

## Breaking the Programming Obstacles using an Automatic Tool

Luis Humberto González Guerra, Armandina J. Leal Flores  
*ITESM – Monterrey Campus México*

### Abstract

*There are some programming obstacles that the students have to deal when they are trying to solve a problem using a programming language, like the inexperience of the language, the unknnowledge of some algorithms, and some cases the amateurishness of how to test a solution. The technological tools for developing programs and computer systems heavily explore the syntax and semantics of a program but do not provide support to determine its correctness and this is where students may failed by not exploring the correct scenarios and test cases. In this paper, we present the use of automatic evaluation tools as a way to help to remove these obstacles and improve learning experience, coding and testing skills in students.*

### 1. Introduction

In order to develop strong programming skills, it is critical for students to be constantly challenged by task and problems involving computational solutions [6]. Generally students face a large number of these during the semester [4] so they can build and acquire the necessary skills and techniques needed in program development [1]. To do their jobs, students use development tools for writing code and building computer systems. This set of tools reduce code related issues and strongly explore syntax and semantics of a program, but do not provide support in the analysis of the solution of the problem, this being a major drawback since, most students verify their solution with few cases -usually the most basic to the problem in question- and therefore produce incorrect or incomplete solutions [5]. Some of these exercises are made inside of the classroom where the teacher could guide to the students, and outside of the classroom the student made them without any support or help, and they do not could know how their solution is.

In this paper we present how automatic evaluation software can provide teachers and students a way to improve the programming learning experience, coding and skills, and how a tool like this can complement the problem-solving skills to students, and break programming obstacles inside and outside of the classroom.

### 2. Background

As part of the programming courses professors suggest using one of the multiple development tools on the market such as Eclipse, Visual Studio, XCode, among others, as they rely heavily on the development of programs from the perspective syntax and semantics in addition to which may be employed for programs in multiple languages. For several semesters we have seen that most of the solutions of our students are not entirely correct, as they usually poorly tested solutions by only working with the basic scenarios and not exploring the results on corner cases. Based on this fact, we researched on platforms that support the evaluation of the solutions given by students to their assigned problems. The tools found provided a set of problems that students could solve in several programming languages and validate their solutions by running well designed test cases. Through its use began to show good results with the students' performance, there was a major inconvenience: the existing tools do not allow professors to add new problems to the database so finding problems suitable to the multiple objective of the course became a hard task. Given this problem and aware that the automatic evaluation supports to solve the above [2, 3], the development of computer system SEPAP (Spanish acronym of "Evaluation System for Learning Programming") started with the aim of helping students in reviewing their solutions. This tool has the functionality to verify the student's solution with multiple test cases including extreme cases where the student fails often related to their lack of experience in some areas.









