# Logic Programming Didactics

*Demosthenes Stamatis[1], Petros Kefalas[2]*

*[1] Dept of Informatics, Alexander Technological Educational Institute, P.O Box 141, 57400 Thessaloniki, Greece, demos@it.teithe,gr*

*[2] Dept of Computer Science, City College, Affiliated Institution of the University of Sheffield, 13 Tsimiski Str, 54624 Thessaloniki, Greece, kefalas@city.academic.gr*

**Logic Programming (LP) courses are part of many Computer Science or Artificial Intelligence related programmes. In this paper, we present a systematic approach on teaching an LP course, using Prolog as the main computational paradigm. We argue that LP is an excellent didactic tool for teaching Intelligent Programming Systems as well as a vehicle for an in depth understanding of the programming methodology activity as a whole, both declarative and imperative. A student model is defined which in turn is used to facilitate the learning outcomes and process. The model is based on student misconceptions, which were identified using action research derived from our long experience on teaching LP. We demonstrate that, by lifting these misconceptions through specifically designed teaching sessions students are led towards a better understanding of Logic Programming both as a tool for developing intelligent systems and program construction in general.**

### Keywords

Logic Programming, AI logics, Mathematical Foundations

## 1. Introduction: Logic Programming courses in CS Curricula

Computer Science is an engineering discipline, and as such it should integrate a fair amount of mathematical concepts. CS curricula should be designed such that they include introductory courses that relate to mathematics specific to the domain, e.g. Discrete Mathematics including logic, set theory, graph theory etc. [1,2]. Among those, Logic Programming, although not a core in IEEE/ACM Computing Curricula [1], can be found in many CS Departments programmes, especially those with an Artificial Intelligence flavour. It is thought that logic programming, particularly through the use of Prolog as the main programming paradigm, lead to the development of an improved student model that is more capable to cope with programming methodology in general, both imperative and declarative, as well as with Artificial Intelligence techniques, which are normally introduced later in their studies.

It has been long argued that mere possession of knowledge is not sufficient for students in higher education, if the student does not learn how to use this knowledge effectively. To do this, a learner must possess certain intellectual or cognitive processing skills, e.g. the ability to analyse, synthesize or evaluate. These are clearly identified in Bloom's taxonomy of educational objectives, in which the major categories in the cognitive domain are listed, i.e. knowledge, comprehension, application, analysis, synthesis, and evaluation [3,4]. Logic programming courses offer the opportunity to develop such skills in the context of program construction.