Déjà Vu—Or, Whatever Are We Doing Here?. . .

Hassan Aït-Kaci
School of Computing Science
Simon Fraser University
Burnaby, BC, V5A 1S6
Canada

> The difficulty lay in the form and economy of it, so to dispose such a multitude of materials as not to make a confused heap of incoherent parts but one consistent whole.
>
> EPHRAIM CHAMBERS, *Cyclopaedia*

Abstract

It is said that long ago, in primeval India, there was a kingdom of blind people. These were in all respect endowed with human traits save for sight. In particular, they were warriors. At one point, King Bat, the king of the Blind, thinking of expanding his kingdom's hegemony, sent out (blind) scouts to the neighboring (seeing) kingdom. That other kingdom was a powerful one. Its army disposed of a famed and formidable war unit of elephants. Elephants were unknown to the Blind and their king King Bat. The scouts soon came back, all excited with intriguing news.

"An elephant," said one, *"is like a rough-textured wall!. . . "*

"Not so," said his neighbor, *"an elephant is a snakelike beast that blows foul-smelling air and putrid water in and out of its maw!. . . "*

"Nay!" said another, *"it is thinner than a snake, stringlike, and has a tuft of hair on top of its small head!. . . "*

The last, older, scout, snorted his scorn at his inexperienced younger comrades, and stated unequivocally, *"May your Blindness forgive these three stooges for their poor senses! An elephant is in fact a bunch of trees that hammer the ground with their trunks in turn to frighten their enemies! They have long and sharp low-reaching branches that are sharpened to needlepoints by African elephant trainers. . . "* [1]

The blind Bat blinked, then saw the light: *"We shall ask Our Royal Computer Astrologer of Scient-O-Logic Programming, Holy Mole Ravi Olly, to advise Us. We shall blindly follow what she will tell Us—she shall query her oracle and let the Answer be Our Fate!. . . "* Thus spake King Bat to Holy Mole.

Soon, the solemnly clad Scient-O-Logic Programmer performed a neck-cut and gutted a goat, executing her higher-order priestly functions in a monotone language. *"Aha! Let's see what we have in store,"* said Holy Mole, adjusting her multifocal stained-glass spectacles, *"an elephant is simply an invention of our wicked neighbors to make us believe that they possess that which they do not*—L'arme ultime*!"*

[1] Some historians speculate that those were from Ivory Coast; in French: *Des Ivoiriens!*

Dixit Holy Mole Ravi Olly.

She paused to enjoy the effect, and as an aside, added, *"I submitted a seminal Braille paper on the topic to the Bombay Symposium of Scient-O-Logic Programming last year, but it was rejected. The referees, silly bunch, did not decipher my Braille slates upside right!. . . And now, if you'll excuse me, I have to see to my sabbatical in Egypt. . . "*

Exit Holy Mole Ravi Olly.[2]

Thereupon, everyone applauded, King Bat led the attack, and all but one of the Blind were blindly leveled to platitude by a stoic stampede of myopic elephants who did not even care to see them for it was night. The only survivor of the Blind was Frau Doktor Professor Ravi Olly, on sabbatical leave in Alexandria..

Indeed, Holy Mole Olly went to the Library of Alexandria where she taught at the Research Institute for FUEL[3] Programming. Let us hope that her class' inheritance is not the object of this panel, for what happened to that Library of Science was but blaze and cinders.

History tells us to ask ourselves not to repeat it.. . . We, this panel's actors, should perhaps heed its message?

What follows is an appendix. It is a Graduate Seminar proposal that I am thinking of teaching in Spring. Its contents might be both relevant and revealing. Where does the Logic start or end? I do not know. . . Do you? I include it here in this controversial position paper to illustrate that there is more that meets the eye in the logical functionality of computation and information.

Symbolic Computation

This course is intended as a bridge between Theory and Practice. It will consist of a tutorial on symbolic computation models based on functional, algebraic, and logical calculi, as well as advanced state-of-the-art next-generation programming language research. The main focus will be on explaining how some abstract concepts derived from Logic and Algebra can be used to describe symbolic computation and data structures, and how these concepts may be implemented efficiently. During the course, many existing, as well as original, experimental examples will be studied in detail.

Prerequisite

Although meant to be self-contained, this course will offer optimal benefit to anyone acquainted with senior-level discrete mathematics and computer programming. The only *real* prerequisite in an open mind.

[2] Bump! *"Whoops! Sorry. . . (Ouch! May I ask who rearranged the darned furniture in this store without telling?. . .)"*

[3] FUNCTIONALLY-ULTIMATE EXTENDED LOGIC

Participation

Students will be expected to participate actively in presentations of assigned readings, implementation of experimental programs, and in completing a term paper and/or a programming project.

Text

Most of the material covered will be taken from articles and extensives notes written by the instructor. A (far from exhaustive) list of recommended readings would be (1) Burge's *Recursive Programming Techniques*, [4] or Ullman's more recent book on ML [6], (2) Aït-Kaci's and Nivat's *Resolution of Equations in Algebraic Structures*, [2] [3] (3) Aït-Kaci's *Warren's Abstract Machine* [1], and (4) Pierce's *Category Theory for the Working Computer Scientist* [5]. Anything else related to the outlined topics is also recommended (See references).

References

[1] Hassan Aït-Kaci. *Warren's Abstract Machine: A Tutorial Reconstruction*. MIT Press, 1991.

[2] Hassan Aït-Kaci and Maurice Nivat, editors. *Resolution of Equations in Algebraic Structures*, volume 1. Academic Press, 1989. (Algebraic Techniques).

[3] Hassan Aït-Kaci and Maurice Nivat, editors. *Resolution of Equations in Algebraic Structures*, volume 2. Academic Press, 1989. (Rewriting Techniques).

[4] William Burge. *Recursive Programming Techniques*. Addison-Wesley, 1975.

[5] Benjamin Pierce. *Basic Category for the Computer Scientist*. MIT Press, 1991.

[6] Jeffrey Ullman. *Elements of ML Programming*. Prentice-Hall, 1994.

Tentative Outline

1. Mathematics for Symbolic Computation
   - (a) λ-Calculus
   - (b) Universal Algebra
   - (c) Predicate Logic
   - (d) Category Theory
2. Functional Computation
   - (a) Applicative Programming
   - (b) Interpreting the λ-Calculus
   - (c) Compiling the λ-Calculus (SECD Machine)

(d) Extensions

   i. Delayed evaluation and Streams
   ii. Non-deterministic Computations
   iii. Binding by Matching/Unification & extensions to constraints
   iv. Monads
   v. Labels

3. Algebraic Computation

   (a) Term Rewriting

   (b) Knuth-Bendix Completion

      i. Original Formulation
      ii. General Formulation
      iii. Extension with Constraints

   (c) Termination

      i. Well-founded Term Orderings
      ii. Symbol Relative Weight Inference

   (d) Resolution of Equations (Narrowing, Congruence Closure, Fixed Points)

4. Logical Computation

   (a) Horn Clause Resolution

   (b) Compiling Unification and Resolution

   (c) Equational Logic Programming

   (d) Constraint Logic Programming

   (e) Extensions

      i. Declarative Coroutining
      ii. Logical Object-Orientation
      iii. Non-Horn Logic Programming
      iv. Higher-Order Logic Programming

5. Unifying Principle: Categorical Computation

   (a) Cartesian-Closed Categories

   (b) Categorical Abstract Machine

   (c) Categorical Monads and Accumulators

6. Data Types

   (a) Algebraic Abstract Data Types

   (b) Polymorphic Types (Universal, Existential, and Order)

   (c) Constructive Type Theory

7. Conclusion