Teaching Computer Programming with a Coaching Mindset

Abstract—Teaching computer programming with the coaching mindset assumes an inherent knowledge on part of the learner. Conversely learning is efficient when novices learn from people who already mastered the craft. In this paper we redefine computing teacher as a Coach, an extension to the cognitive teaching model based on a set of values and practices that emphasize a radical model of student-teacher relationship. The proposed model resulted in a significant improvement in the confidence and skill levels of beginner students which reflected in their pass rate as well as arrested dropout tendencies. Further, we describe the coaching paradigm in the context of cognitive teaching model proposed by Maslow as the most efficient method of teaching programming.

Index terms — Teaching Computer programming, Coaching, Cognitive Theories; Computer programming

I. PROBLEM DEFINITION

Teaching programming is challenging which is the reason for all the research from many different perspectives for the last two decades [1] [2]. Yet, it still is. It is so primarily because the way we approach it, as we understand in this study. We always used traditional teaching format in the introductory computer science courses that consisted two parts: lectures and demo sessions where model solutions to the exercises are shown as classroom activity; and work-sheets with similar set of problems as take-home assignments. This approach is not different from that of other universities and tertiary education providers [3] [4]. Furthermore, we notice that our programming lectures were, a large part, structured according to the language constructs with the assumption that learning the syntax or semantics of individual language constructs would allow the students to master the process as to how to combine constructs to meaningful programs. Again this practice is identical to what other universities do for their graduate programming courses [5] [1] [6].

In addition, the same constructs of the selected programming language are typically recycled in the take-home exercises and doing these exercises, without any knowledge of the programming process, is hard for majority of the beginner student population [3] [4]. Many such students end up not pursuing computer programming career [2] due to not being able to solve problems and therefore feeling inadequate [1]. The studies in educational psychology [7] notify that, due to the nature of human cognitive architecture, a minimally guided approach is not optimal for novices learning a cognitively challenging task, such as programming. That is why students developed bad work habits from doing take-home exercises all by themselves.

Teaching programming this way does not seem to make the grade. Programming courses, as a result, suffer frustration by flat-lining test scores and under-performance on formative tests and teachers wonder: what are we doing in class that would inhibit that intellectual creativity and cognitive inventiveness? Why the drop-out rates of introductory programming courses are tend to be high? It is, therefore, quite evident that the traditional teaching approach is inadequate and we needed to shift the focus onto the process rather than just on the end products.

In this paper we want to revive the forgotten paradigm for teaching, coaching - a method of interacting with another person to help them identify critically important values, to explore new ways to think and behave, to achieve cherished personal and professional goals, and to feel more vibrant and present. This approach is actually a variation of Cognitive model of teaching that has a strong emphasis on guided programming exercises. We report the experiences from its application at the UUNZ Institute of Technology Department of Computer Science.

II. COACHING AS A VIABLE TEACHING MODEL

Coaching is defined as “an ongoing professional relationship that helps people produce extraordinary results in their lives and careers” by the International Coaching Federation (ICF) . Coaching, therefore, is done by encouraging creativity, engaging participants in self-generated solutions to problems, and by promoting responsible and accountable behaviors and actions [8]. Coaching as a credible means of boosting performance has grown exponentially over the last decade in a number of high-performance fields including such varied fields as business and medicine. According to the current research [9], Coaching as a viable teaching solution, is becoming increasingly sought out among heart surgeons [10], C-suite executives [11], and even in the office-rooms and teachers lounges [12]. However, the idea that coaching as a viable alternative approach to metamorphose the pedagogical relationship between student and teacher is yet to take deep root. This is because the fundamental precepts of professional coaching paradigm has not been fully realized [8] in educational circles. One of the reasons for this lapse is that teachers do not seem to assume competence, resourcefulness, and wholeness of every student let alone see this as the center of student-teacher relationship.

Educationists, therefore, need to realize the fundamental principles that professional coaching rests on. Professional coaching to work, teachers need to believe that a student,
or set of students that they are interacting with, are already skillful, has useful knowledge, and has a profound desire to learn and to achieve the goals they feel are important. In New Zealand te ao Mori, the concept of ako recognizes the knowledge that both teachers and learners bring to learning interactions, and the educational research acknowledges the way that new knowledge and understandings can improve students' achievement [8]. The stance of assumed competence and resourcefulness on the part of the coach, "the belief part" is critical, and a significant departure from the conventional teaching relationship, which presumes that students lack something that the teacher must give them. As Paulo Freire [13] described it, in the hierarchical method of education in which the teacher knows and the student is to know is still a pattern very widely abroad and instantiated in the lived patterns of teaching in our society. To be a professional coach means giving up the all-knowing stance of the paid consultant, and even the knowledge-bearing mantle of the teacher. You become the inquirer, the one who ask imploring questions, the one who foment curiosity. When you coach, you stop giving advice and stop thinking about what you know, and start getting really curious about what is going on with the other person (or people). Cheliotes and Reilly [12] note that, "Coaches operate with an underlying assumption that giving advice to others undermines the confidence and self-worth of others. Others don't need to be fixed."

In teaching we need to move to exactly this stance in order to foster creativity in our students to allow our students the choice, control, novelty and challenge that builds their creativity the essential conditions as defined by Csikszentmihalyi and others. Without the assumption that our students are already competent, imaginative, and ready to burst forth with regular exhibitions of novel and valuable ideas and products [14] we are limiting their creative capacities before they've even had a chance to discover them. As Lou Cozolinos [15] wondered new book on the social neuroscience of education makes clear, how we feel about our learning environments, and the assumptions that are made about us as learners within them, dramatically affect our brain development and our capacity to produce creative and novel work products. This means, in practice, moving from giving advice to students or giving them answers, to creating awareness of what they want to know and helping them design actions to achieve their learning goals. This also means not offering options for learning, but encouraging learners to design possibilities themselves, and then insisting students themselves plan and goal-set, monitor their progress, and then analyze what worked and what didn't. Thus teachers conceiving of themselves specifically as coaches rather than teachers, though radical, yet it is an incredibly promising idea. It also assumes that most of us find reflection, experimentation, and accountability helpful in achieving our goals.

III. Coaching as an Effective Model for Teaching Programming

The method of Coaching was applied in introductory programming courses at the Department of Computer Science at the UUNZ Institute of Technology in three separate academic semesters each of which lasted 10 to 11 weeks. Until Nov 2012 the introductory programming courses followed the traditional lecture and take-home exercise model. The first course implementation following Coaching model started in Semester 1, 2013. The whole programming course is taught as two separate units namely Introductory Programming and Advanced Programming where advanced programming further deepens the knowledge built during the introduction to programming course. The teaching components in the Introductory Programming course included assignments, expressions, terminal input and output, basic control structures, classes, objects, methods, arrays and strings whereas advanced programming focused on object oriented features such as inheritance, interfaces and polymorphism, and familiarizes students with the most essential features of Java API, exceptions and file I/O. Based on the four stages of learning as uncovered by Abraham Maslow [16], we organized the coaching model vary and run through the first three stages in order to achieve the fourth.

A. Unconscious Incompetence

I don't know that I don't know how to do this. This is the stage of blissful ignorance before learning begins. In this stage we provided students a conceptual model of the process, with which an expert performs the task under study. One effective way of teaching at this stage was to base the lectures on worked examples [17] instead of concentrating on language structures. A worked example shows e.g. completion of a programming task from start to finish. While completing the task, we were performing thinking aloud all the time, explaining the decisions made during the process.

B. Conscious Incompetence

I know that I don't know how to do this, yet. This is the most difficult stage, where learning begins, and where the most judgments against self are formed. This is also the stage that most people give up. After the Unconscious Incompetence stage, students move to the Conscious Incompetence stage. Typically this means that the students are exposed to exercises that are made under the guidance of an experienced instructor. We also used some students as peer teachers at this stage. The key idea is that students are not given straight answers, but rather just enough hints to be able to discover the answers to their questions themselves. The teaching done here is based on Vygotsky's idea that learning is most efficient when a student is given just enough information that is enough to boost the students ability to finish the task [18].

C. Conscious Competence

I know that I know how to do this. This stage of learning is much easier than the second stage, but it is still a bit uncomfortable and self-conscious. When the student starts to master a task by themselves, the scaffolding was carefully dismantled. This is the final stage of learning through coaching. The coaching-based approach to learning programming seems to be advocated also by the Agile and Software industry gurus such as Robert Martin who stated that Software is a craft that takes years to learn, and more years to master. The only way to properly learn the craft is to be taught at the side of a master [19]. Martin seems to call for
coaching-type mentoring to the software industry, where the recently graduates would work in a software project context with constant interactive guidance by masters in the field.

D. Unconscious Competence

What, you say I did something well? The final stage of learning a skill is when it has become a natural part of us; we don’t have to think about it.

IV. Course Results

Teaching programming with a coaching mindset was implemented in Semester 1 of 2013. In previous instances of the course, passing the course was based on an exam. With our new model, the course began on the assumption that to pass the course, students should do all the exercises given to them. However, because we were running the course for the first time, we had to learn what number of assignments it was reasonable to require of the students during the course. We finally relaxed the initial do all requirement and passed all the groups who had completed, on average, at least 80 percent of the weekly exercises. Although we find no clear way to compare between the two rather different course instances, the new teaching model appeared to engage students better. In the previous instances, the pass rates were 61 percent (2011) and 43 percent (2012). For our redesigned course, 48 students registered and 41 attended a starting lecture where the course was introduced to students. Altogether, 35 students passed the course, increasing the pass rate to 85 percent. During the first third of the course, the groups completed nearly all the exercises given, and later, even with the most difficult topics, the completion rate was well over 50 percent. For one group with three students, we had to give extra exercises at the end of the academic period to be able to accept their performance as pass. The rest of the student cohort passed the course, in my subjective view, with skills ranging from acceptable to extraordinary.

V. What a Coaching Stance Looks like in a Classroom

As we notice [20] [21] extraordinary teachers have always viewed themselves as coaches of students. In the past gifted education has provided a model for this approach in its playful, inventive invitations to learn, and assumptions of competence on the part of the learner. (What if we regarded all students as gifted?) Paula White, a longtime teacher in Albemare County, Virginia observes about her own stance as a highly successful teacher, “In my 38 years of teaching, I always begin with an assumption of competence. That means we believe learners are competent and come to us with strengths and knowledge and skills and talents and curiosities and yearnings and expertise and questions, and it is my job to discover what those skills and talents are and in doing so I always discover new capacities within myself.” [22] Like a professional coach, Paula begins with the belief that her learners have a lot to teach her, and she gets the privilege of playing with them and helping them achieve their learning goals throughout the year. One of the most creative and widely respected teachers in her region, Paula was one of the first Apple Educators to be recognized and leads a lively blog on transforming teaching. Long time teacher Chad Sansing also notes, like a coach, that the key to unleashing creativity in students is giving up his claim as knowledge authorizer. “I feel very self-conscious, selfish, and unsure writing this, but I wanted to share what happened today in my classroom because if I have anything to offer (besides the occasional oblique reference or terrible pun), its an approach to teacher failure that remains open to student success. The best I can do is to be delegitimized as an authority-figure and known as a person and learner by my students. The work isn’t there to isolate resistant students, to assert my control, or to protect my feelings like a curtain wall; its there to be torn to pieces and remixed or discarded as we build our relationships and community together”[23].

Paula and Chad’s coach-like stances: assuming competence on the part of every learner, believing their roles are to create positively-charged and accountable space for learner growth, and giving up their authority as knower, all point to the power of the professional coach in the classroom, and its social justice implications. Discovering and clarifying what the student wants to achieve, encouraging self-discovery, getting the student generate solutions and strategies for solving problems, and holding them responsible for results, upends and re-balances the traditional student teacher relationship. Often it may well worth to pose the question to each of us as teachers of programming, What if the students looking up at you from the classroom didn’t need anything from you? What if you assumed they weren’t lacking anything? How would that change about your job as teacher?

VI. Unanswered Questions and Future Work

In this paper, we presented a teaching deviation model and discussed it through cognitive learning theories, within the framework of Maslow [16]. However, we do not assume a straightforward success without addressing several challenges in our first pilot in 2013. These include:

- How can we develop our dialog towards promoting a learning oriented study culture, especially in the review sessions?
- How do students react to no grades? Does the lack of grades actually result in learning orientation or does it cause objection due to the impression of a missing reward? It is well known that good grades are good currency in academic world [24].
• How can we motivate students to take control of their own learning? We would not like to impose self-direction on unwilling students.
• How do we manage the degree of difficulty in programming assignments? How can we implement the appropriate level of support (scaffolding) in the progression of the weekly assignments?
• How can we construct pedagogically effective summaries from student questions in a manner that directs students towards more abstract topics in functional programming?

From a research perspective, we will be able to monitor the dropout rate and compare it to previous course instances. We can also easily monitor whether the students use the opportunity to formulate questions to help their own learning. If we receive little questions, we will more thoroughly examine reasons for this at the time of course completion. In particular, we will collect the students’ experiences of learning programming in the peer learning context. This will take place at least once during the course and once near the course completion. We will use a fixed set of questions:

• How would you describe your role in the group?
• Does this role challenge you as a learner?
• How would you change your role?

These questions are based on well-known ideas that some kind of norms and social organization will occur in a newly formed group, affecting the roles of the group members; see for example [25]. Equally important for us is to collect feedback of how students perceive the classroom study culture during the contact sessions, which we will also collect during the course and at the time of course completion. This will be collected anonymously. The data will be collected using questionnaires with open ended questions, with the aim of being able to elaborate the course model towards another action research cycle. Our own teacher reflections will provide an equally important research tool.

VII. CONCLUSION

Coaching paradigm presented in this paper provides a good structure for teaching skills that require building routine and learning best practices from the masters. Emphasizing scaffolding in combination with the set of values and practices yields very promising results as seen in the initial implementations with 67 and 44 students, the most important result being the significant decrease in dropout rates. We believe that the idea of taking continuous feedback and scaffolding to a high level provides enough support to also help some of the inefficient novices, who usually drop programming courses, to learn programming. The role of relevant exercises for making learning by doing a reality is a key factor in this approach. The amount of work that a student puts into exercises can have a negative impact on motivation if the exercises do not support his learning process in a meaningful way. The majority of the anonymous student feedback indicated that learning by doing was considered motivating and rewarding. A quote from an anonymous feedback summarizes the positive outcome of this approach:

• The best thing on the course was the availability of teachers
• the amount of exercises and exercise groups
• It was very rewarding to be simply working diligently
• Making mistakes also helped to learn things

Another respondent remarks: “We certainly need to alert students to the fact that socialising with friends is not learning and in fact eats into their time for learning.”

REFERENCES


