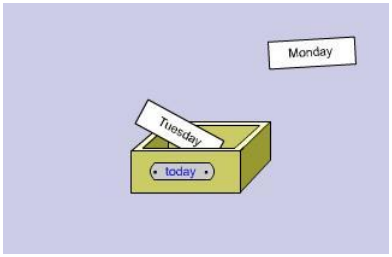


76.0 %



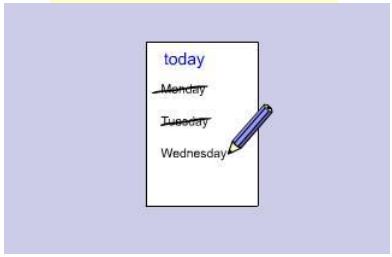
72.1%

loose



77.9%

Mark as invalid



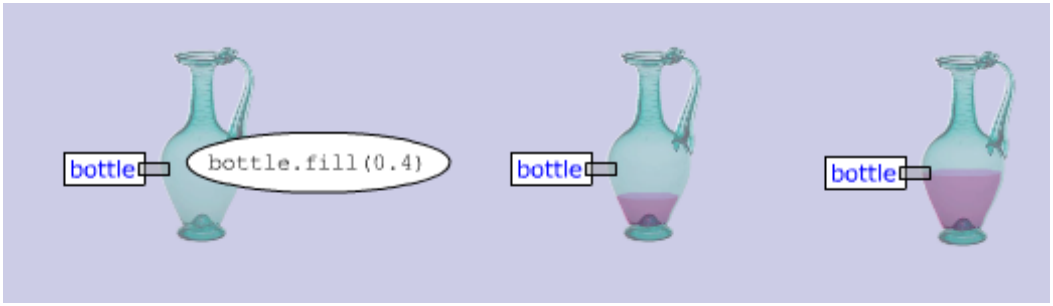
66.2 %

Objects

Allocation of activity

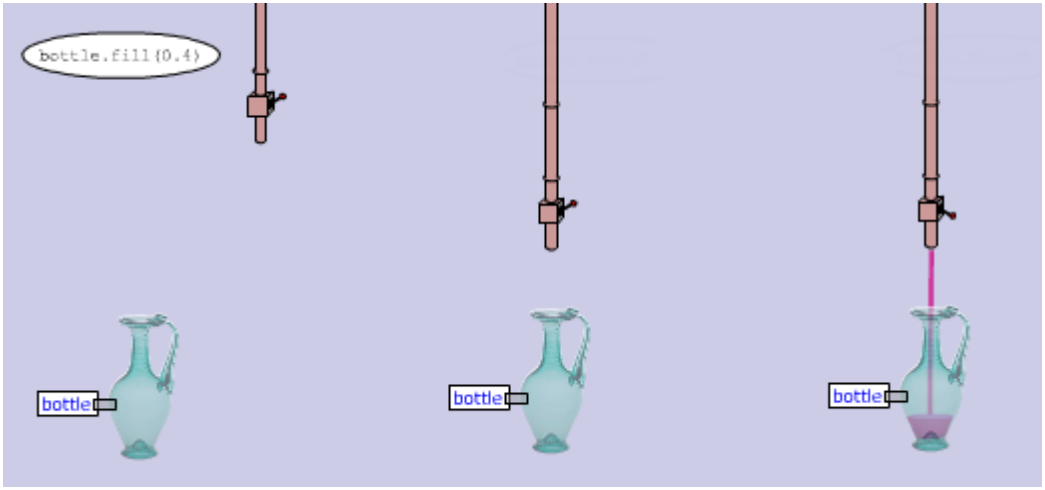
```
bottle.fill(0.4)
```

OOP



21/23

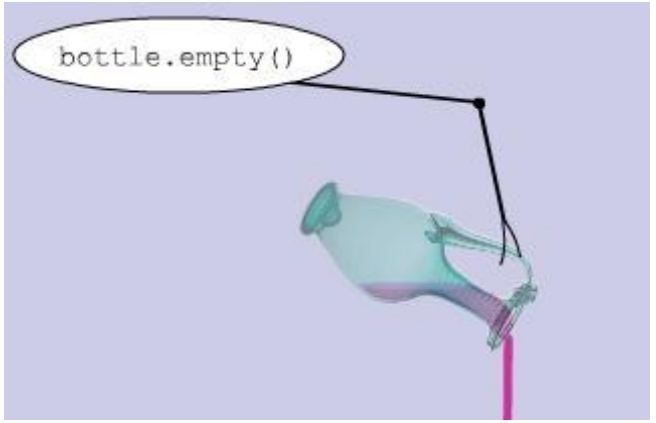
naturalistic



16/23

Message as actor

```
bottle.empty()
```

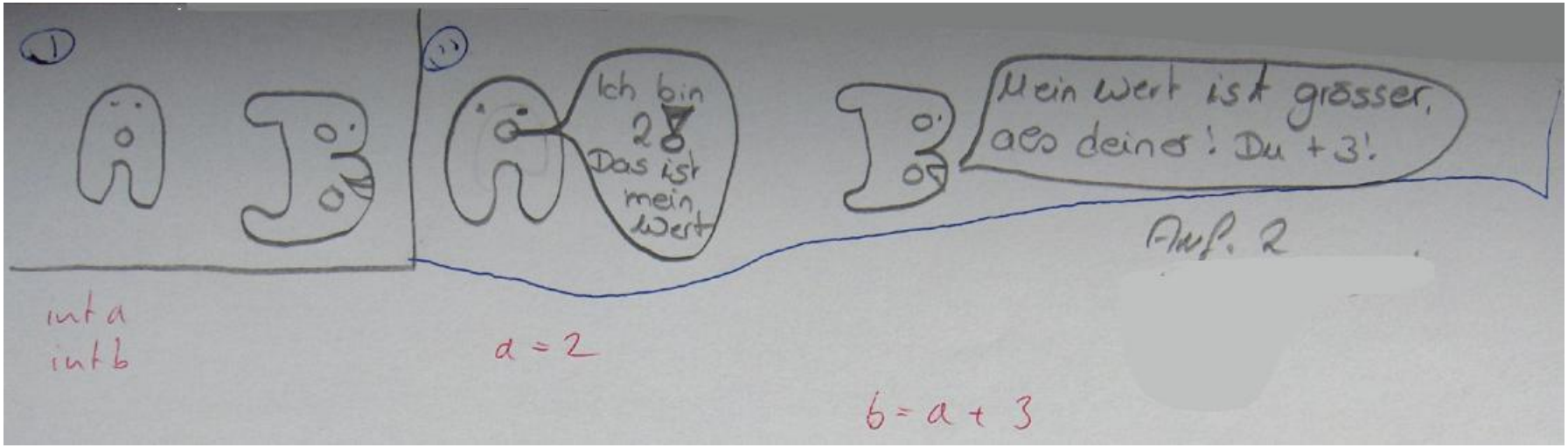


4 Intuitive Modeling

= understanding (reconstructing)
formal program text
by intuitive models

Dramatizing

```
int a;
int b;
a = 2;
b = a + 3;
```



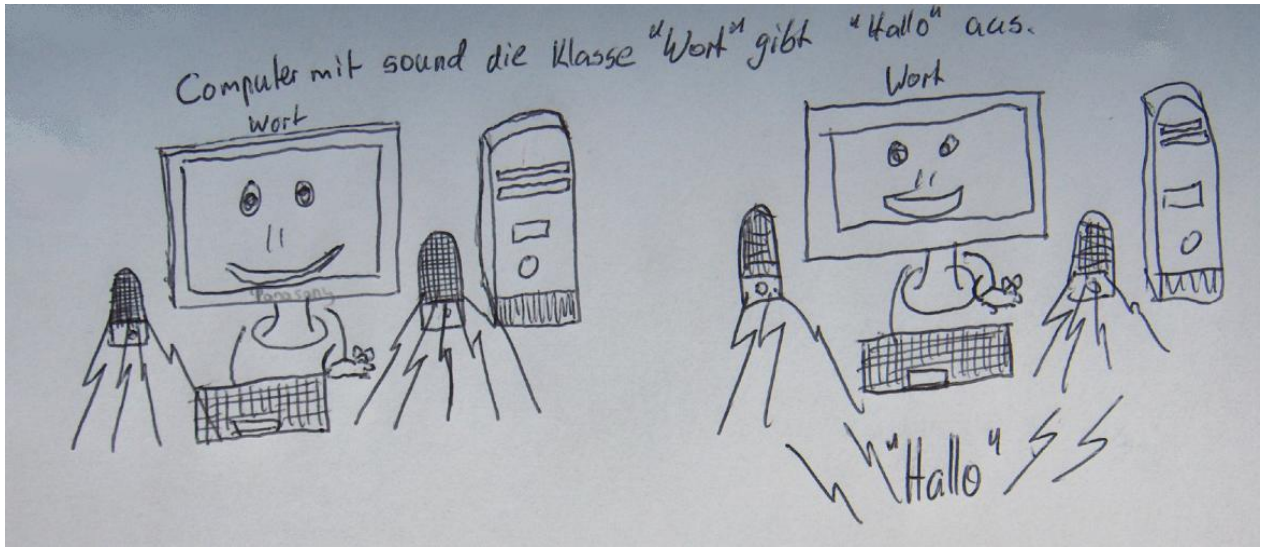
Storyboard

Reverse modeling

modeling

Reality → Program

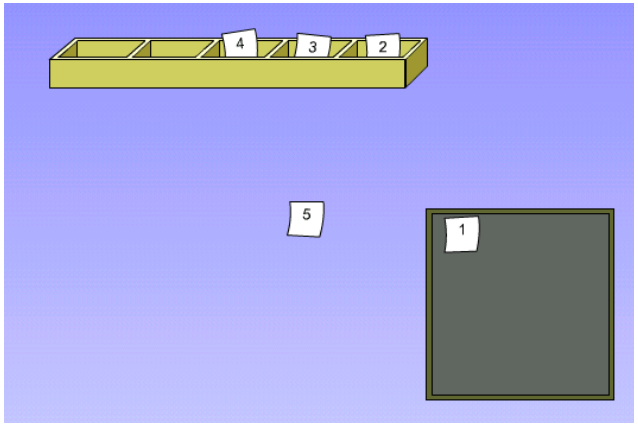
Reverse modeling




```
String wort;
wort = "Hallo";
```

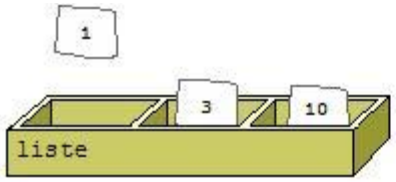
Focusing

```
for i in [1, 5, 4, 3, 2]:
    print i**2
```



Risk: misconception

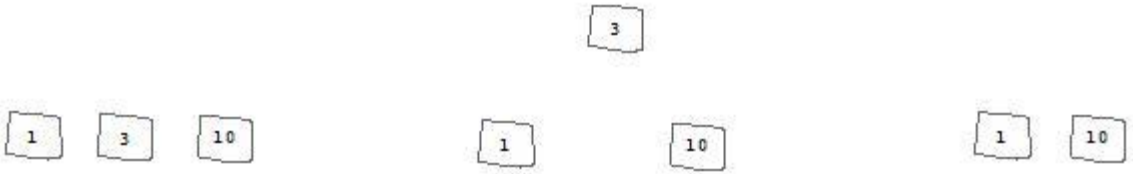
Using different models for the same thing



Entity on its own

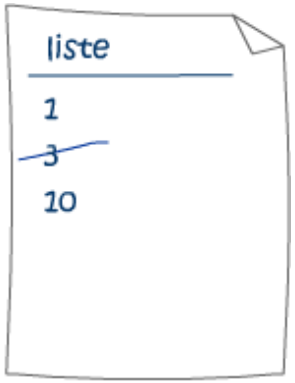
swap

change



remove

insert



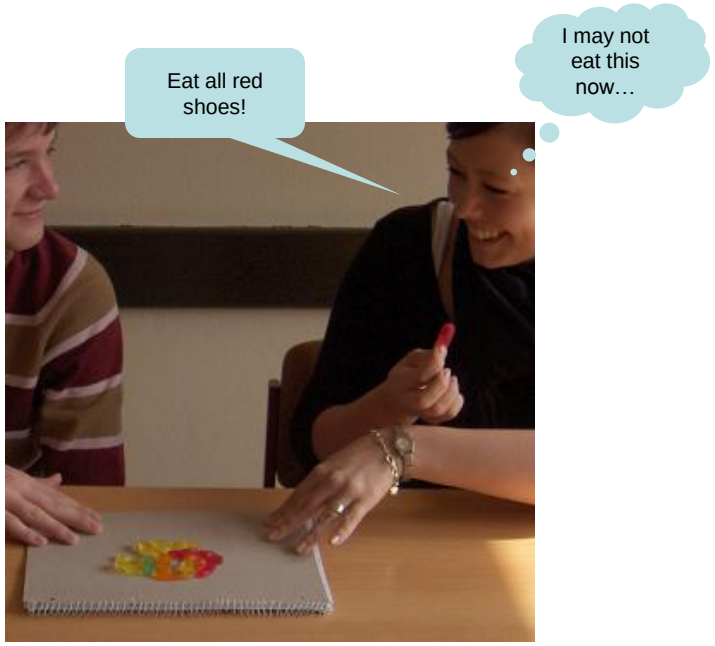
append

remove

5 Visualization in classroom activities

1rst example: Roleplays

```
Algorithm eat all red shoes (paper)
if not red shoe on paper:
      return paper
else:
      take red shoe from paper
      eat all red shoes (paper)    # recursive call
      eat red shoe
      return paper
end
```



Discussion

- Which elements of the role play correspond to which part of the program?
- Which elements are decoration?
- Is the model easy to understand? Why?
- Where are differences between model and algorithm?

How to design role plays?

Objectives:

- Each participant should be active
- Clear roles, it should be possible to improvise and interpret freely the role

Care to make a workshop on role play design tomorrow?

- Try the “Processor Game” by Lars Tijsma (NL)
- Develop variations …

2nd example: Scaffolding

Intuition

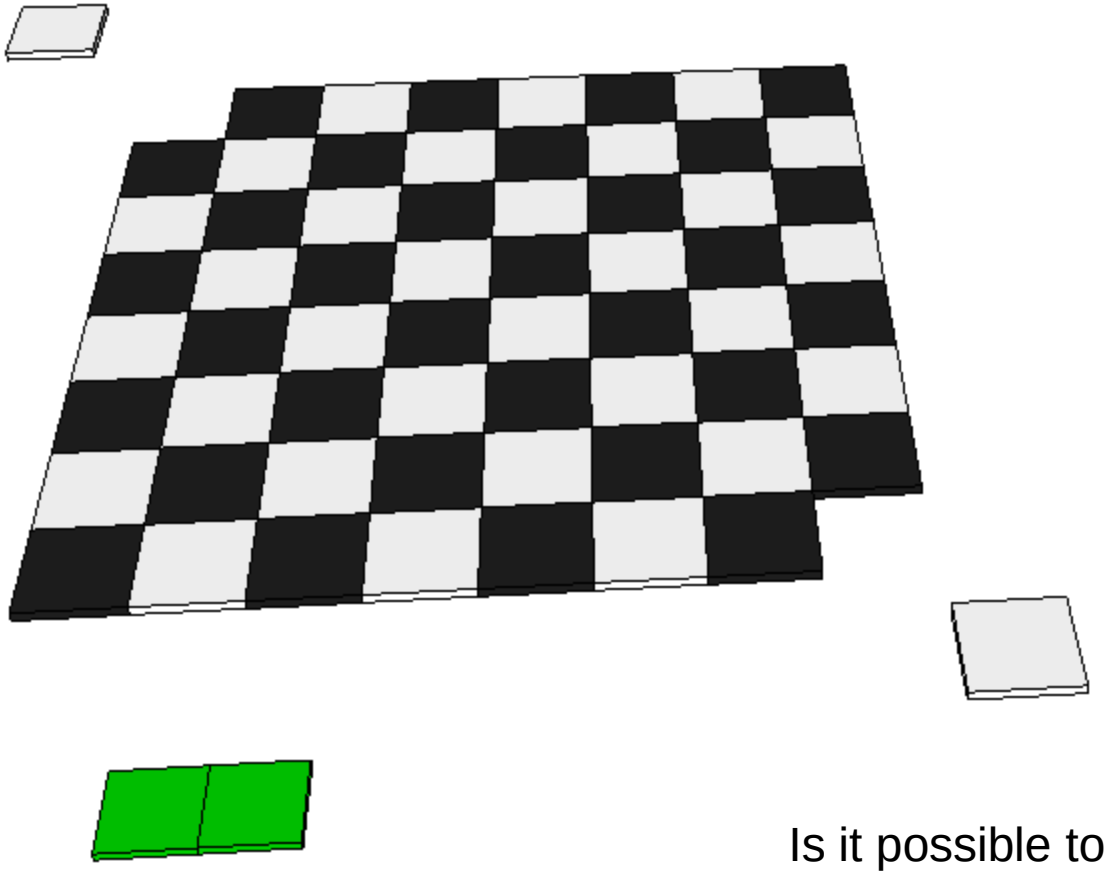
Program

```
s = [10, 4, 1, 3]
for i in range(len(s)):
  for j in range(i+1, len(s)):
     if s [j] < s[i]:
        s[i], s[j] = s[j], s[i]
```

Try to “read” the intuition

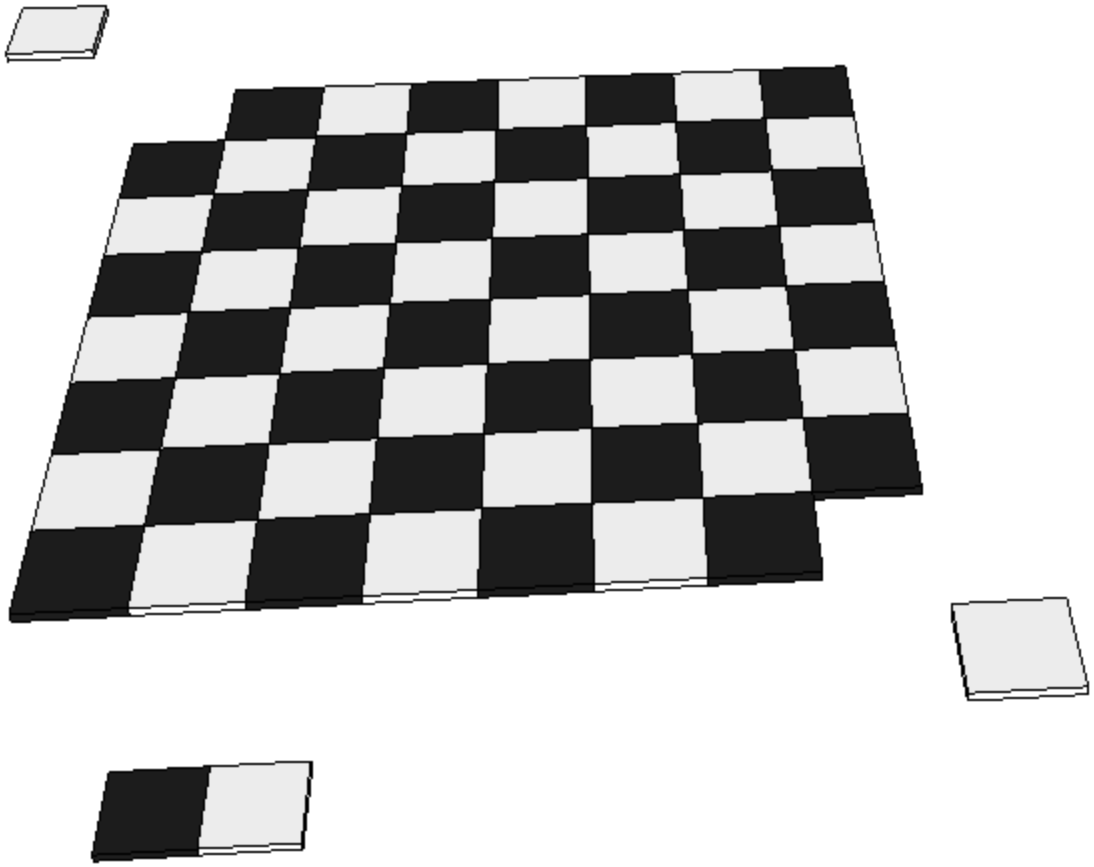

```
def sort (s):
    indices  = range(len(s))
    for i in indices:
        x = min (s[i:])
        s.remove (x)
        s.insert(i, x)
    return s
```

Experiment: The problem of the incomplete chess board



Is it possible to cover the incomplete chess board with domino blocks?

An appropriate representation

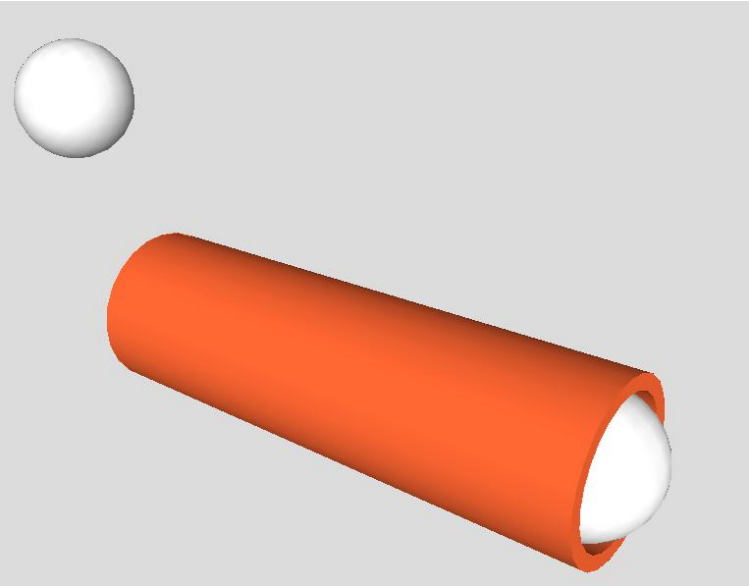


John Anderson 1996

3rd example: Visualization as "door opener"

Does not represent the whole solution, but it "opens a door" and leads to a solution

Example: How to coordinate producer and consumer processes?

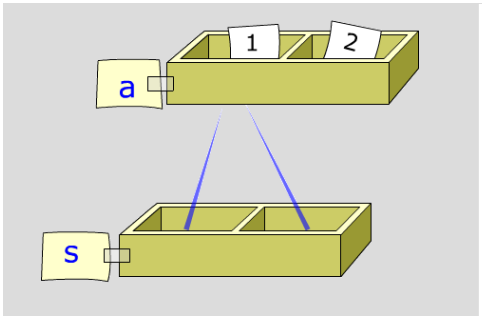


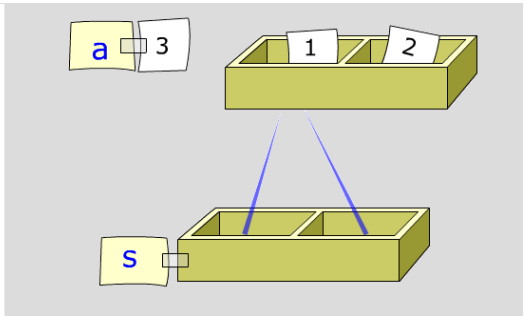
Just think of a smart pipe in the middle – that's all!

Thread safe Queue

4th example: Puzzles = artificial logical bugs

```
>>> a = [1,2]
>>> s = [a, a]
>>> a = 0
>>> print (s)
[[1, 2], [1, 2]]
```





Why?

Conclusion:
Talk about intuitive models in class

- get deeper understanding
- improve communication
- succeed in programming

Thank you for your attention