Teaching Principles of Big Data in Fundamentals of Computer Programming II

Christian Servin[1], Jason Ivey[2] and Jacob Loosa[3]
Computer Science Program & Info. Tech.
El Paso Community College
919 Hunter Dr.
El Paso, TX 79915, USA
[1]cservin1@epcc.edu, jason.ivey.not.michaeal[2], jacob6816@gmail.com[3]

Big Data is a contemporary trend in computing that has been expanding in many different computing areas. Some of the Big Data challenges include data capturing, storage, search, and sort besides many others. Principles of big data can be slightly introduced in the *Elementary Data Structures and Algorithms* course, also known as CS II.

Although many assessments or practices such as computer labs in traditional CS II courses may be designed to increase programming skills and/or enhance the computational thinking, often these assessments do not necessary reflect direct application to real-life scenarios in the workforce.

Common factors that may influence the lack of real-world scenarios in the courses include:

- Number of contact. Contact hours through lecture and lab is mandated by the curriculum requirements in a specific state. For example, the field of study degree in computer science in the state of Texas is approved by the Texas Higher Education Coordinating Board (THECB) under specific recommendations;
- The CS programs/degrees in different colleges. Computer science degrees may be found under a variety of colleges and/programs in different community colleges, liberal arts or junior colleges. Or in the case of the universities, often the degree of CS may fall under the college of science, engineering or liberal arts, having a different perspective about what learning outcomes are more significant for the education of the students.
- Conventions, standardization, and accreditation. In particular institutions, the computer science degree belongs to the college of Engineering, then there are well-known standards that each institution may need to follow in order to be in compliance with Accreditation Board for Engineering and Technology (ABET). For example, UTEP follows the ACM/IEEE CS2013 Curricular Guideline, which offers 18 Knowledge Areas that compose the body of knowledge in Computer Science Education.

In order to avoid generalization or deviate from factors already mentioned above, we designed programming labs that are appropriate for CS II, with infusion of Big Data. These learning outcomes are based on the ACM Committee for Computing Education in Community Colleges CCECC recent released new document: *Computer Science Curricular Guidance for Associate-Degree Transfer Program, with Infused Cybersecurity.*

In this presentation, we present several examples on assessments that can help students understand particular topics in CS II through real-world scenarios. The CS II topics include recursion, searching, sorting, and two-dimensional array manipulation. These assignments were originally designed by peer-leaders and honors students.