2009

Strengthening Learning Habits Through Individual Unique Assignments

Aharon Yadin
Management Information Systems Department, aharony@yvc.ac.il

STRENGTHENING LEARNING HABITS THROUGH INDIVIDUAL UNIQUE ASSIGNMENTS

Aharon Yadin
Management Information Systems Department
The Max Stern Academic College of Emek Yezreel
aharony@yvc.ac.il

Abstract:

This paper describes a method for reinforcing learning habits among students in the Computer Science and Information Systems department. On top of the sharp reduction in the number of enrolled students, there is an increase in the dropout rate that stems from poor learning habits. Strengthening learning habits can play a significant role in increasing students' overall motivation and self-accountability, and it can reduce the relatively high failure rate. These learning characteristics are necessary as well for future collaborative activities, which are a cornerstone of the computer science profession. The individual unique assignments method described in this paper was employed on two courses and helped reduce the failing rate. This paper relates to some quantitative results combined with a qualitative analysis obtained through the students' own reflections on the method, and concludes with a discussion on the results and future follow-up directions.

Keywords: Learning habits, individual unique assignments, decrease dropout

I. INTRODUCTION

In spite of the recent economic turmoil and market fluctuations, there is a strong demand for Information Technology (IT) and Information Systems (IS) professionals [www.executivebrief.com, 2008; itpro.co.uk, 2008; Thompson, 2008]. However, this demand fails to reach potential students, as demonstrated by the steady decline in Information Systems enrollment [Granger, Dick et al. 2007]. Many studies have offered explanations for the phenomenon, including the crash of the dot-com industry and its artificial hype, the false perception that the job market is being outsourced, and the understanding that it is a highly demanding profession with poor teaching practices [Clear, Edwards et al. 2007]. On top of that, there is the underrepresentation of women in CS (Computer Science) and IS, as stated by Yasuhara (2005):

> "The "shrinking pipeline" is a common metaphor for the underrepresentation of women in Computer Science (CS), an increasingly well-known (if not well-understood) phenomenon."

In addition to the low enrollment, there is a relatively high dropout rate to be considered, as described by McGettrick, Boyle et al. (2004):

> "People view the curriculum as being too complex and too crowded. Evidence for this is the dropout rate in many institutions, which often is as high as 30-50 percent."

With regard to this dropout rate, D'Souza, Hamilton et al. (2008) claimed:

> "Programming is central to Computer Science and cognate disciplines, and poses early-learning challenges in problem-solving and coding."

However, although programming is a central pillar to CS and some IS programs, dropping out is triggered not only by programming but by other introductory courses as well. In our case, the failing rate among first year students is about 25% and only half of them are directly linked to programming courses.

Due to the importance of these introductory courses on the students' retention decisions; we offer a variety of additional teaching assisting aids and mentoring services. However, in spite of these aids, in many of the courses we deliver, the lack of good learning habits is apparent and demonstrated by the grades bimodal distribution (a two-peak distribution). This distribution is characterized by a relative low class average and a high dispersion (a high Standard Deviation value). As stated by Chang and Chang (2000), this bimodal distribution is mainly due to the students' learning patterns. In our case, the bimodal distribution is explained by the heterogeneous students' population. Some of our students don't posses any prior computer and programming knowledge, while others are already working in the industry. Unfortunately, during their studies and due to their learning habits, the students did not narrow the existing knowledge gap between these two groups. In addition to the bimodal grade distribution, during the last several years a steady increase in the courses failing rate was observed. This elevated failing rate triggered several changes and the usage of a new pedagogical method and some modifications to the courses' structure.

In the current global world, where distributed teams and collaboration are a key factor for success, many researchers stress the importance of collaborative learning, in which the group members actively support the learning processes of one another [Gol and Nafalski, 2007]. However, collaborative learning that encourages students to work together depends on individual capabilities, motivation, and responsibility [Yadin and Or-Bach, 2008]. Our students' grade distribution which reflects their prior knowledge level highlights some of the problems associated with collaborative learning. In many advanced courses we use collaborative learning practices and through students' reflections identified some of the associated problems, mainly "free-riders" (students who are part of the collaborative team, but fail to do their share). This research is aimed at the individual learning level and it uses a learning method for strengthening individual capabilities (or learning habits). This approach is not intended to replace the collaborative learning practices; on the contrary, it is used as a preliminary stage that is mandatory for current courses' achievements as well as for any future successful collaboration activities. Strengthening learning habits increases students' capabilities and enhances their motivation and sense of responsibility, the characteristics required for becoming a better team-player in future collaborative activities.

II. THE PARTICIPATING COURSES

This research took place as part of two courses: (1) Systems Architecture and (2) Computer Organization and Programming. This section provides a brief description of these two courses.

The Computer Organization and Programming Course

The Computer Organization and Programming (COAP) is a mandatory, introductory course providing basic understanding of the computer system operations, data representation, system's architecture and the assembly language. This course is a pre-requisite for the SA course. The participant students are in their second year after successfully completing all first year requirements. The main course objective is to enhance software developers' understanding of hardware functions and operations. This understanding is demonstrated by programming using assembly language. This course is delivered during the first semester.

The Systems Architecture Course

The Systems Architecture (SA) is an elective second year course open to students who are willing to be involved with Information Technology beyond Information Systems studies. The course objectives are to provide a basic understanding of computer hardware operations with special emphasis on technological developments, future computing trends and their organizational applicability. The course focuses on the system components and their suitability to the required overall system functionality. Special attention is given to the hardware support for efficient OS (operating system) performance. This course is delivered during the second semester.

III. THE COURSES' VERSIONS

Being aware of the students' learning habits, both courses are based on reinforcing continual learning throughout the whole semester. The consistent increase in the students' failing rate over the last several years was addressed by introducing modifications to the two courses. The several versions of these courses that were taught during the past seven years, and that were accompanied by action researches and studies, highlighted the need for structural changes, such as the inclusion of a mid-term exam and various revisions of both manual and computerized assignments. However, in spite of the introduced changes, the failure rate in both courses continued to increase. The different versions were applied in parallel on both courses.

The first version included several assignments. Some of them were individual assignments, while others were collaborative assignments (2 students per team). However, for some of the students there was a large difference between the average assignments' grade and the final exam grade. For students with relatively lower grades, the par between these grades was even greater. The possible explanation is that although some assignments were performed collaboratively, the learning was not equally shared (pointing out the possibility that one student on the team was "free rider"). For the individual assignments, a possible explanation is that the students compared their solutions with other students' work and corrected their mistakes without the required understanding. Some of the students did not prepare themselves sufficiently for the homework and no learning took place. In this course version, students studied and prepared themselves only for the final exam.

In order to encourage comprehensive learning throughout the semester and not only at the end for the exam, the second course version employed a large variety of exercises for the students' self-study. Unfortunately, it did not produce the anticipated results. Apparently students did not address these exercises seriously enough, and so they did not produce the intended positive results. This version demonstrated that the students' motivation and responsibility are not high enough and relying on them is not sufficient.

The third courses version was based on a combination of the self-study exercises and a mid-term exam with little or no assignments. In this case the action research revealed some inconsistent behavior. If a variation of some question answered properly during the mid-term exam was included in the final exam, the answers (on average) on the final were worse. Students who, on the mid-term, managed to answer the question correctly, failed to do so on the final. This implies poor learning habits, or learning just for the mid-term exam. This version strengthens the observations gained during implementation of the second course version. Working individually or collaboratively, the level of motivations and responsibility demonstrated by the students was not sufficient and the course structure had to be modified again.

The last version of the course consists of self-study exercises, a mid-term exam, a final exam, and in addition several (at least six) individual and unique assignments. Unique in this sense means that each student receives a different assignment, therefore it is impossible for students to compare their solutions. Collaboration is possible, and is even encouraged. But with these kinds of assignments, collaboration can only mean working and learning together, and not simply copying from one another.

The following is an example of an individual and unique assignment given in the SA course. The assignment consists of two parts and its purpose is to encourage students' learning and assess their understanding of (1) dynamic pipeline scheduling for out-of-order instruction execution using the "Scoreboarding algorithm," and (2) the "Branch prediction" concepts utilized by the processor's control unit.

a. Scoreboarding algorithm

    1. On top of the assignment write your home phone number (without the area code). If you do not have a phone at home, contact the instructor for obtaining a different type of input for your assignment.

2. Copy the first three digits (on the left hand side of the phone number) to its right hand side. The seven digit phone number becomes a ten digit number. For example, if the phone number is: 5324186, it becomes: 5324186532
3. A specific CPU consists of ten registers. Construct the Scoreboard control circuit for a code segment that contains nine multiplication instructions between these registers. The first instruction multiplies the register specified by the first digit in the number obtained in the previous step (R5 in the example given in step number 2), by the register specified by the second digit (R3). The second instruction multiplies the register specified by the second digit (R3), by the register specified by the third digit (R2) and so forth up until the ninth instruction which multiplies the register specified by the ninth digit (R3), by the register specified by the tenth digit (R2). The result of each multiply instruction is stored in the appropriate register. The first result is stored in the register specified by the first digit (R5) the second result is stored in the register specified by the second digit (R3) and so on. For example, if the number in step 2 above is 2510374251, then the nine instructions are:

R2 = R2 * R5

R5 = R5 * R1

R1 = R1 * R0

.

.

R2 = R2 * R5

R5 = R5 * R1

You may assume the all registers are available when this code segment starts execution.

b. Branch prediction

4. On top of the assignment write your cellular phone number (including the area code). If you do not have a cellular phone, contact the instructor for obtaining a different type of input for your assignment.
5. Split the number into single digits starting on the left hand side.
6. Every digit represents a branch condition. If it is an odd digit the branch is taken, if not (if it is an even digit) the consecutive instruction is executed.
7. Build the appropriate prediction table while utilizing a single bit branch prediction mechanism. Assume the CPU always performs the next consecutive instruction and calculate the success rate for this sample.
8. How does your answer change if the CPU utilizes a two bits branch prediction mechanism? You have to build the new prediction table and calculate the new success rate.
9. Compare the behavior of the two methods as observed by this specific sample and explain the findings.

These assignments are impossible to replicate or even compare with fellow students. If students seek help from other students, they can receive only an explanation on the topic. They will still have to try and understand the work for themselves and try and cope, best as they can, with their specific assignment based on their own comprehension. Students cannot ask a fellow student to check their solution, because checking a different assignment would require a significant amount of time (equivalent to the time needed to solve it).

IV. THE STUDY

The SA course was delivered six times and COAP was delivered seven times during the last seven years. However, the unique assignments were implemented in the 2008 academic year. During this year there were only fourteen students enrolled in the SA course and eighteen students in the COAP course. In each of the courses there were seven assignments during the semester worth 10% of the final grade. The mid-term exam contributed an additional 20% to the final grade and the final exam was worth 70%. All the assignments in the courses were unique assignments as described in the previous section. This means that each assignment had several versions and each student received a different one. Each submitted assignment was evaluated and graded and was sent back to the student including detailed corrective feedback with extra explanations (when needed) and links to the learning materials and to additional exercises. The feedback also enhanced the course materials for the following year. These included additional explanations inserted into the course textbook, a number of in-class slides demonstrating the principles at hand, and at-home exercised intended to further strengthen the students' comprehension. Our electronic submission system was used to publish the assignments, set the last date for submission, collect the assignments, and return the relevant feedback for each submitted work.

The main measure used for assessing the learning effectiveness of this individual unique assignments method was the overall course failing rate (in each of the participating courses). In addition, the students provided some reflections on the course and the learning method employed.

Figure 1 illustrates the individual unique assignments' effect and it relates to the COAP course. The two lines represent the two exams student are allowed to take for each course. If the student feels the grade of the first exam does not truly represent his/her knowledge (or in case of failure), he/she may take the second exam. The final grade is the one obtained on the second attempt. The number underneath the year indicates the number of students in the course for that specific year.

Figure 2 illustrates the individual unique assignments' effect and it relates to the SA course. During the 2006 academic year the course was not offered and the values used are the average of the year preceding and following it.

Employing the individual unique assignments significantly reduced the courses failing rate, and in 2008, for the first time ever no one failed the SA course. This achievement was repeated during 2009 as well. The improvement observed for the COAP course is significant however the unique assignments method was not sufficient to assure that no one will fail the course. Among the six students who failed the first term COAP exam during 2009, four submitted only one assignment. An additional interesting observation relates to the difference between the average assignments' grades (for all students) and the average final exam grades. Since the implementation of the unique assignments method, the gap is maintained at less than 10 point. This provides an additional assurance that the submitted assignments really represented the students' individual work and knowledge.

The individual unique assignments have increased students’ motivation, involvement, and accountability as was demonstrated by the number of times they accessed the course web site. While in 2007 on average each student accessed the system 22.9 times per semester (or an average of 1.7 times a week), during 2008 on average each student accessed the course web site 72.6 times (or an average of 5.6 times a week). This is an additional demonstrator for the students' active involvement.

The students’ reflections shed light on some additional encouraging observations: (1) the students expressed a higher degree of motivation and self-confidence in their ability to cope with other, new, and difficult topics related to system architectures; (2) the students spent more time on their assignments, both before submitting their work and after receiving feedback; (3) the students expressed a higher degree of appreciation for the feedback they received; (4) there was common agreement that the unique assignments increased their understanding, although it required a significant investment of time.

An important personal observation relates to the fact that there was a higher degree of students' involvement in class and in responding to fellow students' questions posted on the course forum. In addition, more students provided support to their peers when they explained some of the topics intrinsic to coping with the requirements of the assignments.

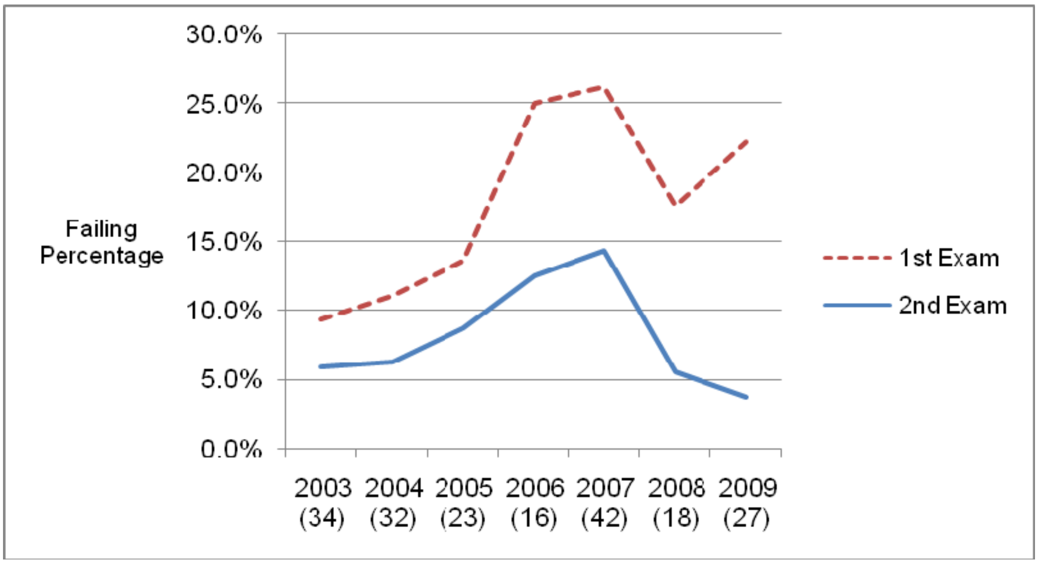

Figure 1: COAP Course Failing Percentage

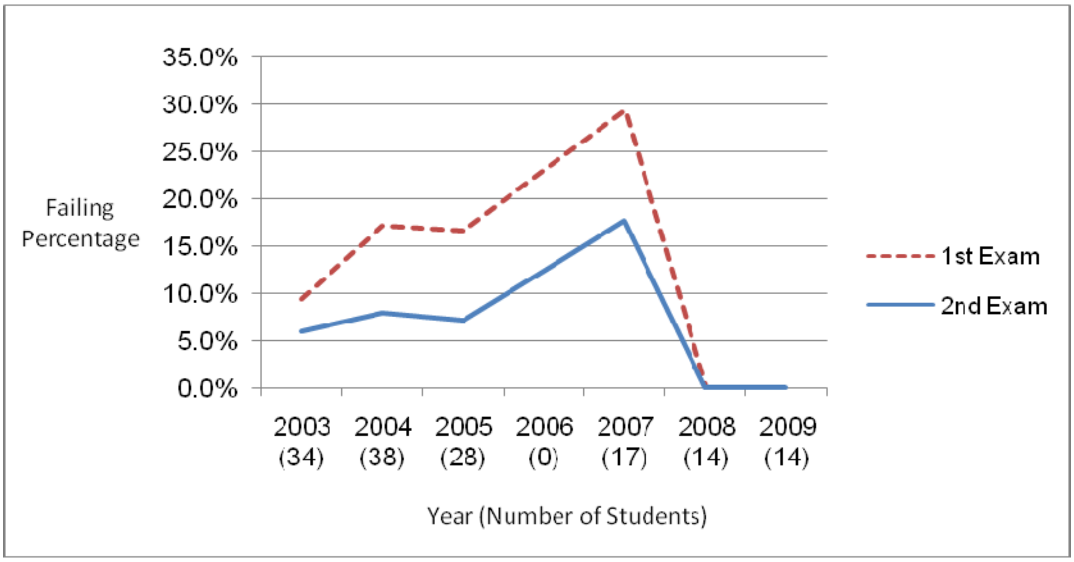

Figure 2: SA Course Failing Percentage

V. DISCUSSION AND FUTURE RESEARCH DIRECTION

The results achieved by using the individual unique assignments method were extremely positive and encouraging. The quantitative results demonstrated a significant decrease in the number of students failing the courses, which represents a counter direction to the trend seen the years before, and it holds additional promising potential. Even the increase in the failing rate exhibited in 2009 COAP first exam supports these results, since 67% of the failing students, submitted only one assignment, which was not sufficient for changing their learning habits and pass the exam. These results strengthen the impression that even in their second year students still need to improve their learning habits. They continue not to spend the requisite time to properly address their coursework. With unique assignments however, students cannot rely on others to complete parts of their work; they cannot patch together 'borrowed work' with their own. With individual unique assignments there is no alternative but for them to invest their own time entirely to complete their homework. As a result of spending this additional time on their assignments, students become more involved and learn to appreciate the feedback they receive, even though the responses might be very similar to that provided in previous years. But in 2008 and 2009 the students appreciated it more, claiming faculty responses were more relevant and helped them to better understand the material. This can be explained by the extra efforts the student made when completing their assignments. This increased their accountability and their sense of proprietorship over the solutions submitted. This sense of ownership was not apparent when the assignment was only partially thought through, cobbled together from various pieces obtained from classmates. This conclusion coincides with other findings related to individual assessment and a higher degree of students' involvement, and likewise affects feedback appreciation [Yadin and Or-Bach, 2008].

Reducing the overall failure rate demonstrates one success factor attributed to the method. However there is an additional benefit to the individual unique assignments as shown in Figures 3 and 4. These figures depict the grade dispersion (one SD from the mean) for the two courses. For the SA course, the bimodal grade distribution, which was extremely evident in the academic year 2005, does not exist in 2008 and 2009. It is demonstrated (in the 2008 academic year for example) by the combination of a higher average class grade and lower grade dispersion. ($\sigma_{2005} = 24.0$, $\sigma_{2008} = 9.4$)

An additional benefit, which was not presented in the students' reflections, but was observed in class, was a higher degree of involvement and participation both in class and on the course website posts. Some students, usually the best performers, assumed a new role of explaining the assignments to others. They did not merely show their peers what to do, but explained to them how to approach the work and solve it. From the instructor's point of view, the method provided clear and objective understanding regarding each student's achievements and the class's as a whole. Topics that were not fully understood were easy to spot and since this usually happened in the early stages of a taught subject, providing additional materials and explanations corrected the problem.

It should be noted that preparing unique assignments requires some degree of innovation and a significant amount of time on the part of the instructor. The results obtained, however, are rewarding and compensate for this extra investment.

There are two future directions planned for the individual unique assignments method. It will be implemented in additional courses so a larger comparison will be possible. In addition, the contribution of this method for enhancing the students' collaborative skills will be assessed in follow-up courses.

There are several studies within the collaborative learning research that highlight the conflicts between individual solutions as triggers for effective collaborative learning [Constantino-Gonzalez, Suthers, and Escamilla, 2003; Or-Bach and Joolingen, 2004]. It might be interesting to see if this method helps in these conflicting solutions and indirectly supports more effective collaboration.

Understanding the students' problems in comprehending course topics, provides a mechanism for, on the one hand, enhancing the materials, and on the other, fine tuning the unique assignments to cover these hitherto unknown difficulties.

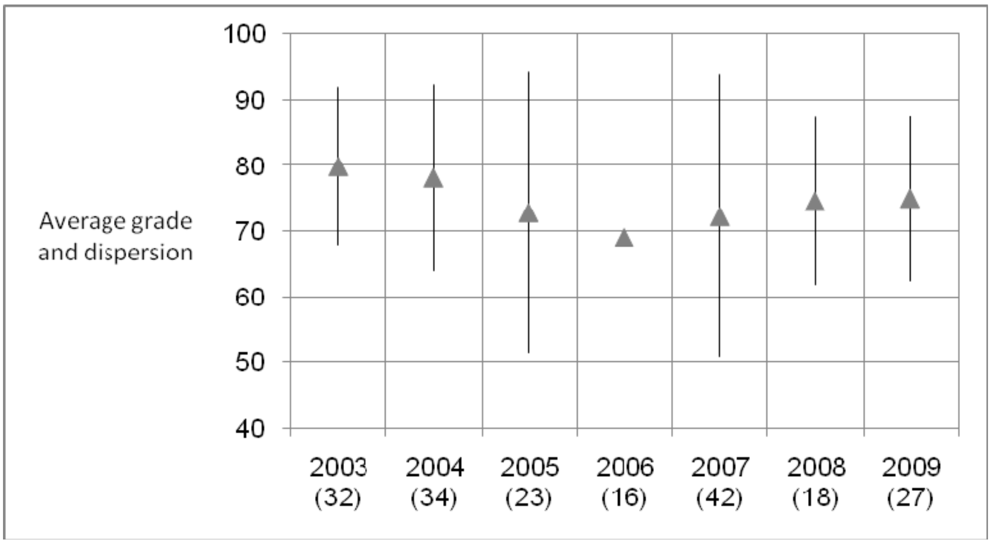

Figure 3: COAP Grade Dispersion

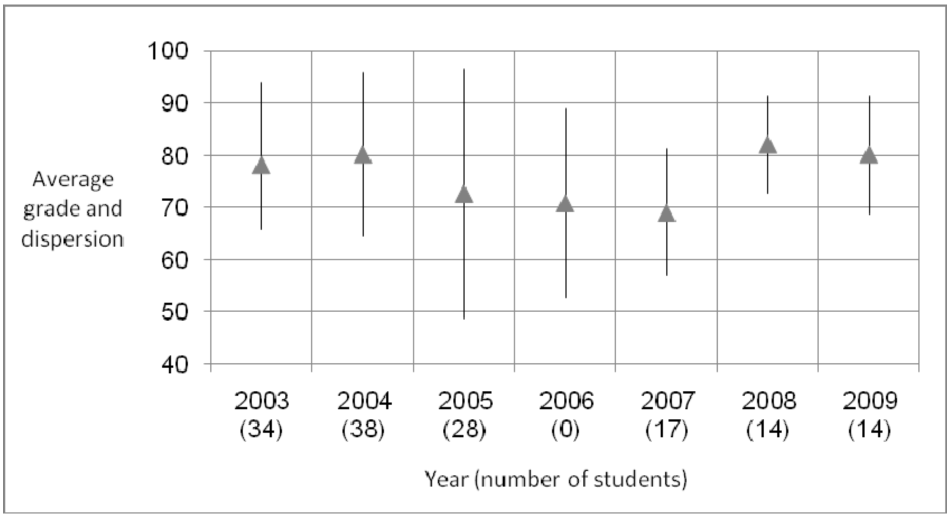

Figure 4: SA Grade Dispersion

REFERENCES

Chang, T., and Chang, D. (2000). The Role of Myers-Briggs Type Indicator in Electrical Engineering Education. URL: http://www.ineer.org/Events/ICEE2000/Proceedings/papers/MD6-2.pdf (accessed January 2009).

Clear, T., Edwards, J., Lister, R., Simon, B., Thompson, E. and Whalley, J. (2007) The Teaching of Novice Computer Programmers: Bringing the Scholarly-Research Approach to Australia. Tenth Australasian Computing Education Conference (ACE 2008), Wollongong, NSW, Australia. CRPIT, 78. Simon and Hamilton, M., Eds., ACS. 63-68.

Constantino-Gonzalez, M., Suthers, D. and Escamilla, J. (2003) Coaching web-based collaborative learning based on problem solution differences and participation. International Journal of Artificial Intelligence in Education, 13(2-4), 263-299.

D'Souza, D., Hamilton, M., Harland, J., Muir, P., Thevathayan, C. and Walker, C. (2008) Transforming learning of programming: a mentoring project. In Proc. Tenth Australasian Computing Education Conference (ACE 2008), Wollongong, NSW, Australia. CRPIT, 78. Simon and Hamilton, M., Eds., ACS. 75-84.

Gol, O., and Nafalski, A. (2007) Collaborative Learning in Engineering Education, 10th UICEE Annual Conference on Engineering Education. URL: http://eng.monash.edu.au/uicee/gjee/vol11no2/GolNafalskis.pdf (accessed January 2009).

Granger, M.J., Dick, G., Jacobson, C. and Van Slyke, C. (2007) Information Systems Enrollments: Challenges and Strategies, Journal of Information Systems Education.

McGettrick, A, Boyle, R, Ibbett, R, Lloyd, J, Lovegrove, L, and Mander, K. (2005) Grand Challenges in Computing: Education – A Summary. The Computer Journal. The British Computer Society. Vol. 48, No. 1.(pp 42-48).

Or-Bach, R. and Joolingen W.R. van. (2004) Designing adaptive interventions for online collaborative modeling. Education and Information Technologies. Vol. 9, No. 4, 355-375.

Thompson, C. (2008) E-Government; ICT Professionalism and Competences; Service Science in IFIP International Federation for Information Processing, Volume 280; Antonino Mazzeo, Roberto Bellini, Gianmario Motta; (Eds.) (Boston: Springer), pp. 69–80.

www.executivebrief.com (2008) "Continued demand for IT professionals in 1Q 2009 seen, survey says". http://www.executivebrief.com/news/continued-demand-for-it-professionals-in-1q-2009-seen-survey-says/ (accessed April 2009).

www.itpro.co.uk (2008) "Demand for IT pros hits high". http://www.itpro.co.uk/153138/demand-for-it-pros-hits-high (accessed April 2009)

Yadin, A. and Or-Bach, R (2008) Fostering Individual Learning – When and How. ACM's Inroads – SIGCSE Bulletin, 40(4), Association for Computing Machinery, New York, 83-86.

Yasuhara, K. (2005) Choosing Computer Science: Women at the start of the Undergraduate Pipeline. Proceedings of the 2005 American Society for Engineering Education Annual Conference & Exposition. URL: http://staff.washington.edu/yasuhara//cv/publications/Yas05.ASEE-CS_1_gender_prelim.pdf (accessed January 2009).