Active Learning in Computer Science @ UT Dallas

Dr. Jey Veerasamy, Professor
Mr. John Cole & Dr. Miguel Razo, Sr. Lecturers
Computer Science department
The University of Texas at Dallas, USA.
Email: [jeyv|john.cole|mrazo]@utdallas.edu

Most technical jobs in the old days (decades ago) look like assembly line - well defined inputs & goals, strict processes, well-defined hierarchy in the companies & complete expertise to handle the assigned jobs. Now, it looks more like a hi-tech playground - get in there without knowing everything, learn about the new blocks as needed, ask for help and collaborate with others, try new novel approaches hoping to succeed or fail fast, refuse to leave work due to exciting work, etc.

How do colleges adapt and make the students ready for the modern workplace? The University of Texas at Dallas has incorporated several mechanisms in the past decade to prepare the students. We will cover these items below.

ECS 1200

It is well understood that the early school years should focus on building the fundamentals and teaching principles. Another key focus for computer science is problem-solving ability. As a result, many universities have created a "First Year Engineering Experience" course. The purpose is not to teach any specific skill, but to foster an attitude of experimentation, teach soft skills, and promote teamwork. At UT Dallas this resulted in the ECS1200 course.

The course involves several activities as well as traditional lecture. One of the first activities is a "scavenger hunt," in which students are required to take photos of themselves at various locations around campus, including the computer lab, the advising office, their professor's office, and others. The purpose is to get them to familiarize themselves with the campus while working as a team.

Another issue is that we see many students who did well in high school but do not have the time management skills and study habits that will let them do well in college. One way ECS1200 addresses the first problem is with a homework assignment to keep track of how much time the student is spending on studying for each course during a week. Thus a student might mark that he spent 1.25 hours studying for CS1337 on Monday, none on Tuesday, 2.5 hours on Wednesday, and so on. Many students are surprised at how little time they actually spend on schoolwork. We also talk about the effectiveness of their studying. If they are only reading the material but not working problems, that is not enough. A related issue is procrastination, a very common problem. We address this by not accepting late work for certain assignments.

It is difficult to teach problem-solving skills, and one of the issues is that although they are all freshmen, there is a wide range of programming ability, from those who have never programmed to students who have placed out of a year of courses. To address this, we teach techniques such as flowcharting and pseudocode, which allow them to think about the problem without worrying about the details of a particular programming language.

Another soft skill the marketplace demands is communication ability, both written and oral. While other courses, such as Rhetoric, require writing, ECS1200 requires both writing and a presentation. The presentation is usually done as part of a project, in which students design a program, create high-level pseudocode for it, and present their work.

It is hard to measure the efficacy of ECS1200, given that there is no control group and the student body is not comparable from one year to the next. However, anecdotal evidence is that it helps retain students at UT Dallas and in Computer Science as a major.

CS Outreach

You might have heard – "Best way to learn is to teach." We use UTD students to teach the coding skills to the school students. We also believe that all public universities should do public good - Since UTD is a public university funded by state, teaching coding to school students is a great way to give back to the community. While more & more schools are introducing Computer Science into the curriculum slowly, serious gaps continue to remain – specifically schools in low income neighborhoods are likely not to offer any coding at all, due to various reasons. We have seen the high school students graduate without knowing the meaning of "computer programming!"

With the plenty of free resources available through the web (code.org, MIT Scratch, CMU Alice, BBC micro:bit, KhanAcademy JavaScript, GameMaker, MIT App Inventor & Finch/Scribbler/WeDo robots, just to name a few!), it is easy

to find an appropriate learning tool for school students at any grade level. In addition to getting paid for their teaching hours, UTD students, especially international graduate students, improve their communication skills and build confidence before they face the interviews. Additionally, "CS Outreach instructor position" makes their resumes to stand out among their peers'.

We have been running two major programs, starting from 2013:

- AfterSchool Coding Clubs during the school year – we work with the interested local schools individually to identify specific weekday and conduct a coding club in the school's computer lab for an hour after school is over, for 8 to 10 weeks. This is very popular & efficient option for most elementary schools since they finish early in the afternoon & most parents prefer to come one hour later to pick-up their children. Depending on the status of each school's neighborhood, we decide whether to charge a fee for each student or do it for free. You can access more details @ utdallas.edu/k12/access

- Summer Coding Camps – most school students get 11 weeks of summer break. Typically, families may travel for a few summer weeks and stay at home for the remaining weeks since the parents have to work. US has a long tradition of sending school-age children to attend summer camps to explore various areas to get an idea about which field they like & to select the college major. We offer 10-15 camps every week of summer covering all grade levels. With 200-300 school students every week, our building looks like a Summer school!

- Goal of each camp is to introduce coding skills in an enjoyable hands-on manner at the age-appropriate level. Students can climb on their own pace and move up the levels. We always have a few students every summer who spend almost 10 weeks with us. We also have residential facility – out-of-town high school students can come and stay like UTD student and attend the camps. When a student completes all our camps, (s)he will have the coding skills of $2^{nd}/3^{rd}$ year BS CS student! It will enable such students to excel when they go to college. More details and the summer camp schedule can be found @ utdallas.edu/k12/summer

These programs employ 50-100 UTD students at a time. With local corporate sponsorships, we provide low-fee & no-fee programs to low income families. All our course materials are available online @ github.com/UTDallasCSO for other colleges to download and use. We can also provide our instructors to kick-start your camps and bring your staff upto speed.

Hackathons

The great thing about hackathons is that they provide a time-limited cycle to design, write, and test a program. This is far more realistic than writing a program for a class assignment, where you have a week or more and access to information from search engines and the textbook. Hackathons are also highly creative, since most people don't go into them with a project in mind, and the rules discourage (or outright forbid) coming with a pre-written program. The top teams win prizes ranging from electronic gadgets worth under $50 to hundreds of dollars.

Another benefit of hackathons is that they let corporate sponsors meet and talk with top students. That gives them a far better idea of who they might want to hire than just reading a resume.

UTD holds several hackathons every year. The big one is HackUTD, which in recent years had had about 500 students competing and has drawn hackers from as far away as Canada and England.

Other hackathons are more specialized. We held our first Hacks for Humanity, working with Arizona State University, in September. HackAI, organized around artificial intelligence programming, was held this Fall. Occasionally, companies will sponsor a hackathon around using one of their products, although they sometimes sponsor a category within HackUTD.

The UTD Computer Science Department actively supports hackathons, with faculty sponsors working with the student groups, assistance getting corporate sponsors, and providing the building and some of the infrastructure.

UTDesign

The capstone course is a requirement for any engineering and computer science department, seeking ABET accreditation. Initially, a senior student would work with a professor (an expert in the field of interest) to define a project, set the goals and the milestones for a semester. The student would then meet the professor throughout the semester, aiming to achieve the original milestones. At the end of the semester, the results are evaluated and stored.

Lately, colleges define a capstone course for all the senior students to register and the whole group meets early in the semester with the professor to take care of common items like project management. The project and its goals are still defined by the student and the professor continues to meet with each student individually to ensure progress.

The School of Engineering and Computer Science at UTD, took steps to enhance the capstone experience further, adding the following:

i. All projects are developed by student teams (no individual projects are allowed)

ii. There is an advisor for each team. The advisor is part of the faculty, research or staff that serves as a silent project manager.

iii. Projects are defined and managed by companies (local or national), or projects can also be faculty sponsored or student defined. Faculty sponsored projects are either a well-defined and challenging component of an ongoing research project, or an initial work for a new one, Student defined projects are an incentive for entrepreneurs; creative students who are looking to challenge themselves.

First requirement is to cultivate teamwork and leadership, offering students the opportunity to interact with other students & help each other to progress towards the common goals. Regular meetings outside the class time, are required and everyone is required to report their progress on a weekly basis.

The faculty advisor is a project monitor following the team's progress and a resource to set realistic expectation from both, the students and the sponsor, as well as the first point of contact to resolve conflicts among the team members.

Each company project is called as UTDesign project - it comes with a project mentor from the company. This role is fulfilled by an expert in the field, with deep technical knowledge, needed to advise and train (if needed) the team of students in order to achieve the project goals. Company mentor takes very active role in the project running the weekly meetings & is crucial person that ensures the success of the project. This engagement also ensures that there are no last-minute negative surprises! Each project typically falls into these 3 broad areas: proof of concept for a new product, enhancements of an existing product or exploration of new technology and frameworks.

Projects can also be defined and sponsored by a faculty member. These projects are appealing for students planning to continue with graduate studies or focus bit more on research. The faculty serves as the client, and the role of mentor is fulfilled by the faculty or a member of the faculty's research group.

Student sponsored projects, on the other hand, are ideal for those students who are ready to jump into the industry but would like to explore some interesting ideas on their own, potentially leading to a Start-up. All ideas are evaluated by experts in the field and also by UTD Entrepreneurship Institute. We provide suggestions/improvements that could increase the chances of success.

Industry sponsored projects offer many benefits to both students and sponsoring companies. From the student perspective, these projects are different to academic projects that may focus on a single technology aspect or framework. It may be the first industry-size project that many students work on – it is complex enough that it cannot be completed by one person either. Regular status meetings with the industry mentor brings time-discipline to the students. Typically these meetings are held at company premises and the students get to work with other employees as needed. It can become an immersive experience to learn about the company culture and the work atmosphere there.

For companies, UTDesign project is a great way to get a back-burner project done since regular employees may be busy with high priority tasks. Several companies also use "project mentor" role to groom their employees' project management skills. Finally, companies also use it as long-term assessment of students' skills & behavior and make full-time job offers at the end.

Since the inception of UTDesign projects in 2009, the computer science alone, has completed more than 350 projects, with near-perfect success rate. Such phenomenal success is attributed to upfront right-sizing the projects & active involvement from the industry mentor throughout the project. We do charge a project fee to the companies to ensure that the projects are of some value inside the company & get "enough" visibility within the company to take it seriously and assign a capable mentor.

Has any UTDesign project failed, i.e. did not meet the expectations? Yes, few projects have failed. Primary reasons range from company mentor is not committed, to unexpected roadblocks with new technology. In such cases, we take them as real-world learning experiences and provide a plan to the students to bring the project to meaningful close. What if the students are not committed, or the team is not working well together? In addition to weekly status meetings, we collect mid-term peer-review for students and mid-term assessment from the industry mentor and the faculty mentor. After analyzing the performance of each team, we assign a mid-term grade – we have had a few teams that got wake-up calls with low mid-term grade and managed to complete the projects successfully!

Visit utdallas.edu/utdesign for more details about this program.

About the authors

Dr. Jey Veerasamy works as Director of Center for Computer Science Education & Outreach and also as Teaching Faculty in the Department of Computer Science at the University of Texas at Dallas, USA. Prior to joining UT Dallas in August 2010, he worked in US wireless telecom software industry (Nortel & Samsung) for 16 years & taught online courses in several colleges for 11 years in parallel. Center for CS Education & Outreach offers 100s of coding camps & clubs to school students in Dallas area - it is one of the nation's largest CS Outreach programs run by any university. 1000s of school students benefit from these programs every year. Center also conducts technical workshops for UT Dallas students and professionals.

Mr. John Cole is a senior lecturer at the University of Texas at Dallas. Earlier, he had taught at Collin County Community College for three years, and prior to that, at Illinois Institute of Technology in the mid-1970s, which is also where he had earned his degrees. Before joining the full-time faculty at UTD in Fall 2012, he had taught part-time for 13 semesters. He has been a software developer for many years, working on projects as diverse as a SNOBOL4 compiler, a DATABUS compiler, a word processor, the operating system for an early computer, statistical analysis of insurance claims, telecommunications, and embedded programming. He has been using a variety of languages, including Java, C++, C#, Visual Basic, Databus, Intel assembly language, and many others.

Dr. Miguel RAZO obtained his M.S. in Telecommunications Engineering in 2006 and a Ph.D. in Computer Science in 2009 from The University of Texas at Dallas. He worked as Research Assistant from 2007 to 2009, and is currently a Research Associate, at the OpNeAR (Open Networking Advance Research) Laboratory and Senior Lecturer within the Computer Science Department at the Erik Jonsson School of Engineering and Computer Science. He has collaborated in design and implementation of software prototypes for telecom industry. His research interests include network planning, fault protection, telecommunication software design, protocol design and network modeling, emulation and simulation.