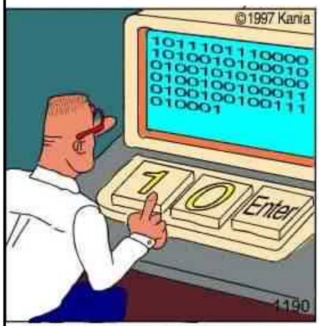


Real programmers code in binary.

Software Visualization

Education
and
Evaluation

Education and Evaluation

- Education
  - Example Scenarios for using Algorithm Animations for Teaching
- Evaluation
  - Questions
  - Evaluation Methods
  - Results of some experimental studies

See [Hundhausen,Douglas&Stasko:02]

Classical Scenarios

	Learner Involvement
• View animation of algorithm – Fixed animation – With user input • Read algorithm and view animation – Fixed animation – With user input	Low
• Implement algorithm and create your own animation	High

Advanced Learning Scenarios

- Visualized Path Testing [Korhonen,Sutinen&Tarhio:02]
- Exploring the Functional Structure [Faltin:02]

Exploring the Functional Structure

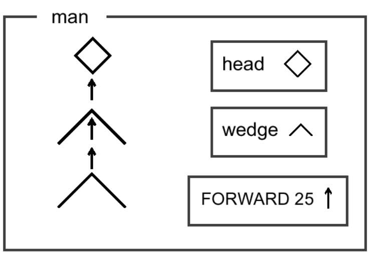
- Goal:
  - exploratory learning, so that the learner actively reinvents parts of an algorithm.
- Method:
  - Algorithm is structured into many small functions so the student only has to think about one function at a time. Finding the steps of a function for a specific data input can further be eased by providing constraints that delimit the exploration space.

Exploring the Functional Structure

- Provide simulations of sub tasks, student can test sub taks and compose them.

An Example in LOGO:

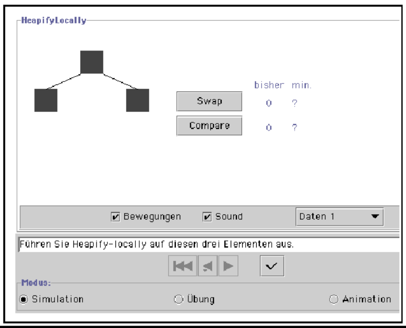
```
TO man
    wedge
    forward 50
    wedge
    forward 25
    head
END
```



Exploring the Functional Structure

- Sub-Task of Binomial-Heap Program

Exercise:
Establish heap property (parent greater or equal than its children) by comparing and swapping.



Visualized Path Testing

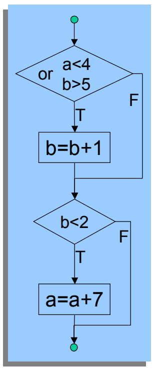
- **Statement coverage** is satisfied, when every (non-control-flow) statement is executed at least once with the test set.
- **Branch coverage** is satisfied, when every edge of the flow graph of the program is applied at least once with the test set.
- **Path coverage** is satisfied, when the test set contains a test case for every possible control path in the flow graph of the program.
- **Problem:** Number of paths is exponential to the number of branches.

Visualized Path Testing

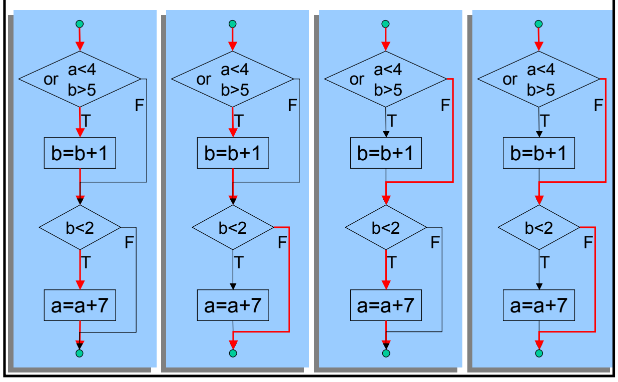
- Exercise: Find minimal test set satisfying a certain coverage condition.
- Example:

  if ((a<4) || (b>5)) then b=b+1;

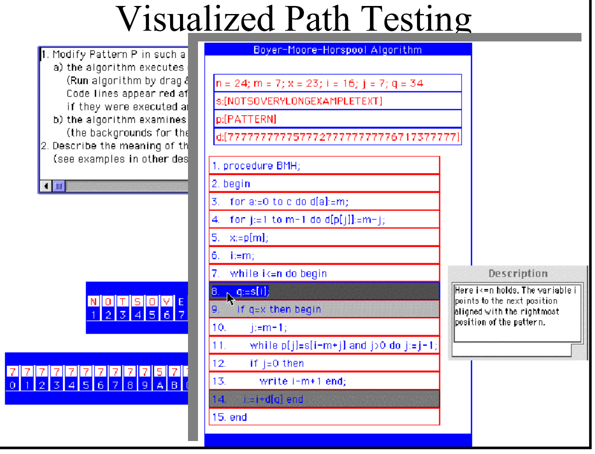
  if b<2 then a=a+7;
- The following test sets of pairs (a,b) satisfy the coverage conditions:
  - Statement coverage: { (0,0) }
  - Branch coverage: { (0,0) , (4,2) }
  - Path coverage: { (0,0) , (0,1) , (4,1) , (4,2) }



Path Coverage



Visualized Path Testing



Design of a Study

- Choice of test persons, test groups
  - two or more groups of participants use alternative means to learn about an algorithm.
- Kind of experiment
  - Controlled experiments aim to assert a causal relationship between factors (i.e., independent variables) and measures (i.e. dependent variables).
- Pre-test
- Post-test

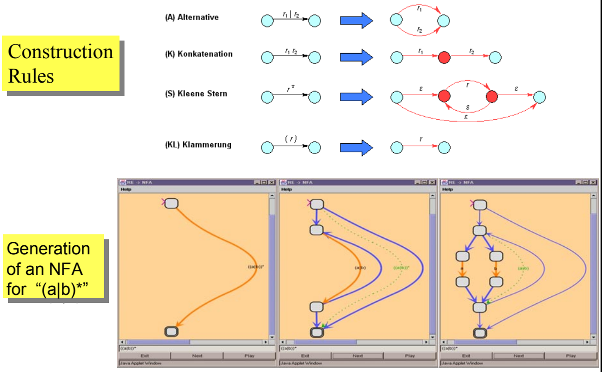
Types of Knowledge

- *conceptual* or *declarative*—an understanding of the abstract properties of an algorithm
- *procedural*—an understanding of the procedural, step-by-step behavior of an algorithm

Typical Questions

- What does it mean that a learning scenario is effective?
  - Comparison with other scenarios
  - Learners‘ knowledge and skills have improved
    - What did they know before ? → Pre-Test
    - Knowledge questions → conceptual and procedural
    - Transfer questions → transfer/apply knowledge in a different context

Example: Generation of Finite Automata



Example Questions

- Taken from an evaluation of the animation of the generation of finite automata.
- Pre-Test
  - Do you know what finite automata are ?
  - Which word belongs to the language defined by the regular expression (ab)* ?
- Post-Test
  - Knowledge Question:
    - Which word belongs to the language defined by the regular expression ab*a ?
  - Transfer Question:
    - We add the notation a^+ to our regular expressions. Give a construction rule for a transition diagram of a NFA ?
  - Open Question:
    - What properties of the animation helped to better understand the generation algorithm ?

Learning Theories

- **Epistemic-Fidelity Theory:** emphasizes the value of a good denotational match between the graphical representation and the expert's mental model.
- **Dual-Coding Theory:** visualizations that encode knowledge in both verbal and non-verbal modes allow viewers to build dual *representations* in the brain.
- **Individual-Differences Theory:** asserts that measurable differences in human abilities and styles will lead to measurable performance differences in scenarios of AV use.
- **Cognitive Constructivism:** asserts that there is no absolute knowledge. Individuals actively construct their own individual knowledge out of their subjective experiences in the world.

Results of some Studies

- Meta study by Hundhausen_et_al
  - More than 40% of the 24 studies considered did not find significant results.
  - „Thus, according to our analysis, *how* students use AV technology, rather than *what* students see, ap-pears
  - to have the greatest impact on educational effectiveness.“
- Several studies found that electronic learning material (multimedia or hypermedia) with algorithm animations outperforms lectures. Comparisons with textbooks are less clear.

Results of some Studies

- The form of the learning exercise in which AV technology is used is actually more important than the quality of the visualizations
- AV technology has been successfully used to actively engage students in such activities as
  - prediction exercises
  - programming exercises

Open Research Question:
Evaluation of Industrial Software Visualization

- To what extent has software visualization been effectively applied in industry?
  - Increased productivity ?
  - Decreased costs ?
  - Support for the large software teams typical in industry?
  - Support for *distributed* programming teams that are common today?