A Framework of Preparing Minority Students in Computer Science for the Industrial Career

Yujian Fu
Department of Electrical Engineering & Computer Science, Alabama A&M University, Normal AL, U.S.A.

Abstract— *The lack of preparedness of Computer Science graduates for a professional career is a common complaint raised by industry practitioners. The career progression of many new graduates is severely impacted due to the lack of problem solving capabilities and well rounded skills. For example, some of the technically stronger graduates lack communication and managerial skills and vise versa. Industry based capstone projects, incorporated as a part of an undergraduate degree, are a well accepted means of preparing students for their professional careers. This paper using a curriculum design of object oriented implementation class of graduate students at the Alabama A&M University with engaging in industry based projects at the first year of their degree to illustrate how to increase student motivation, improve software development capability, increase problem-solving capability and speed up the skills and knowledge of engineering discipline. The project developed in the curriculum aims at providing students a real-world software engineering development experience, improving student programming skills and project management capability and object oriented design concepts, and preparing them for industry. At the end of semester, we observed a successful achievement of these projects in giving aspects of evaluation. We believe this achievement is due to the practice on the object oriented programming language and project design framework. Finally, we believe this approach can be extended to any other objective problem solving framework, focusing on managerial, engineering and science classes.*

Keywords: Problem solving skill, capstone project, LEGO mindstorm, rentention

1. Introduction

Helping under-represented populations pursue STEM (science, technology, engineering, and mathematics) careers is a challenge issue now. Research shows an expanding ethnicity gap for Americans pursuing STEM careers. A new report from the National Action Council for Minorities in Engineering (NACME) reveals that the number minority students pursuing STEM degrees and careers has flattened out or even declined in recent years with the increasing requirement of workforce in IT professions [24]. Typically in Alabama A& M University, we have observed 66% drop of graduation rate from 2005 to 2013 for undergraduate students, and 29% drop for graduate students in computer science department. This symptom is due to several reasons. Loss of attraction of the major is one of the main reason. What students have learned are not useful and what are expected in industry are not taught in class – there is a gap between classroom teaching and career marketing. To solve this problem, encourage and support minority students in pursuing STEM careers, we need to a) promote classroom teaching in tangible and real-life oriented ways and b) promote fun ways to explore STEM interests and attract more students [26].

Real world projects and capstone projects are well accepted by industry practitioners, professional bodies and educators as necessary for providing practical experience to science and engineering students to prepare them for a professional career [11]. Such projects aim to consolidate the theoretical knowledge gained through subjects in related areas by practical application. Real world projects are particularly important in Object oriented design and programming course, Software Engineering (SE), and more advanced courses (e.g., software testing) in graduate level in minority institution, since most of the theoretical concepts in these courses are abstract in nature, many students are not ready for professional career preparation and, therefore, are hard to understand without adequate hands-on experience. Therefore, it is common for computer science courses to include capstone projects âĂŞ in some cases real-life software development projects âĂŞ as a part of both the undergraduate and graduate degree.

The curriculum of Master's of computer science degree at the Alabama A&M University includes both software engineering and object oriented programming as core courses. In addition to fundamental computer science subjects these courses cover both theoretical aspects, as part of their degree, students engage in year-long, industry-based projects. These projects are aimed at preparing students for their professional careers, as competent software engineers, through a real-life project experience. It is a well accepted fact that a competent software engineer requires a wide variety of skills such as managerial, engineering, team working and communication [2], [23]. One of the important learning outcomes we are interested in achieving from these project experience is the right balance between the depth and the breadth of problem solving skills. Over the past years we have tried various schemes for bringing about these learning outcomes with

varying degrees of success.

In addition, it is worth to observe that computer science (CS) has lost its appeal, and robots can help find it. Even as computing has invaded every aspect of our lives, CS as a field of study is often seen as disconnected from these same lives. This is a key issue especially in minority institutions [31]. There over 40% students dropped from CS major in the past 20 years [33], [12], while there are over 60% students dropped from cs in AAMU, comparing to the increasing of CS jobs in the past. So to reestablish the connection, the Institute for Personal Robots in Education (IPRE, www.roboteducation. org) is developing a personal robot, software, and curricula to help teach introductory CS courses. Imbued with the proper pedagogical philosophy and training, it can help make CS more personal. Though there are islands of hope, overall, fewer students are enrolling in CS courses, staying with them, or moving into industry because we have washed them out of the classroom and the pipeline.

We believe that an objective framework built on the industrial oriented project (in our curriculum is robot project) to guide student learning is an important element for bringing about the expected learning outcomes through projects. The absence of such a framework tends to leave the students unguided, resulting in students not getting the right balance between the breadth and the depth of skills. Based on the experience gained from teaching project based subjects over several years, we have developed a skill-based objective framework, focusing on managerial, engineering and personal skills. In this paper, we present this framework and share our experiences of using it.

The rest of the paper is structured as follows. Section 2 introduce background of framework in two aspects and an overview of computer science education using robot projects. Section 3 presents the problem solving skill based framework with problem statement. Section 4 introduces the project design and observings in the problem solving based framework. Section 5 describes our early experiences of using the new framework and results analysis. Finally, section 6 concludes the paper.

2. Background

In this section, we will discuss the related works of problem solving and the basis of cooperative learning. Finally, an introduction of CS education in AAMU was described.

2.1 Problem Solving

Problem solving skills have been typically discussed in the engineering and science education [18], [27], [25]. Because the multidimensional requirements of the engineering program requires students to be able to handle diverse problems using multiple skills and knowledge. Inspired from engineering and science strategies, Several works from computer science education are established in the introductory or senior curriculum and program [9], [19]. A recent work of open-ended problem solving skills [21] used in the fluid engineering was discussed. In this work [21], a three step process of problem identification & formalating, skills teaching and data collection and analysis is presented. This paper have a definition of problem solving skills for each sample question. An alternative approach to problem solving skills was discussed in the work of [3] by Allan and Kolesar. Their approach is to introduce a preliminary course in general problem solving before teaching programming concepts. However, lack of problem solving capabilities among minority graduates is a serious problem raised from industry while was paid less attention by educators and researchers. Due to lack of background, inefficient learning, and other issues, minority students does not have motivation and therefore loss the interesting [32] in computer science. This has been reflected in the retention issue among minority institutions [30], [22]. A data from AAMU has shown up to 47% drop from sophomore to junior in computer science major [14]. How to improve student's problem solving capability in computer science is a key issue for student's career development and industry satisfaction.

Commonly, robots are used in introduction to computer science courses. Many institutions have introduced robot concept and implementation to the classes. Lego Mindstorms have been adopted for the introductory computer science course in George Washington University to teach problem solving using computers and applications of computer science to the real world [20]. Likewise, Lego Mindstorms and the LeJOS system were used in the introductory computer science course at Cansius College [5]. To teach basic computer science concepts such as control flow and the use of arrays, the United States Air Force Academy has studied using a previous version of Lego Mindstorms to teach an introductory course using Ada [10]. At some colleges, the visual programming language used in the Lego-supplied software to teach the concepts of computer science without making students who may be unfamiliar with a programming language to learn the syntax of a high-level language like C++ or Java, two of the languages that are commonly used in introductory computer science courses [6].

Robots have also been used in single-topic computer science courses. One of the areas that robots are most often used for pedagogical purposes is the field of artificial intelligence. At Williams College, robots are used to introduce students who may not be computer science majors to the concepts of artificial intelligence and behavioral programming through the use of robotic vehicles and a high-level language [8]. The City University of New York, Staten Island, also uses robots to teach artificial intelligence, but the course is tailored for third and fourth year computer science majors and requires a proficiency in C++ [13]. At the University of Salzburg, robots are used to teach embedded software engineering,

covering real-time operating systems and communications protocols, scheduling, real-time programming, and code generation [16] (this link does not work). Villanova University has used Mindstorms robots to teach computer organization and operating systems concepts [17].

A common feature in both the introductory courses and the single-topic courses is that none of them require students to design a software artifact from a specification using software engineering practices. In the introductory courses, both for computer science majors and non-computer science majors, the projects are rather simple, like programming a robot to follow a line by using a light sensor, or pushing cans out of a dark area of a board [6]. While it is common for groups to be formed in introductory classes that use robots, the projects do not require, nor do introductory classes teach, software engineering concepts like specifications elucidation, design practices, or verification. Additionally, these classes do not attempt collaboration between groups. In the advanced, single-topic courses, the focus of the course is on a particular concept, and while it requires the use of sound software engineering practices, the course treats knowledge of such practices as a prerequisite [16].

2.2 CS Education in AAMU

Computer science department is one of the oldest department at Alabama state with the initiate of computer science education. With the past decades development, the department has both Bachelor and Master's degree with over 200 enrollment each year. The number is fluctuated with years. It is estimated to be around 30% to 70% African American students enrolled into computer science department Master's program roughly. There is a dramatic drop of enrollment of Master's students for the past 20 years. At the same time, the quality and recruiting of Master's students to computer science department has been a challenging issue since past years – it is necessary to maintain students numbers and improve student high level skills based on the undergraduate training background.

This problem-solving framework is especially designed for minority institutions with low enrollment of computer science students. Although Master's students are more motivated than undergraduate students, many students that do not have enough computer science background still feel very hard and boring about the current content. Secondly, the feedback from the industry of Master's student expects more solid skill and problem solving capabilities for graduate students than undergraduate students. Finally, for the course we teach, it is expected to have more interesting projects than just the theories and background knowledge. In addition to it, traditional projects cannot satisfy current needs and students' expectation. More challenging and interesting projects are highly desirable.

Due to the above typical characteristics of minority education in computer science and especially software engineering education, we propose this new interleaving framework to target problem solving capability of minority students in the software engineering education. Similar as the goal of the work in [15], we both prepare students to meet for industry requirements, which is one of the typical goals of education. Observing that it is hard to realize most of blocks in the framework of [15] in our school due to the above characteristics, we have two solutions on the table. One possible solution is improving our student quality by recruiting students with strong background and facing a dramatically drop down of student enrollment. Another possible solution is to improve our teaching strategy and adopt a new framework that works for the minority group. We believe that, after three years of implementation, our framework fits for the graduates of minority students with the above features.

3. Problem-solving based Framework

The main goal for introducing the problem solving-based learning framework is to provide objective guidance to ensure that the team based projects provide the expected software engineering skill outcomes with a proper balance between the breadth and the depth of problem solving skills. The new framework addresses the problems in the previous scheme while retaining some of the positive aspects. Project based problem solving framework includes four phases – problem identification, solution searching, solution carrying out, and validation & feedback. The main goal of the four phases is to provide stepwise management for the team based projects and instructions to the expected software engineering skills.

3.1 Framework

The Object oriented programming and design class at Alabama A&M University provides "various object oriented methodologiesâĂİ as part of its program capability and career support contribution [1]. In a similar vein, [7] also proposes a generic standard focusing entirely on the abilities and skills that graduates will develop as part of their bachelorâĂŹs and masterâĂŹs level programs. In line with this literature, our problem solving-based framework also identifies a detailed list of software engineering and development skills that computer science graduates must possess. The framework is categorized as four phases – problem identification, solution searching, solution carrying out, and validation & feedback. These are the sequential steps students choose to specialize in during the projects as described in the next section. A schematic representation of the skill-based framework is shown in Figure 1.

There are two facilitate streams around the four-phase. In other works, each phase is supported by pedagogical goal and studied by the research issues. The *problem identification* aims at selecting project topics from the real world

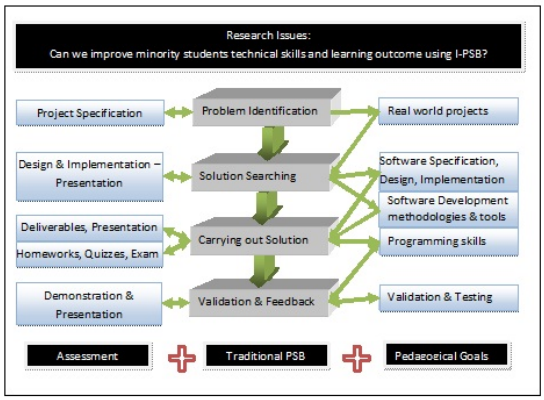

Fig. 1: Framework

problem domain, design and adapt it to the class for the students. The research issues of this phase are:

- Can this project motivate students?
- Are students interested in the project or the series of projects?
- How does this project affect student's future career?
- Can this project attract more students?

The *solution searching* involves knowledge-based skills and application abilities. In this class, the software design, development methodologies and certain CAD tools are the fundamental purposes. The key issue in this phase is how to improve the student problem solving capability and help them to identify and choose the proper methods to realize the design. This is a critical phase for the capstone project, while iterated and incremental instruction will benefit student to make decision. *Carrying out solution* is a student-leading activity. Application of knowledge and methodology will be reflected with programming capability as well as the conceptual learning and critical thinking of the content. The project is designed as a team project so that student will have experience for collaboration work. Project management such as analysis, scheduling, and progress control are part of the goal in this phase. The evaluation of learning outcome does not include the results. Other assessment methods, such as required document, presentation and project demo, will help to assess the evaluation.

One of the key goals of the framework is to provide students with the right balance of problem solving capability from all domain areas. To facilitate this, students are asked to choose at least one problem domain from the provided projects and one development method from the provided methodologies. Students must fulfill all requirements of the selected project as these are aimed specifically at individual development. It is important to note that choosing certain development method does not constrain students from demonstrating their abilities in other areas.

3.2 Assessment

In addition to providing students with the industry oriented project and domain related topics, our problem solving-based framework also paves an objective path to assessment. The artifacts that submitted projects by students for their chosen topics in different pedagogical areas, along with related research issues and assessment criteria, form an objective and quantitative means of assessing students. The systematic assessment model also aids instructors to assess student performance on a fortnightly basis. Trends in student performance and the improvement shown over the course of the project can be extrapolated from systematic assessment and used for assigning an accurate final grade at the completion of the project. Such an objective framework lowers the risk of an instructorâĂŹs subjective views being induced into assessment, thereby providing a consistent grading scale across all project teams.

The grading is divided into four main components: project design component; project implementation component; project validation and presentation component; and demo component, each is weighted 25% of the final grade. The component grading criteria include: project specification, management, schedule, engineering, quality, and documents. The component weight are 10%, 15%, 15%, 25%, 15%, 20%, respectively. Although the project is assessed on the group based, each individual's contribution is also considered and expected to be identified based on the underlying skill. Such a distribution equips students with a well-rounded know-how of technical and non-technical software design, development and engineering issues.

4. Project Design and Observation

The projects are conducted and categorized as two groups – experimental group and control group. To reduce bias and increase the observation, all projects are controlled in the one domain in the year 2013 – embedded systems. The experimental group focus on the embedded system design and implementation with equipments, while control group works on the simulation of embedded systems using

web technology or database. The projects of experimental group are designed on embedded system with the industrial based scenarios. All projects are modified based on the industrial advisory board members. The projects of study group includes the information system design and web design projects. There are 33% students chose the embedded system design and development based on their experience and background. Others chose information system design. A qualitative study of embedded system is given to participated and unparticipated students. The survey, including two aspects of the class project design and learning framework of the class, was given in 40 related questions at the end of the semester. We received very positive response from student survey. A roughly description of embedded system design project is given as follows.

4.1 Project Design

The industrial advisorary projects are based on the educational version of the Lego Mindstorms NXT system, which is available through Lego Education. It contains a programmable brick, three motors, one light sensor, one ultrasonic (sonar) sensor, one sound sensor, and two touch sensors. The programmable brick contains three output ports used to power the motors, and four input ports used to connect sensors. Legacy support is included via converter wires that connect to the NXT programmable brick and legacy 9V motors and Robotics Invention System sensors. [4].

The Mindstorms NXT brick uses a 32-bit ARM processor as its main processor, with 256 kilobytes of flash memory available for program storage and 64 kilobytes of RAM for data storage during program execution. To acquire data from the input sensors, another processor is included that has 4 kilobytes of flash memory and 512 bytes of RAM. Two motors can be synchronized as a drive unit. To give the robot the ability to "see,âĂİ the ultrasonic sensor, which is accurate to 3 centimeters and can measure up to 255 centimeters, and the light sensor, which can distinguish between light and dark, can be attached to the brick. A sound sensor that can be adjusted to the sensitivity of the human ear can be used to give the robot the ability to hear and react, if programmed, to noises. Finally, the two touch sensors give the ability for a robot to determine if it is pressed, or bumped, and react accordingly [4].

As a replacement for the standard Lego firmware, the LeJOS project has support for threading, arrays, recursion, synchronization, exceptions, non-generic data structures, standard data types, and input and output [29]. The LeJOS virtual machine supports much of the java.util package, but the data structures require that data be stored as type Object and then cast to a type that inherits Object. For input and output, both streams and sockets are available for use which is defined in the java.io package. For control purposes, the LeJOS platform supports the direct connection of Bluetooth-enabled GPS units for spatial location information and keyboards for the navigational control of a robot [28].

Table 1 lists three (3) project modules based on LEGO Mindstorms NXT.

4.2 Observing Problem Solving

Project management and the application of software modeling and testing concepts enabled students to observe the construction of the building a quality software system. To integrate *Problem Solving* into our curriculum, we have developed new projects, with a systematic assessment and evaluation procedure that explicitly specified in the syllabus. The only difficulty of this graduate class is that different students have different background in programming languages and skills. This is a typical situation in graduate courses in many HBCUs. Some even worst case is that a few students do not have enough programming skills and have to learn and compose programming concepts, to solve more abstract problems. In this special case, what the instructor has to do is to reduce the workload for the specific student and prepare make-up lectures. To make more students benefit from the all provided materials, all materials are available on line, as well as we insert a lecture dedicated to the demonstration of live problem solving.

5. Experience, Analysis & Discussion

All three projects can be finished within semester time, including new API learning, environment setting up, and programming and testing. There is one project was updated during break with more elaborated functions. The projects are very attractive to students. One student even got CO-OP from a company after introducing this project that he had been working on.

a) Experience: The computer science department of Alabama A&M University is an ABET credit and offers both Bachelor and Master degrees with enrollment of over 200 each year. Object oriented programming and design is offered in each fall semester following with software engineering methodology for first graduate students. Usually there are 50% to 75% African American students enrolled into graduate program. Students feel that it is hard to understand object oriented design such as UML (Unified Modeling Language). It seems it is not useful in many systems for the future especially in industry. Facing this problem and keeping the program outcome of computer science department in mind, the instructor has upgraded the class with real world project with consultant with industry board and discussing with education and curriculum department.

This paper is to show the experience of using the problem solving based framework, to improve student skills in the software development and design area and fit for the future

Table 1: Summary of modules based on LEGO Mindstorms.

Modules	Description
T7. Obstacle Removal	This module uses multiple sensors to achieve obstacle detection and removal. This project includes two tasks: programming Bluetooth-based remote control to the robot car and using ultrasound and touch sensors to detect and remove objects from the path. Students will practice object and class creation, method definition, event handling, control structures, streams, and array.
T8. Tracing and Yielding	This module traces a route with a light sensor, and yields obstacles with a touch sensor. Two tasks will be completed: route cues (e.g., strips) along the route are recognized, when stopped by an obstacle a new path is selected. Students will practice object and class creation, method definition, control structures, loop, and event handling.
T9. Automatic Navigation	The module implements a dynamic navigator that reaches the predefined destination. Path is decided based on images and signals from on-board camera and sensors. Shortest path is generated on-the-fly. Students will practice object and class creation, method definition, control structures, loop, and event handling.

career. The projects are developed on the LEGO Mindstorm NXT educational version. Three group projects are experimental group with industry oriented projects that is built on the LEGO Mindstorm NXT toolkit and implemented in Java/LEJOS. The projects were designed by consulting the industry advisory from companies. Some students had received notice from other company, and continuous communication between the company and students regarding to the internship and job openings are discussed. All students from three groups had shown much enthusiasm for their projects. With the continuation of the project, the students are more and more involved into the projects. In addition, some students (2) from control groups had expressed the idea of changing project to these robot projects.

b) Evaluation & Analysis: Our experience is that students are creative when they are allowed to be. If students feel that there is only one right answer to a problem, then they may feel restricted when experimenting on their own until they have been affirmed with selecting the right approach. When presented with the idea that multiple solutions can be correct, but have different tradeoffs, then they are more confident in exploring on their own. This could result in greater motivation to learn as well as understanding and thus manifest itself as improved learning outcomes for students.

Analysis of the course results using student experience surveys indicate that students feel more engaged with the course and the subject matter since the problem solving teaching has begun. Because the new framework has been in place for only one year, we are not in a position to perform a comprehensive evaluation. The qualitative and quantitative analysis are conducted at the end of each semester, feedback from students has been collected and the results are all positive. To reduce the bias, a set of background questions (question from 1 to 10) were given to students to evaluate their experience in the industrial, project design and implementation, programming on system platform, level and interests, etc.. Quantitative feedback (question from 11 to 30) was obtained through a set of student questionnaire at the end of each semester. Students are expected to assign a score ranging from 1 to 5 for each question, 1 being strongly disagree and 5 being strongly agree. Table 2 shows the averaged scores obtained for the subject in semesters 1 and 2, 2013 for evaluation questions applicable to the problem solving-based framework.

High scores for question 11 and 20 indicate that the main goal and research issues of the framework of achieving appropriate learning outcomes through providing clear expectations. Questions for 21 to 30 targets the class contents handling for the framework and organization. One of our main concerns when introducing the framework was of possible de-emphasis of the value of team work due to excessive focus on individual skills. A score of 5 out of 5 for question 19 clearly indicates that the team aspect still remained strong. The score for question 14 indicates that the intellectual interesting of the project stimulate and motivate student.

6. Conclusion

Real-world projects involving real-life project from industry clients are aimed at providing computer science graduates with the necessary skills required for their industry careers. Our experience in teaching computer science and software engineering subjects over the past years shows that the lack of an objective framework to guide student learning during such projects resulted in students not gaining the adequate breadth of software development skills that fulfill the program outcome of the department, and required by industry. This motivated us to develop a problem solving-based learning framework that aims to equip minority students with a well rounded knowledge of technical and nontechnical skills required to thrive in industry. In this paper we presented the problem solving-based framework and our experiences of using it for first year graduate students which are 70% African American students in software development projects at the Alabama A&M University in 2013 academic year. Overall, the application of the framework was a success from the perspective of students, staff and fulfilled its major purpose of providing students with an adequate breadth and depth of software engineering skills. In the coming years

Table 2: Evaluation of The Framework

Questions	Means
11. The project is effective and valid for student to learn the software design and development methodology.	5
12. The project is beneficial for student careers.	4.5
13. The work load of the project is appropriate .	4.0
14. The project topic is very interested and practical for the current class.	5
19. The team work in the project is successful and management of project provides us a lot of experience.	4.5
20. The learning resource and instruction of the project are available any time to the students.	4.5
30. The project topic is more interesting comparing to the peer projects in the peer classes.	5

we plan to improve the framework and its implementation to address some teething problems experienced and reported in this paper.

Acknowledgment

The authors would like to thank all reviewers for the kindly comments and suggestions on this work.

References

[1] AAMU. Aamu. Available from http://www.aamu.edu/gradstudies/, 2009.

[2] M. R. Ali. Imparting effective software engineering education. *SIGSOFT Softw. Eng. Notes*, 31:1–3, July 2006.

[3] V. H. Allan and M. V. Kolesar. Teaching computer science: a problem solving approach that works. *SIGCUE Outlook*, 25:2–10, January 1997.

[4] B. Bagnall. *Maximum Lego NXT: Building Robots with Java Brains*. Variant Press, 2007.

[5] D. Burhans, R. M. Meyer, P. Van Verth, D. Puehn, V. Steck, and J. P. Wiejaczka. Introductory computer science with robots. In *proceedings of the 21st national conference on Artificial intelligence - Volume 2*, pages 1962–1963. AAAI Press, 2006.

[6] D. Cliburn. Experiences with the lego mindstorms throughout the undergraduate computer science curriculum. In *Frontiers in Education Conference, 36th Annual*, pages 1 –6, oct. 2006.

[7] A. J. Cowling, R. Court, and P. Street. What should graduating software engineers be able to do. In *Proc.16th Conference on Software Engineering Education and Training, IEEE Computer*, pages 88–98. Society Press, 2003.

[8] A. Danyluk. Using robotics to motivate learning in an ai course for non-majors. In *AAAI spring symposium*, March 2004.

[9] David J. Barnes, Sally Fincher, and Simon Thompson. Introductory Problem Solving in Computer Science. In *Proceedings of the 5th Annual Conference on the Teaching of Computing,*, pages 36–39, August 1997.

[10] B. S. Fagin, L. D. Merkle, and T. W. Eggers. Teaching computer science with robotics using ada/mindstorms 2.0. In *Proceedings of the 2001 annual ACM SIGAda international conference on Ada*, SIGAda '01, pages 73–78, New York, NY, USA, 2001. ACM.

[11] S. Fincher and M. Petre. Beyond anecdote towards real transfer: using other institutions' experience of project work. In *Proceedings of the 6th annual conference on the teaching of computing and the 3rd annual conference on Integrating technology into computer science education: Changing the delivery of computer science education*, ITiCSE '98, pages 86–89, New York, NY, USA, 1998. ACM.

[12] N. C. for Women and I. T. (NCWIT). Ncwit. Women and Information Technology by the Numbers. http://www.ncwit.org/pdf/BytheNumbers09.pdf, 2009.

[13] S. P. Imberman. Teaching neural networks using lego handy board robots in an artificial intelligence course. In *Proceedings of the 34th SIGCSE technical symposium on Computer science education*, SIGCSE '03, pages 312–316, New York, NY, USA, 2003. ACM.

[14] A. IROSP. Aamu factors. Available from http://www.aamu.edu/gradstudies/.

[15] S. Karunasekera and K. Bedse. Preparing software engineering graduates for an industry career. In *Software Engineering Education Training, 2007. CSEET '07. 20th Conference on*, pages 97 –106, july 2007.

[16] K. Kirsch. Embedded software engineering course. University of Salzburg. http://cs.unisalzburg.at/ ck/teaching/EECS290O-Spring-2002.

[17] F. Klassner and C. Continanza. Mindstorms without robotics: an alternative to simulations in systems courses. In *Proceedings of the 38th SIGCSE technical symposium on Computer science education*, SIGCSE '07, pages 175–179, New York, NY, USA, 2007. ACM.

[18] D. E. Knuth. Structured programming with go to statements. *ACM Comput. Surv.*, 6:261–301, December 1974.

[19] Z. Michalewicz and M. Michalewicz. *Puzzle-Based Learning: An Introduction to Critical Thinking, Mathematics, and Problem Solving*. Hybrid Publishers, May 2008.

[20] B. Narahari. CS 1: computer science orientation. The George Washington University School of Engineering and Applied Science http://www.seas.gwu.edu/ bhagiweb/cs1.

[21] J. R. Nikos J. Mourtos, Nicole DeJong Okamoto. Open-ended problem-solving skills in thermal fluids engineering. *Global Jounal of Engineering Education*, 8(2):189–200, 2004.

[22] K. E. Nugent, G. Childs, R. Jones, and P. Cook. A mentorship model for the retention of minority students. *Nursing Outlook*, 52(2):89 – 94, 2004.

[23] T. Nunan. Graduate qualities, employment and mass higher education. In *Proceedings of the HERDSA Annual International Conference*, 1999.

[24] C. on Professionals in Science and T. (CPST). Engineering degrees earned by under-represented minorities (urms). http://www.cpst.org/URM_Eng.pdf, 2006.

[25] G. Polya. *How to Solve It: A New Aspect of Mathematical Method*. Princeton University Press, 2004.

[26] S. Progress. Making robots personal. http://sciencedprogress.org/2009/04/trower-interview-robots/, 2009.

[27] D. D. Riley. Teaching problem solving in an introductory computer science class. *SIGCSE Bull.*, 13:244–251, February 1981.

[28] L. Team. Nxj api. http://lejos.sourceforge.net/nxt/nxj/api/index.html.

[29] L. Team. Nxj technology. http://lejos.sourceforge.net/nxj.php.

[30] H. L. K. A. Tekian A, Han Y. Do underrepresented minority medical students differ from non-minority students in problem-solving ability? *Teach Learn Med.*, 13(2):86–91, 2001.

[31] J. Vegso. Interest in cs as a major drops among incoming freshmen. *Computing Research News*, 17(3), 2005.

[32] J. Wales. Achievement gap in the united states. http://en.wikipedia.org/wiki/Achievement_gap_in_the_United_States, 2011.

[33] C. S. E. Week. Computer science education week. http://www.csedweek.org/., Dec 2010.