Some Thoughts on the Teaching of Informatics Engineering

by: Luis Guillermo RESTREPO RIVAS ©2000
http://LuisGuillermo.com
x@LuisGuillermo.com

NOTES ON THE ENGLISH TRANSLATION:

- This article was written originally in Spanish, thinking of the Informatics Engineering education in Colombia, and was distributed among concerned stakeholders in this country (mainly in Medellín), but lately –judging from some opinions I have read written by Douglas Rushkoff, David Gelernter, Alexis D. Gutzman. and others- I have come to the conclusion that this article touches some issues which are relevant also in the U. S. A. and possibly in other countries.
- I refer to "Informatics Engineering", but this term in the original colombian context may be more or less equivalent to "Software Engineering" in other cultures.

---

1. Introduction

As an observer of the education on Informatics Engineering at our local institutions, I have come to the following reflections, remarking a loss of emphasis in some themes and approaches which I deem fundamental for a good academic level and for the creative development of this discipline in Colombia.

My calling is to be alert against too much pragmatism, in order to not let lose from Informatics Engineering the perspectives brought by three elements which I deem highly important and sometimes seem to be "endangered species"; these are:

- The scientific fundamentals.
- The historic and causal relations among developments in informatics subdisciplines.
- The richness brought by the ability to have diverse options and resources.

I consider that each one of these elements makes valuable contributions, as I will expose shortly, to the academic level and to the engineer's performance even in his/her most pragmatic activities, because these themes gives to the new engineer a knowledge that is actually useful to conceive and design new solutions, and because they allow a more profound and broader comprehension about his/her discipline and also because they enrich the engineer's armory to face diverse situations.

2. The Engineering

Engineering has as definitional core the application of scientific, mathematical –and even empirical- knowledge to the invention, improvement and utilization of systems to the solution of practical problems, or to profit from these scientific knowledge.

I think of engineering as some kind of "bridge" between science and technology. As science does search knowledge without much concern about its eventual application or profit, and technology focuses itself on practical applications of systems, regarding the "fundamentals" only to the depth necessary to perform effectively and efficiently, engineering sits on a middle ground: it avoids to "lose" itself into scientific or theoretical ramblings, because it always must have "its feet on the ground" and a pragmatic view, but it also must concern itself with the theoretical fundamentals, more than technology does, because one of its more important tasks is to think up and design new practical systems, applying scientific knowledge.

Although in its pursuit to develop systems of ever increasing complexity, each engineering branch resorts to an ever-broadening interdisciplinary knowledge source, it is still true that each branch has a core of characteristic scientific disciplines which defines its specialty. In the case of Informatics Engineering, its scientific knowledge body is Computer Science.

3. The Science

As a bridge between science and technology, Informatics Engineering must not overlook that it has significant theoretical and formal fundamentals, based on mathematics, mainly on logic, formal languages and numerical methods.

During his/her formative years, the future informatics engineer –who sometimes seems to lose itself into the mere learning of commercial tools use- must receive a robust education on the fundamentals which make its discipline an engineering, as more or less different from a technology or empirical craft.

This engineering owes its fundamentals to themes as: logic, numerical methods, data structures, automata theory, algorithm theory, graph theory, Shannon's information theory, computational complexity, formal languages, latest developments related to knowledge management, and many other fields.

4. Historical and Causal Development

Each concept and tool utilized by a discipline, be it of internal origin or adopted from another knowledge branch, didn't appeared "from the nothingness", but it has being born from an evolution or development, from a search for solutions, from the improvement, enhancement or combination of existing elements.

So, in Informatics Engineering, one must not overlook this perspective or of conceptual evolution, because only from this perspective the student can get a full comprehension about the historical logic or reasoning processes which brought us to the significant contemporary developments –conceptual and pragmatic ones- we take for granted.

The knowledge of the seeds and roots gives a more comprehensive perspective of the tree.

The student, future informatics engineer, must develop a sort of conceptual structure where he/she can put, in an ordered and interrelated distribution, the knowledge chunks he/she gets.

To mention only a case in point: Object Orientation, as contemporary methodology, would be better understood if it is explained locating it in the context of the development of analysis, design and programming methodologies, along with themes as: algorithms applied to data structures, non-structured programming, the GO-TO controversy, structured programming, SIMULA 67, and the "frames" concept by Marvin Minsky from A.I. research.

Briefly: the history of computers, and the evolution of ideas and tools on computing and informatics must occupy a noteworthy place in the formation of this engineers.

5. Richness of Options and Resources

It is well known that if a man only knows and has a hammer, all he can think of and do is “to hammer”, he even will intend to insert a screw by hammering. If this man also has a screwdriver he will be much more efficient putting screws. But there is an interesting twist: if he doesn’t has the screwdriver, but does have the “screwdriver concept”, it likely that he will be able to adapt some object to shape an approximation to the screwdriver, which probably will be somewhat more effective than the hammer on putting screws.

I consider the richness of options to be of great importance, and actually a measurement of development level. Undoubtedly, when it comes to richness of alternatives regarding concepts and tools, diversity constitutes a valuable armory with which the engineer may more confidently face the problems requiring solution.

The knowledge, maybe not so deep, of a variety of: analysis methodologies, data structures, programming languages, operating systems, human-machine interface possibilities, conceptual tools, etc. opens the mind of the engineering student, to wise up to the fact that there are many other possibilities, other ways to make something, other alternative ideas. This framework helps creativity.

To finish this idea, I will add that there are possibilities in a different paradigm which simply aren’t thinkable in another one that limits our thinking. As an example, lets consider the capabilities offered by the LISP language to manipulate computer source code as if it were data, and to have functions which may take other functions as arguments. These two concepts may be unthinkable for someone who only knows more limiting programming languages.

I know that the richness in physical resources –computers, compilers, etc- is severely limited by the available economic resources, but I consider that the conceptual richness, which depends on information access and thought exercising is indeed attainable, and we must work more on this issue in the formation of new informatics engineers.

6. Summing-up

I know that the three elements which I have exposed in this article: the scientific fundamentals, the historic and causal relations in informatics developments, and the richness brought by options diversity, may be seen as themes located far away from the pragmatic everyday need to be productive soon, and may be regarded as only of theoretic interest, but my viewpoint is that –as I briefly tried to exemplify- these three elements has implications into the robust formation of valid informatics engineers, the ones with creative skills for design, with good knowledge of the theoretic foundations, origin, location and relations of every concept and tool at their disposal, the ones with richness of concepts and tools.

We must strive to get an equilibrium between the pragmatic and the theoretical interests, as it seems adequate to the character of engineering, that middle ground between the science flight and the technology immediacy.

Translated from Spanish into English by the author.