School of Computer Science and Engineering

4COSC008C Trends in Computer Science

Lecture- Week 1

4COSC008C Trends in Computer Science Lecture 1

Welcome!

Module Team

Ms. Sulochana Rupasinghe (sulochana.r@iit.ac.lk) - Module Leader

Prof. Prasad Wimalaratne (prasad.w@iit.ac.lk)

Mr. Cassim Farook (cassim.f@iit.ac.lk)
Mr. Rathesan Sivagananalingam (rathesan.s@iit.ac.lk)
Ms. Sulari Fernando (sulari.f@iit.ac.lk)
Ms. Dileeka Alwis (dileeka.a@iit.ac.lk)
Ms. Hiruni Samarage (hiruni.s@iit.ac.lk)
Ms. Lakna Gammedda (lakna.g@iit.ac.lk)

Mr. Kushan Barathi

Outline of the lecture

- In this lecture we will introduce the Trends in Computer Science module.
- We will discuss its content and assessment.
- We will finish with a short discussion on the principles that underpin Computer Science and Software Engineering.

4COSC008 Aim

From the module pro-forma (syllabus)

"The module focuses on Trends in Computer Science which currently attract considerable industry and academic interest. It allows students to acquire research skills which will prepare them for the group and final year project and will support their employability prospects. At the same time, it introduces key aspects of working as a professional in the world of computing, including consideration of ethics, privacy, data protection and confidentiality, and how these are incorporated into professional codes of practice such as the BCS Code of Conduct."

4COSC008C will cover

- A link between the past and the future of Computer Science and Engineering.
- Important academic skills, including undertaking research and present your findings in an oral and written form, useful both for your studies (e.g. 2nd and 3rd year subjects) and for your professional career.
- Exiting developments in Computer Science and Engineering (e.g. Quantum Computing)
- Areas where there is an urgent need for more specialists (e.g. Cyber Security, Artificial Intelligence)
- Areas where exciting new research is taking place (e.g. Machine Learning, Internet of Things).

4COSC008C will give you the opportunity

- To meet members of School staff and their research
- To feel more confident when undertaking research
- To become familiar with areas you might want to engage with later in your studies, as part of your option modules; group project; and final year project.
- To meet employers and boost your employability.

4COSC008C structure

- The module will be delivered in through pre-recorded and live on-line lectures, as well as through onsite tutorial (practical) sessions and additional engagement week activities.
- Please consider the **module**'s **schedule** and **pro-forma**, which are published on **Blackboard (BB)**.

How you will be assessed:

CW1-Group presentation and report- weighting 40%

The assessment consists of two Courseworks:

CW1. Group Presentation and report

- A 5 min. presentation, with PowerPoint slides, accompanied by a short individual 1000 words report, worth 40% of the module. The specification is already published on BB. Presentations involve a choice of 2 topics with 5 sub-questions each. Groups will choose their topic (and individual students your sub-question) on week 2.
- Your tutor for the practical tutorial session is your first point of contact for the assessment of this module.

4COSC008C Assessment 1:
Group presentation and report- weighting 40%

- You can chose your groups during tutorials (week 2). You should let your 4COSC008C tutor know during the week 2 tutorial: who are the members of your group; which topic you chose; and which sub- question each one of you chose. Students who will be absent during the week 2 tutorial, or unsure which group to join, will be advised and allocated to a group by your tutor.
- Groups, topics and sub-questions allocated will be published on BB, together with the date presentations will take place on week 3. Presentations will take place during tutorials of weeks 7 and 8.

CW1 Topics

1. Legal, Social and Ethical concerns for Computer Scientists and Software Engineers

1a. Overview of Legal, Ethical and Social concerns for Computer Scientists and Software Engineers. Using one area of application as an example (e.g. Self-Driving Cars, face recognition applications, AI interview selection applications, among others) focus on the **Legal aspects** that such applications entail and that you need to be aware of.

1b. Overview of Legal, Ethical and Social concerns for Computer Scientists and Software Engineers. Using one area of application as an example (e.g. self-driving cars, face recognition applications, AI interview selection applications, among others) focus on the **Ethical and social aspects** that such applications and that you need to be aware of.

1c. How does the need for a Computer Scientist/Software Engineer to familiarise themselves with new Computing trends, and their associated social, legal and ethical concerns, relate to a Code of Practice (such as the BCS Code of Practice)? What is the purpose of such a Code of Practice?

1d. What impact might new computing trends have on a user's privacy? Provide examples of such Trends and discuss the legal and ethical implications of collecting data about users and ways our users' privacy can be protected.

1e. How are individual users legally protected? What are the implications for a company if a security breach occurs?

CW1 Topics

2. Machine Learning

2.a. Overview of Machine Learning. How does it compare with conventional computing? 2.b. Overview of Machine Learning. Describe and compare two different machine learning techniques.

2.c. Overview of Machine Learning. How does the need for a Computer Scientist/Software Engineer to familiarise themselves with Machine Learning relate to a Code of Practice (such as the BCS Code of Practice)? What is the purpose of such a Code of Practice?

2.d. Overview of Machine Learning. What are the opportunities it brings to developers? What threats does it pose to society?

2.e. Overview of Machine Learning. Provide examples of legal, ethical and social concerns associated with Machine Learning.

4COSC008C Assessment 1:
Group presentation and report- weighting 40%

- Students will present onsite, during your tutorials of weeks 7 and 8. You will also need to submit a copy of your slides and of your short individual 1000 word report on BB on or before Friday 4th November at 13.00 pm (SL Time).

4COSC008C Assessment 2:
Portfolio of written work - weighting 60%

CW2. Portfolio of written work

- Your second individual CW for this module will involve three reflections (of approximately 850 words each), on topics related to the lectures. One of them will focus on your employability engagement.

4COSC008C Assessment 2:
Portfolio of written work - weighting 60%

B. Portfolio of written work

It is, therefore, very important to **attend at least one of the employability related events**, published on **Engage**, or organise a meeting with a career advisor.

We will discuss further how to maximise employability on week 8.

- Further details on CW 2 will be provided on week 8.

Tutorial exercises and Self-managed activities

- Tutorial exercises will support your learning and communication skills, as well as your assignment preparation.

- You will also be advised to undertake independent study exercises. This will not be formally assessed, but you will find it helpful, as it will allow you to prepare for the following week's lecture and /or it will support your CW preparation. It is your responsibility to attempt them, as part of the total 200 hours that you have to devote to this module.

Computing Principles

- Reference: the slides below are based on Schneider, G.M. and Gersting, J.L. (2017) Invitation to Computer Science, Boston: Thomson, 8th edition chapter 1.
- You might also want to consider

  Brookshear, J.G. (2014) Computer Science: An overview, Boston: Pearson, chapter 1.

Computing Principles

- In order to ensure we all understand the term Computer Science (CS), in its broad sense, which encompasses Software Engineering, we need to define it.
- Often in newspapers, magazines, and TV, we hear reports on new advances in computing related technologies. (Can you name some?)
- However, Computer Science differs from other fields, in that many people cannot follow the types of problems faced by Computer Science professionals (Can you name any?).

Computing Principles

- Most people can describe relatively accurately most scientific fields, but not Computer Science.
- E.g. what is biology?

[the study of living organisms]

- What is Computer Science?
- What were your taught as part of Computing/ICT at school/college?

What Computer Science is all about

- Misconception 1:

‘Computer Science is the study of computers’.

- The field of Computer Science is broader that the use of computers or the development of computer software.
- The above definition is incomplete: Theoretical work in computer science took place in 1920-1940, well before computers physically existed.

Computing Principles

- Pre-1940 work was considered part of applied mathematics/logic. Computer Science was recognised as a discipline on its own merit in the late 1950's.
- Today, there are branches of CS, such as Theoretical Computer Science which focus on the logical and mathematical properties of problems and their solutions. Such problems are often investigated not with actual machines, but with formal models of computation (with pen and paper).

Computing Principles

- Fellows & Parberry (1993):

  “Computer Science is no more about computers than astronomy is about telescopes, biology is about microscopes, or chemistry is about beakers and test tubes. Science is not about tools. It is about how we use them and what we find out when we do.”

Computing Principles

- Misconception 2:

‘Computer Science (CS) is the study of how to write computer programs’.

- Programming is a **very** important **tool**, by which we can study new ideas and built and test new solutions. However, CS is **not** about how to programme in Java or in C++, despite the fact that programming is considered as a universal entry to the discipline, and forms part of a common introduction to CS.

Computing Principles

- ‘ In Computer Science, it is not simply the construction of a high quality program that is important, but also

  -the methods it embodies;

  -the services it provides;

  -and the results it produces.

- ‘A program is means to an end, not the end itself’.

Computing Principles

- Misconception 3:

‘CS is the study of the uses and applications of computers and software’.

Packages such as word processors, database systems, and imaging software among others, are **not** what CS is about; learning how to use a package is **not** what CS is about. A computer scientist is responsible for *specifying, designing, building, and testing* software packages as well as the computer systems on which they will run.

So, how do we define CS?

- There are many definitions for Computer Science. Following Gibbs and Tucker (1986) definition:

‘Computer Science is the study of **algorithms**, including

- their formal and mathematical properties;
- their hardware realisations;
- their linguistic realisation;
- and their application.

So, how do we define CS?

- The formal and mathematical properties of algorithms:

  Study the behaviour of algorithms to determine whether they are correct and efficient.

So, how do we define CS?

- The algorithms' hardware realisations:

- Design and build computer systems which are able to execute algorithms.

So, how do we define CS?

- The algorithms' linguistic realisation:

- Design programming languages and translate algorithms into these languages so that they can be executed by the hardware.
- [Questions you might want to consider: What are natural languages? What is machine language? How close are programming languages to machine language? What is a first generation programming language? What is a second generation programming language? What is a third generation programming language? Do you know of any examples?]

So, how do we define CS?

- The algorithms' application:

- Identifying important problems and designing correct and efficient software packages to solve these problems.

Computing Principles

- The central concept of the Gibbs and Tucker (1986) definition is the *algorithm*.

  [Can you try to define the term, based on what you learned so far in your respective programming modules?]

Definition of an algorithm

- ‘A procedure for solving a mathematical problem in a finite number of steps that frequently involves repetition of an operation; broadly, a step-by-step method for accomplishing some task’.
- [Question to you: What kind of knowledge do humans posses? Can you mention an example of each? (clearly your answer should include two types of knowledge!)]

And another definition of an algorithm

- ‘A well-ordered collection of unambiguous and effectively computable operations that, when executed, produces a result and halts in a finite amount of time.’.
- Consider the following description of a pseudo-algorithm:

i. Fry the egg.

ii. Take a frying-pan.

iii. Add some form of fat (e.g. butter).

iv. Break the egg or eggs.

v. Serve the egg(s).

Is there something wrong with it?

And another problem

- Consider the following instructions, addressed to a human, from a bottle of shampoo. Assume that a programmer is using it as the basis for a hair_dresser_robot.

What changes does s/he need to make?

i. Wet the hair.

ii. Lather the hair.

iii. Rinse the hair.

iv. Repeat if necessary.

Discussion

- How many times is the robot to repeat the operation?
- When is it to stop?
- Is it ever allowed to stop?
- What if there is no water to wet hair? Etc.

In a nutshell, Computer Science is about…

- The study of CS involves the way we use tools such as computers (or computer programs) and what we find out when we do.
- Apart from designing hardware or putting together complex applications, we are interested in the study of the methods involved, the services provided and the results that are produced.

In a nutshell…

- If we were to compare the performance of a machine with that of a human, we would have to recognise that
- Machines do not get tired; do not get emotional; operate in a consistent way 27/4 etc.
- However, humans are **creative** i.e. they are able to come up with novel ideas; and able to try previous knowledge in new environments.

Further activities

- Please consider researching in your own time
- Programming paradigms: what are they? How do they relate to programming languages?
- The von Neumann computer architecture

4COSC008C Trends in Computer Science

Please do not hesitate to ask any questions during the lecture!

Thank you!