Games Writing as an Instructional Tool

Carlos M Gameros

The goal of the introductory courses in the Computer Science department is to teach students the basic principles of computing, including data structure design, basic algorithms, and control flow language constructs (e.g., loops, selection constructs, sequences, etc). Unfortunately, designing *interesting* assignments to re-enforce lessons on these subjects is a difficult task. On one side, creating problems that simply demonstrate the use of these concepts often leads to boring assignments. On the other side, creating more elaborate problems of greater interest may result in assignments that are too complex for beginning level students to solve. One way to design problems would be to compromise between the two development methods above – this was accomplished through a "Tetris" clone and through the use of abstractions.

In the Fall of 1995, we proposed to change the structure of the Introduction to Computer Science Labs. In the past, programming assignments were typically collections of unrelated assignments that taught basic principles. For example, students might work on a graphing lab one day and then an addressbook lab the next day. These assignments may prove to be boring and tedious and fail to keep the attention of many students. In order to bring unity to our lab assignments, a "Tetris" clone was constructed. This program will be used as a basis for creating our lab assignments.

In order to understand how abstractions can be useful to creating introductory level class assignments, it is important to understand the concept of an abstraction. An abstraction is a subroutine or collection of subroutines designed to perform complex tasks. Its purpose is to allow a user (i.e., programmer) to perform complex actions while ignoring the implementation details of the action. For example, Turbo Pascal provides an entire unit devoted to graphics routines, such as a subroutine to draw a circle on the screen. A programmer need not know the details of how pixels are set, how points on the circle are determined, etc., in order to use this subroutine. He or she only needs to give information about the circle dimensions. The details of how the circle is actually drawn is hidden from the programmer.
The "Tetris" clone serves three purposes:

1) Labs will be based on the game in which assignments will be to write various lab modules that will fit into the clone.
2) Labs will be more enjoyable and will still teach basic programming and data structure principles. For example, predicate logic can be incorporated in teaching array manipulation of the games and recursion can be used in row removal algorithms.
3) Labs will make use of the abstraction concept to do complex tasks.

The use of abstractions has proven to be an especially helpful tool in the development of introductory level assignments. Students find the assignments to be more interesting without being too difficult. By keeping students interested in their work, we are able to increase their understanding of concepts which are important to future work in the field of Computer Science. There are many possibilities in selecting lab assignments and they will be tested this Fall.

Acknowledgments. The author was partially supported by NSF Grants No. CDA-9015006 and No. CDA 9522903.

Department of Computer Science, University of Texas at El Paso, El Paso, TX 79968, email cgameros@cs.utep.edu.