Curriculum Vitae: George Eleftherakis

Teaching

Teaching Philosophy - Statement

Learning, and then teaching all the new things I learn, is my passion. I realized that from the very early days of my life when I was in high school. Professionally, teaching was my main activity all my life. After I studied and taught Physics, I have chosen Computer Science as a field to continue, mainly to satisfy my lust for learning and the desire to explore new things. CS is a science that changes rapidly and continuously which makes the perfect environment for someone like me that wants all the time to learn and create new things.
The last two decades I am teaching in higher education and more specifically in computer science field, with focus in computer programming, software engineering and information security.

I carefully strategically design the teaching and learning experience on every module I teach. I firstly set the aim and learning objectives of the module in alignment with the aim and objectives of the curriculum, and the teaching strategy of the department. These, together with the cohort (size, level, diversity etc), my personal teaching experience, and resources available (classrooms, labs, technological equipment, time, staff, etc) will allow to design the content, priorities, order, teaching methodologies, assignments, assessment strategy and feedback, tools etc. that will be used to achieve the strategic goals. Finally I set an evaluation strategy (design the ways to get feedback from students for all these). I work with students at early stage (particularly at the start of a module or semester for key areas like programming modules for entry level and/or multi-discipline groups) identifying any differences in their previous education and learning approaches and their individual needs. This is a continuous iterative process that will go through refinement and refactoring in all levels throughout the years. This is the inner circle. There is an outer circle too.

Since learning and teaching is my passion, I enjoy every moment I spend with my students. Following a holistic approach towards teaching I support students learning experience inside and outside the classroom. So in another higher level continuous iterative process, I am trying to achieve my learning objectives and to provide employability skills to my students combining formal lectures with several extra-curricular activities, enhancing their learning experience. I am designing my lectures in a way that will allow the individual to develop skills and facilitate learning in an inclusive environment of collaborative learning, feeling an active member of a team. I encourage them to embrace diversity and share/discuss their personal experiences to exchange best practices and create knowledge communities, using several teaching formats such as discussion groups, student conferences, student chapters and groups. I also include group work, with groups representing diverse cultures and nationalities, working on case studies following story telling approaches taking into account the background and interests of students.

The final result is an enhanced learning experience, integration in all levels, inclusion, and development of enhanced employability skills for all students.

Teaching Experience

Highlight of my teaching career was **receiving the Senate Award for Sustained Excellence in Learning and Teaching**, The University of Sheffield Senate Award for Excellence in Learning and Teaching, rewards those who have a demonstrated impact on student learning alongside sharing what they do with colleagues and having a profile in the broader University or educational community.
Commenting on my nomination, the Panel said:

- A truly inspirational academic who provides leadership and support from UG to PhD
- Clearly a colleague who makes a great contribution to student learning and has done much to enhance that experience through extra-curricular activities.
- His passion for both his subject and for learning and teaching shine out throughout the nomination. His approach is broad-ranging covering key learning and teaching strategic areas including interdisciplinary teaching, integration and employability.

Very active in teaching at any level having taught more than 20 (postgraduate and undergraduate) different modules over the last 15 years. I personally proposed, designed and implemented 5 modules in the BSc level and 7 in the MSc level, and all of them had a significant positive impact in my department. Main areas I am teaching are:

- Programming languages (I taught in every level, from 1st level undergraduate to MSc level)
- Security
- Software Engineering
- ICT for Strategic Management

My teaching consistently receives **outstanding student feedback** (always at least 15% higher rating in students' evaluations than the average in my department and amongst the top of the University of Sheffield). Especially in MSc courses I receive on average 4.7 in a scale 0 to 5, when the average of the department is 4.1. Also the comments of the students are really very flattering.

Actively involved in strategic level in the **design and implementation of all MSc and BSc programmes and curricula** offered by the CS department since 2001. Directly involved in every **quality assurance audit** (BCS, QAA, BCA) highlighting the ones (3 of them) from the British Computer Society (BCS), where I had a major role due to my administrative position in the department.

Very significant and important contribution with a great impact were the **initiatives on extra-curriculum activities**, highlighting:

- the Student Spring Symposium (annually for more than 15 years now),
- the ACM and ACM-W student chapters I sponsor (both of the oldest in Europe),
- the Java User Group, and the Ruby User Group I established in Thessaloniki,
- my active support to almost any Tech User Group in Thessaloniki, and

overall my passion over all these years to inspire students to engage and continue their education outside of the curriculum aiming especially to improve their professional and employability skills.

- Founder of an annual event for and from students, the International Student Spring Symposium (S3) that is organized every spring since 2002. Organized together with students this annual student conference (that the Greek Computer Society and SEERC supports after 2003) with great success and participation of students of all the academic institutions in the wider region. After a very successful event and the feedback from students, this event became a trend (an annual event) and the result was to organize it in a yearly basis until today. Thus, the Student Spring Symposium (S3) is an annual event with continuous success and international recognition today with more than 120 participants in average.
  S3 gave birth also to the idea to form the CITY ACM student chapter. The support of CITY ACM student chapter from the beginning offered a lot but the main highlight was the invited speakers they managed to bring through the ACM distinguished lectures program and the support of CITY College.
    - Students benefit in many different ways:
        - learn how to organize a conference,
        - work in groups fostering especially multidisciplinarity;
        - present in big audiences, developing from the early stages of their career their presentation and reporting skills,
        - they are inspired after the success of this event to be involved in other extra curricula activities
        - having the chance to attend presentations from high caliber invited speakers
        - inspires them to pursue an academic career and improves their self-esteem
    - The faculty also benefits in many ways:
        - Students from the wider region attend and they become members of an active scholar community.
        - It increases the satisfaction of students since they acknowledge that they have the opportunity to develop further in the institution with several extra-curriculum activities.
        - Students and staff were trained to organize low budget events and the ground was prepared and the expertise was there when organizing several international research conferences.

My involvement with all the technology related communities allowed me to well **establish a strong network with top professionals working in high-tech companies** not just in Greece but mainly all over Europe and USA. That proved to offer many opportunities to my students, especially the ones I supervised, to not only enhance their knowledge and improve their employability skills but also to expand their networks and find jobs immediately after their graduation.

Founder of CITY College ACM-W Student Chapter, also called Muses, on June 2012. It is the first Women Student Chapter in Europe. Women were always substantially less than men in the department having even extreme cohorts with no women at all, and usually cohorts that had 2-3 women. The aim was to increase the number of women in the Computer Science department and not the ratio, trying to build a critical mass that will provide the sense of community. This fact **enables women inclusion** despite the small number in the cohort. Post graduate students also immediately contacted the chapter and joined. This offered the opportunity in very short time (less than a month) a **community of learning** to be formed that enabled education of diverse groups of students

including those with different life experiences, and teaching UG and PGT students together in extra curriculum activities that boost students' creativity and collaboration.

S**upervised more than 150 BSc students** in 20 cohorts, achieving to have at least one student receiving one of the top three (3) BSc project award in 16 of these 20 cohorts. Also I **supervised more than 50 MSc students** achieving similar results. Apart from the top first class theses, I am also proud having many students achieving marks significantly higher than their average mark in taught modules. Almost all those students found a job immediately after their graduation and have very successful careers. Those graduates enhanced the industrial network mentioned above, both in quantity and quality of the relationships with many of these companies and their employees, amplifying the opportunities available for current students.

Over the last decade and more, most cohorts I had to teach were consisting of **diverse groups of students**. Globalization in combination with the extensive ability to communicate with the use of technology over the last decades had a significant impact to higher education. The inevitable mixture of students from different educational systems, life experiences and cultures created a need for teaching strategies that are extremely flexible and employ innovative ideas that will enable a community feeling to emerge. Ignoring students' previous learning styles, educational cultures, and most importantly knowledge and skills inadequacies results in the formation of minorities that do not engage, are discouraged and eventually abandon their studies. I therefore established a philosophy about teaching diverse groups of students that requires focusing to the aim and learning objectives of every taught unit and the curriculum in general, and rethinking ways, techniques and tools used to achieve them. This led me to redesign the Computer Science Level 1 introductory programming unit in computer programming to make it accessible to a much wider audience. This was achieved by:

- abstracting out everything but the essential computational skills in the first half of the semester,
- introducing an educational tool aiming to bolster the skills of the students, both with or without programming experience, in the area that traditionally is the most difficult for them. To achieve the desired results the tool should enable visual and immediate output of the efforts of students, should be easy to use it with few minutes training, should have a fun factor, and should allow students to use it in a creative way.
- integrating the curriculum with carefully designed extra curriculum activities

D**eveloped in 2011 a tool** that enables everyone (no matter the background, and age) to start **learning programming** and develop some essential computational thinking skills. A number of publications and an initial evaluation to ages ranging from 7 to 77, provide some very promising feedback. The teaching methodology around this tool reached a maturity level that allows to go large scale in order to maximize the impact to the community in the following years.

Mentor was designed to abstract unnecessary complexity, focus on the basic computational skills that are essential for the students to proceed successfully with their studies and most importantly enhance students' engagement. Mentor offers ability to understand the basics of programming and acquire basic computational skills by abstracting all unnecessary in the beginning and focusing to the important. By simulating two dimensional worlds it provides a visual direct output and it is very simple to learn and start using the language (less than 1 hour in theory and 1 hour practice), enabling any student to experiment and engage early. It boosts also students' creativity avoiding the usual computer science problems that are boring for most students. The students had to solve problems for coursework inspired by science fiction stories and fairytales!

The first results conjecture that students with limited or no background in programming managed to engage early and intuitively develop all required knowledge and skill. At the same time the ones with any level of experience on programming managed to achieve deeper understanding of the fundamentals of computer programming and well establish their previous knowledge, while at the same time enjoying the learning process and not loosing their focus at any phase.

No matter the initial diversity, working in group projects to simulate stories inspired by fairytales and Sci-Fi novels, created a sense of community. Because of the simplicity of such stories and their inclusion to every culture, the simplicity of the common tool they were all using for the first time, everyone had something to share. This enabled the emergence of a **community sharing experiences** and giving rise to new ones trying to solve the problems in a way similar of playing creative games in a cooperative and collaborative environment. And while all these are happening students have fun at the same time. The end result was to achieve to have at the end of the first semester a **well-balanced cohort eliminating the initial diversity** on the level of programming skills and knowledge.

Dealing with large groups of students, I am developing a new learning approach that I am calling **"from bit to byte"**. I am implementing the idea in the computer programming courses. The idea is to further enhance collaborative learning by having opportunities for learning and assessment as an individual (2^0), but also as a group of two or four (2^1 and 2^2) and then expand the idea with teams evaluating each other and collaborate forming a bigger team for a better common solution (2^3) in order to compete with other teams in a championship manner (2^4 to 2^7, depending on the size of the group) all the time in a healthy competition that will provide feedback in many

levels and not from one but many, trying for the best solution to emerge from the community, enabling collaborative learning and exchange of best practices. Some steps are assessed formally and some steps include gamification features in order to enhance student engagement.

Based on the above ideas a **number of papers are in working versions highlighting my shift to education oriented research**.

Regarding though my teaching, what makes me proud is that there are some hundreds of highly-respected and influential computing professionals, all around the world, that have been mentored/learned to program with me.

Teaching Activities

Module Title	Contact Hours
Postgraduate Level	
Cyber Security and Forensics (MSc in Advanced Software Engineering / Computer Science Department, CITY)	36
Security Engineering (MSc in Advanced Software Engineering / Computer Science Department, CITY)	36
ICT for Strategic Management (MSc in Business Management and Technology / CITY and MSc in Management, Business, Innovation and Technology / ATHENS TECH)	28
Advanced Java Programming (MSc in Advanced Software Engineering / Computer Science Department, Faculty of Engineering / University of Sheffield)	12
Crash Course in Databases (MSc in Advanced Software Engineering, CITY)	6
Undergraduate Level	
Information Security (3rd level, 2005 – today)	36
IT Strategy in Business (3rd level, 2016 - today)	36

Previous Teaching Activities

Since 1998, I have taught many CS modules both in undergraduate and postgraduate higher education level. Apart from the above that I still currently teach, I taught more analytically:

POSTGRADUATE LEVEL

1. Engineering Internet Applications (using Ruby on Rails) (2012-2013) (CITY MSc in Software Engineering and Telecommunications) (9 contact hours / 10-25 students)
2. Crash Course in Object Oriented Programming (2004-2012) (using Java), CITY MSc SET (2003-2009) (12 contact hours / 10-25 students)
3. Advanced Software Engineering, CITY (2008-2009) (36 contact hours / 10-25 students)
4. Object-Oriented Programming with Java, MSc in Information Systems, CITY (2002-2005) (30 contact hours / 10-25 students)
5. Web Content Management, CS department Aristotle University of Thessaloniki, Greece (10/2006-02/2007) (36 contact hours / 25-50 students)
6. Internet Technologies, Applications and Services, CS department Aristotle University of Thessaloniki, Greece (02/2005-08/2005) (48 contact hours / 25-50 students)
7. Evaluation of IT Systems, CS department Aristotle University of Thessaloniki, Greece (02/2004-09/2004) (24 contact hours / 25-50 students)

UNDERGRADUATE LEVEL

1. Programming Principles and Algorithms (using Java), (1st level, 2002-2018) (36 contact hours / 25-50 students)
2. Programming Methodology and Design (using Java), (1st level, 2002-2018) (36 contact hours / 25-50 students)
3. Introduction to Object-Orientation (using Java), (1st level, 2002-2018) (36 contact hours / 25-50 students)

4. Object-Oriented Programming (using Java), (1st level, 2002-2018) (36 contact hours / 25-50 students)
5. Database Systems, CITY (2nd level, 2005 – 2015), (36 contact hours / 25-50 students)
6. Communication Skills and the Internet, CITY (1st level, 1998-2004) (48 contact hours / 25-50 students)
7. *Discrete Maths,* CITY (1st level, 1999-2003) (36 contact hours / 25-50 students)
8. *Design and Development of Network Applications (using Java)*, CS department Aristotle University of Thessaloniki, Greece (4th level, 2004-2005) (48 contact hours/ 25-50 students)
9. *Software Engineering*, CS department Aristotle University of Thessaloniki, Greece (3rd level 2004-2005) (48 contact hours / 150-200 students)
10. *Software Engineering*, CITY (3rd level, 2002-2003 and 2008-9) (36 contact hours / 150-200 students)
11. *Internet Technologies, Applications and Services*, CS department Aristotle University of Thessaloniki, Greece (3rd level, 2004-2005) (48 contact hours / 150-200 students)
12. *Theory of Languages and Automata*, CS department Aristotle University of Thessaloniki, Greece (1st level, 2003-2004) (48 contact hours / 200-300 students)