# COMPUTATIONAL THINKING: HOW TO TEACH?

# HOW WE TEACH COMPUTATIONAL THINKING?

Dayang Norhayati Abang Jawawi
School of Computing
Faculty of Engineering
Universiti Teknologi Malaysia
Skudai, Johor, Malaysia
dayang@utm.my

DR. SHAHLIZA BTE ABD HALIM
PM DR. RADZIAH BTE MOHAMAD
DR. MOHD ADHAM BIN ISA
PM DR. DAYANG NORHAYATI BTE ABANG JAWAWI
DR. HAZA NUZLY BIN ABDULL HAMED
DR. NOR AZIZAH BTE SA'ADON
PM DR. ROSBI BIN MAMAT

# Session Outline

## 1 Introduction (830 -9am)

- Computing Education (CE)
- Computational Thinking (CT)
- Activity 1

## 2 Teaching CT (9 – 10am)

- From CE to CT
- Teaching CT Trends
- Demo 1
- Activity 2

## 3 Incorporating CT into the Classroom (1030am-1pm)

- Educational Robotics (ER)
- CTER for Undergraduate
- CTER for School
- Demo 2
- Activity 3

## 4 CT TeachingStrategies (2-3pm)

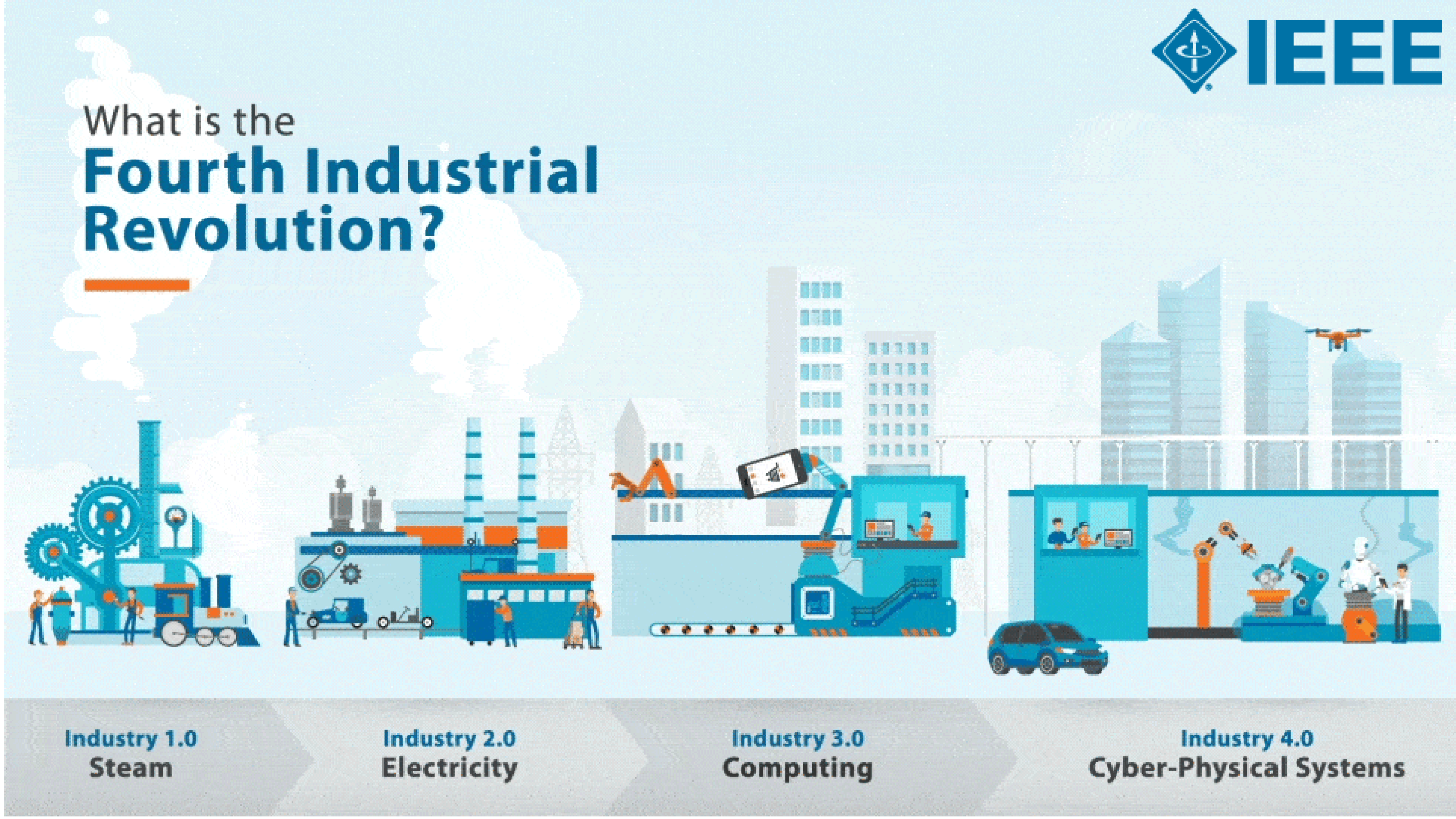
- Teaching CT Online
- Demo 3
- Adaptive Learning Approach

## 5 What we learn (3-330pm)

- New Project
- Learn
- Challenges

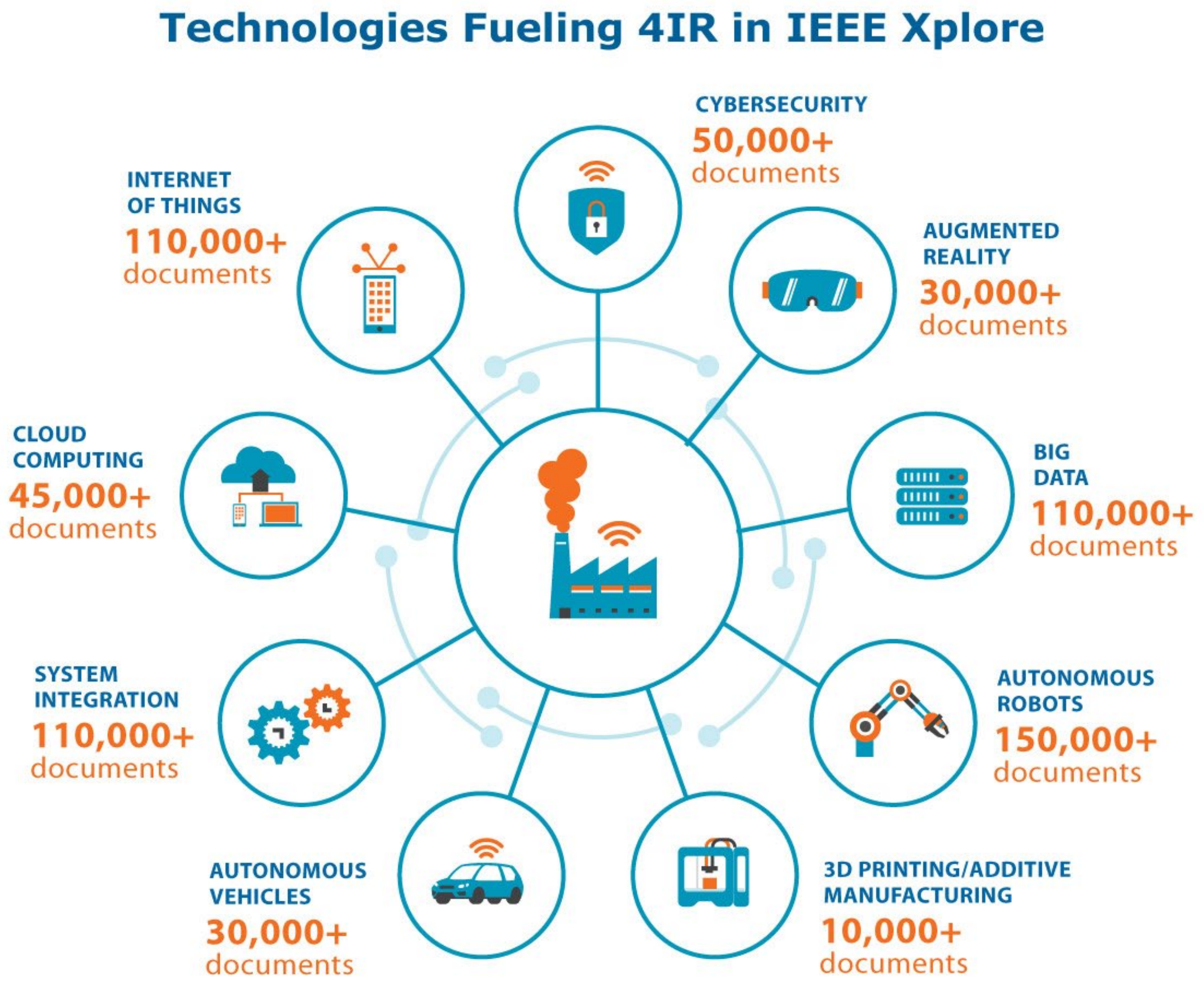
# Session 1
# Introduction

# Fourth Industrial Revolution (4IR)



- we are now on **the fourth industrial revolution (4IR)**, the improvement on quality of life is a promise.
- with the 4IR transforming all aspects of human lives including job and education

# 4IR Emerging Technologies

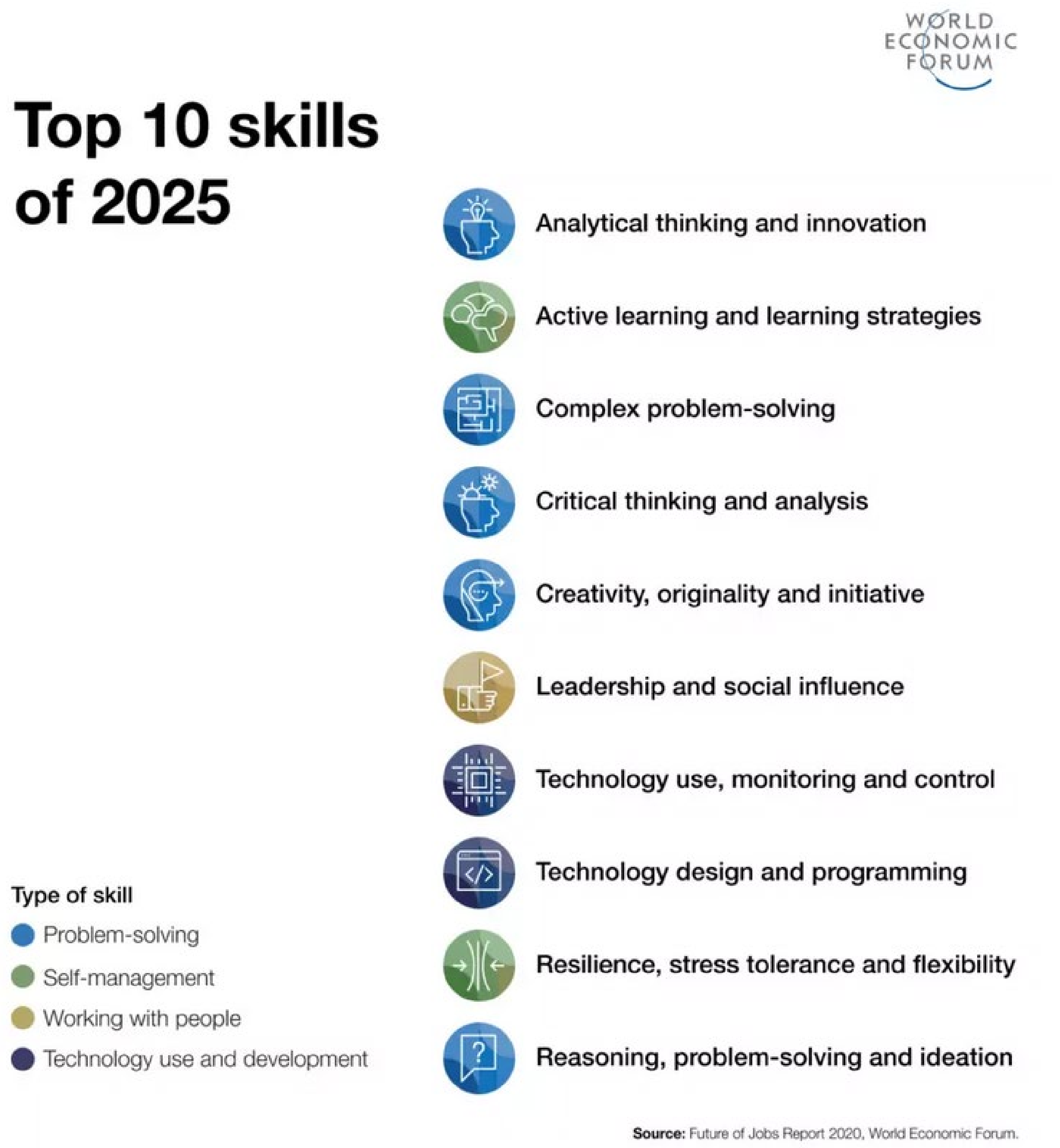


- It is forecasted that huge demand on skills and knowledge for these enabling technologies will be needed.

- Specifically on development of **technologies**.

# 4IR - Change the Way we Teach

# Why Computing?



WORLD ECONOMIC FORUM

## Top 10 skills of 2025

- Analytical thinking and innovation
- Active learning and learning strategies
- Complex problem-solving
- Critical thinking and analysis
- Creativity, originality and initiative
- Leadership and social influence
- Technology use, monitoring and control
- Technology design and programming
- Resilience, stress tolerance and flexibility
- Reasoning, problem-solving and ideation

Type of skill

- ● Problem-solving
- ● Self-management
- ● Working with people
- ● Technology use and development

**Source:** Future of Jobs Report 2020, World Economic Forum.

WORLD ECONOMIC FORUM

## Reskilling needs

40%

of current workers' core skills are expected to change in the next 5 years.

**Source:** Future of Jobs Report 2020, World Economic Forum.

# Computing at School

- 4IR change the way we teach
- Computing@school - to be exposed since school in preparing children and youth for future.

**COMPUTING EDUCATION (CE)**

Teaching how to teach machines

**CHANGE IN CE?**

The changes on computing education syllabus and supporting tool

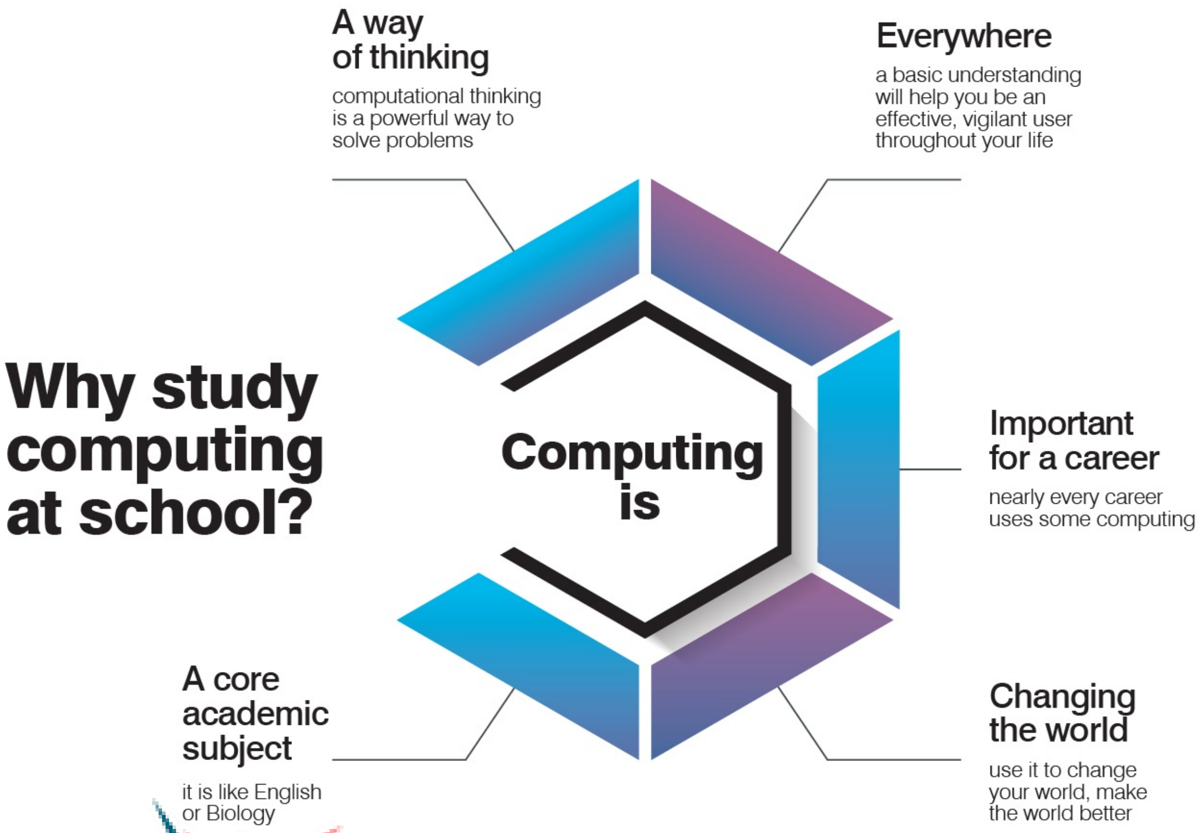
**CHANGE IN SCHOOL EDUCATION?**

Need to develop higher-order thinking skill since childhood

**MORE EXPERTISE**

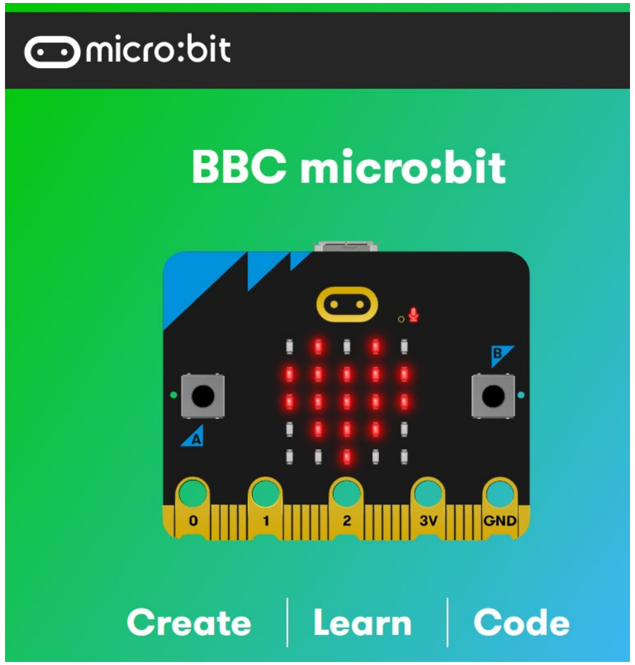
Need high number in expertise on the field to teach others

# Computing at School


Why study computing at school?
Computing is
A way of thinking
computational thinking is a powerful way to solve problems
Everywhere
a basic understanding will help you be an effective, vigilant user throughout your life
Important for a career
nearly every career uses some computing
Changing the world
use it to change your world, make the world better
A core academic subject
it is like English or Biology

Computing at School

https://community.computingatschool.org.uk/

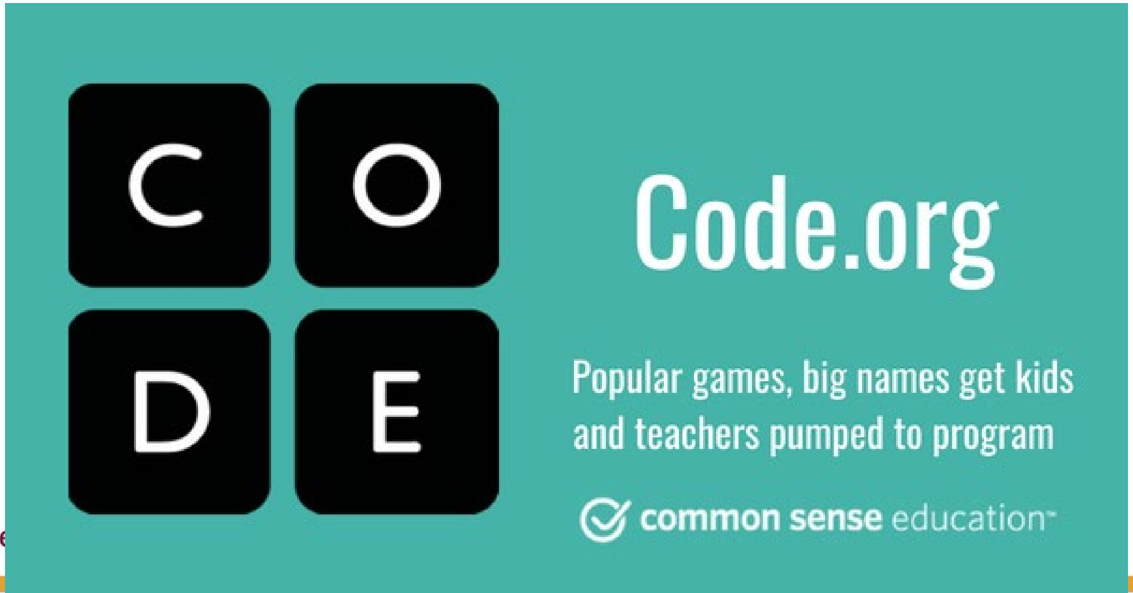


https://microbit.org/

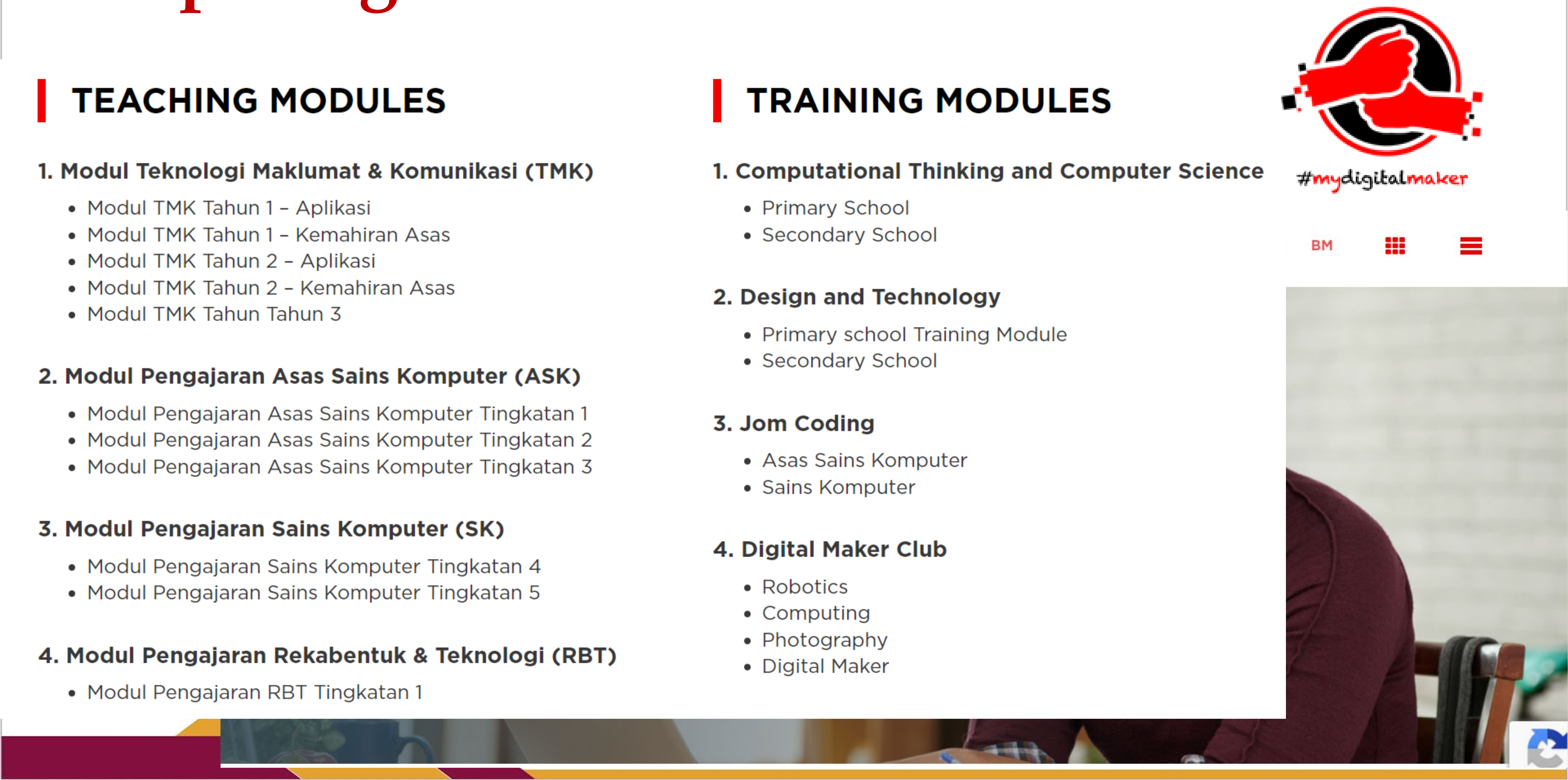
BBC partnership with Microsoft and other companies

https://code.org/

Code.org is supported by generous donors including Microsoft, Amazon, Google and many others.


C
O
D
E
Code.org
Popular games, big names get kids and teachers pumped to program
common sense education

# Computing at School



## TEACHING MODULES

### 1. Modul Teknologi Maklumat & Komunikasi (TMK)

- Modul TMK Tahun 1 – Aplikasi
- Modul TMK Tahun 1 – Kemahiran Asas
- Modul TMK Tahun 2 – Aplikasi
- Modul TMK Tahun 2 – Kemahiran Asas
- Modul TMK Tahun Tahun 3

### 2. Modul Pengajaran Asas Sains Komputer (ASK)

- Modul Pengajaran Asas Sains Komputer Tingkatan 1
- Modul Pengajaran Asas Sains Komputer Tingkatan 2
- Modul Pengajaran Asas Sains Komputer Tingkatan 3

### 3. Modul Pengajaran Sains Komputer (SK)

- Modul Pengajaran Sains Komputer Tingkatan 4
- Modul Pengajaran Sains Komputer Tingkatan 5

### 4. Modul Pengajaran Rekabentuk & Teknologi (RBT)

- Modul Pengajaran RBT Tingkatan 1

## TRAINING MODULES

### 1. Computational Thinking and Computer Science

- Primary School
- Secondary School

### 2. Design and Technology

- Primary school Training Module
- Secondary School

### 3. Jom Coding

- Asas Sains Komputer
- Sains Komputer

### 4. Digital Maker Club

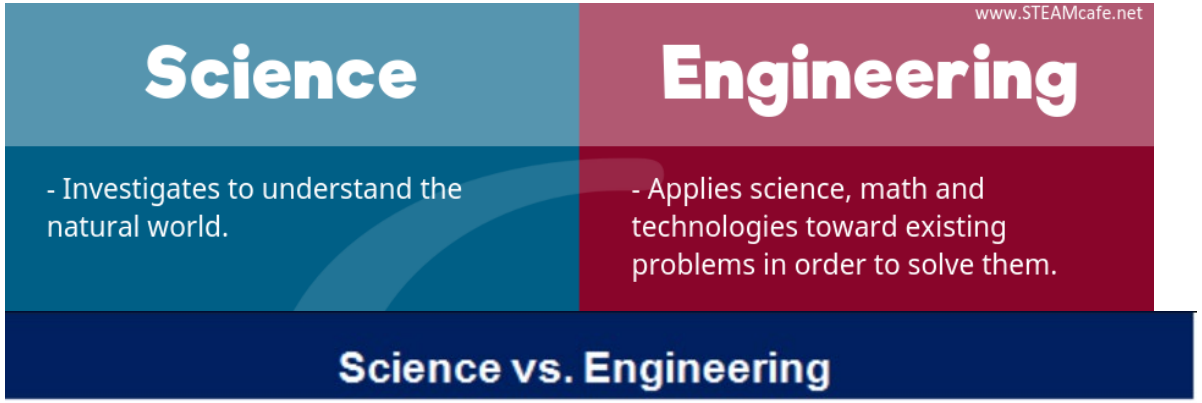
- Robotics
- Computing
- Photography
- Digital Maker

# Computing Education

- CE Is Not the Same as Engineering Education (Mark Guzdial, 2021) Communication ACM

➢ CE - Learning Engineering is not always about becoming an Engineer

➢ CE is about more than transmitting knowledge, but it’s less about promoting professional identity.



www.STEAMcafe.net

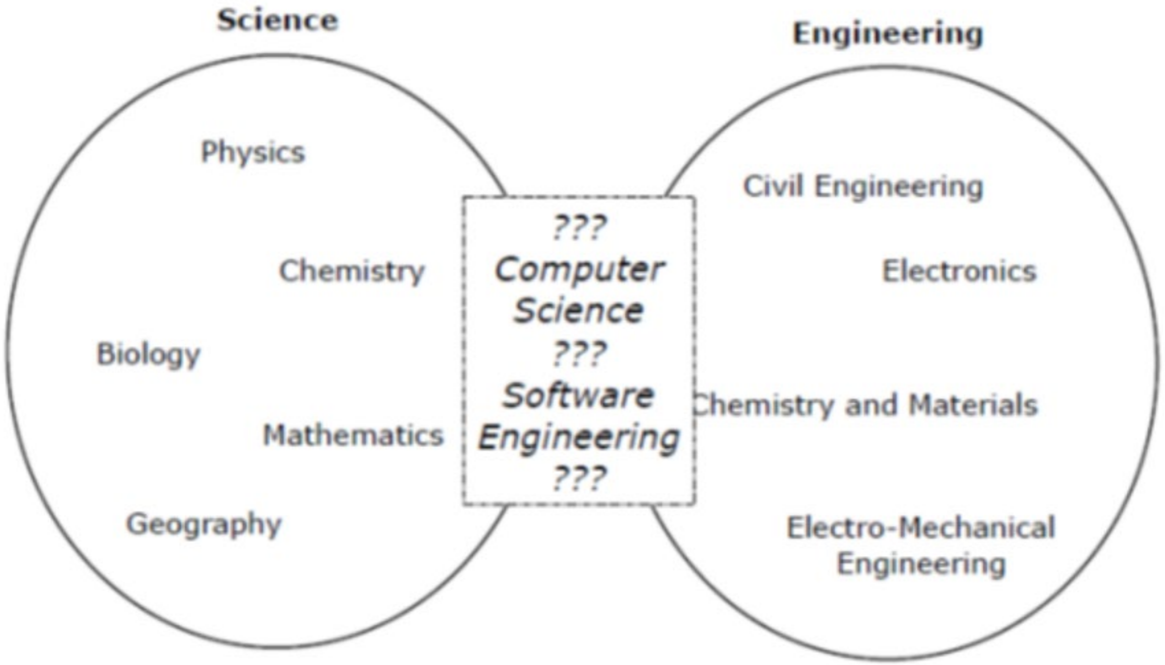
**Science**

- Investigates to understand the natural world.

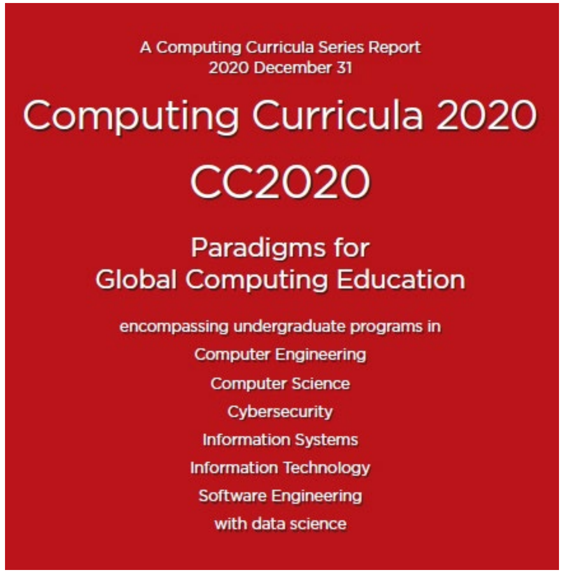
**Engineering**

- Applies science, math and technologies toward existing problems in order to solve them.

**Science vs. Engineering**



# Computing BOK



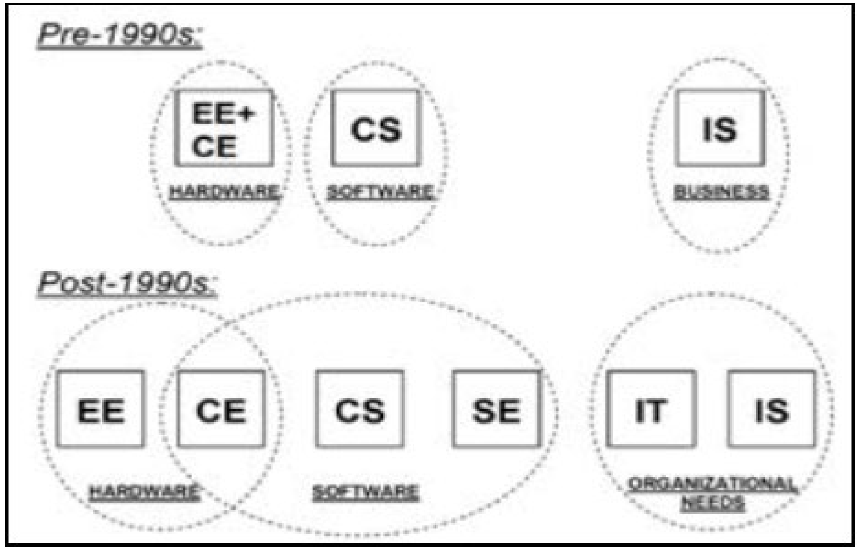
A Computing Curricula Series Report
2020 December 31

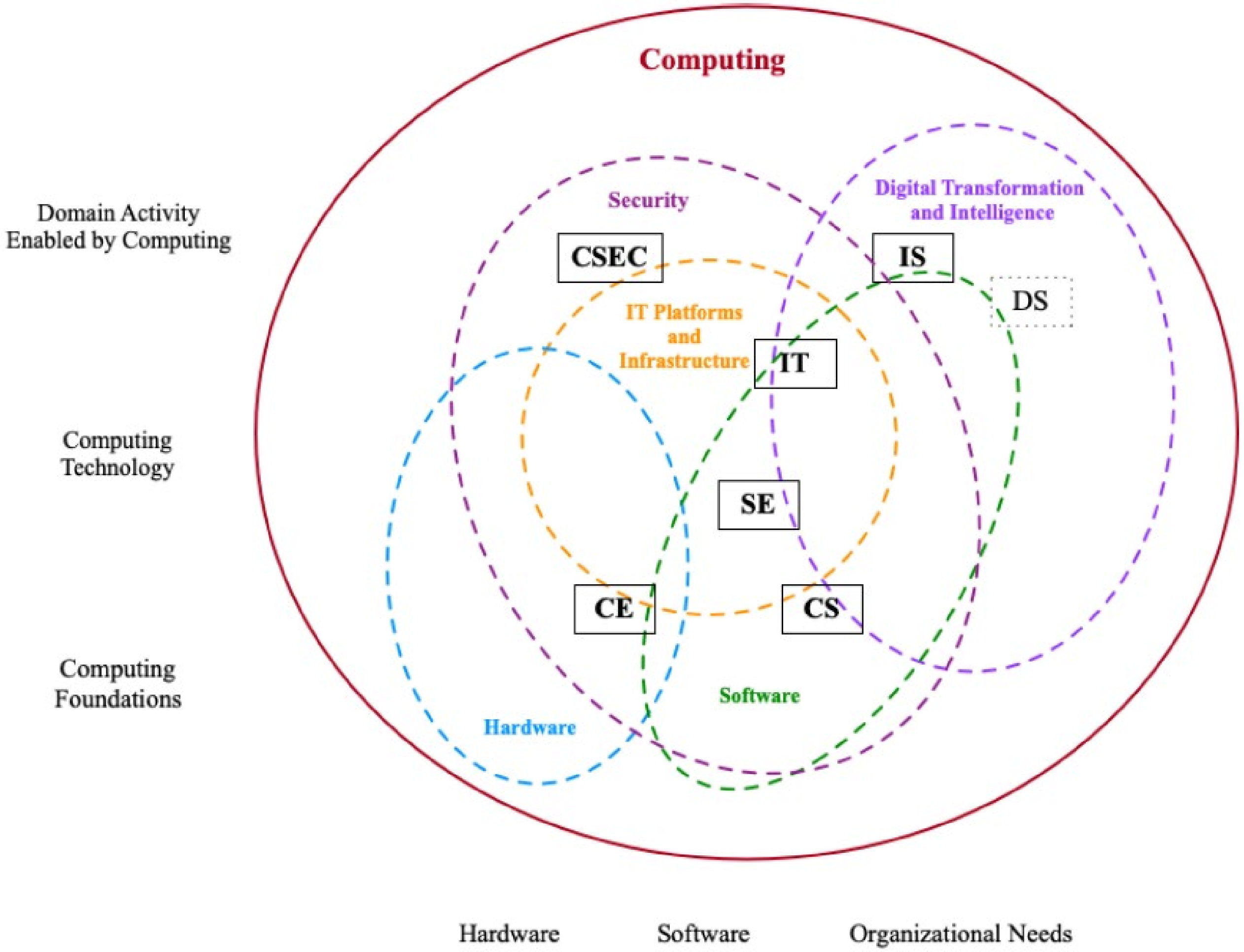
Computing Curricula 2020

CC2020

Paradigms for
Global Computing Education

encompassing undergraduate programs in
Computer Engineering
Computer Science
Cybersecurity
Information Systems
Information Technology
Software Engineering
with data science





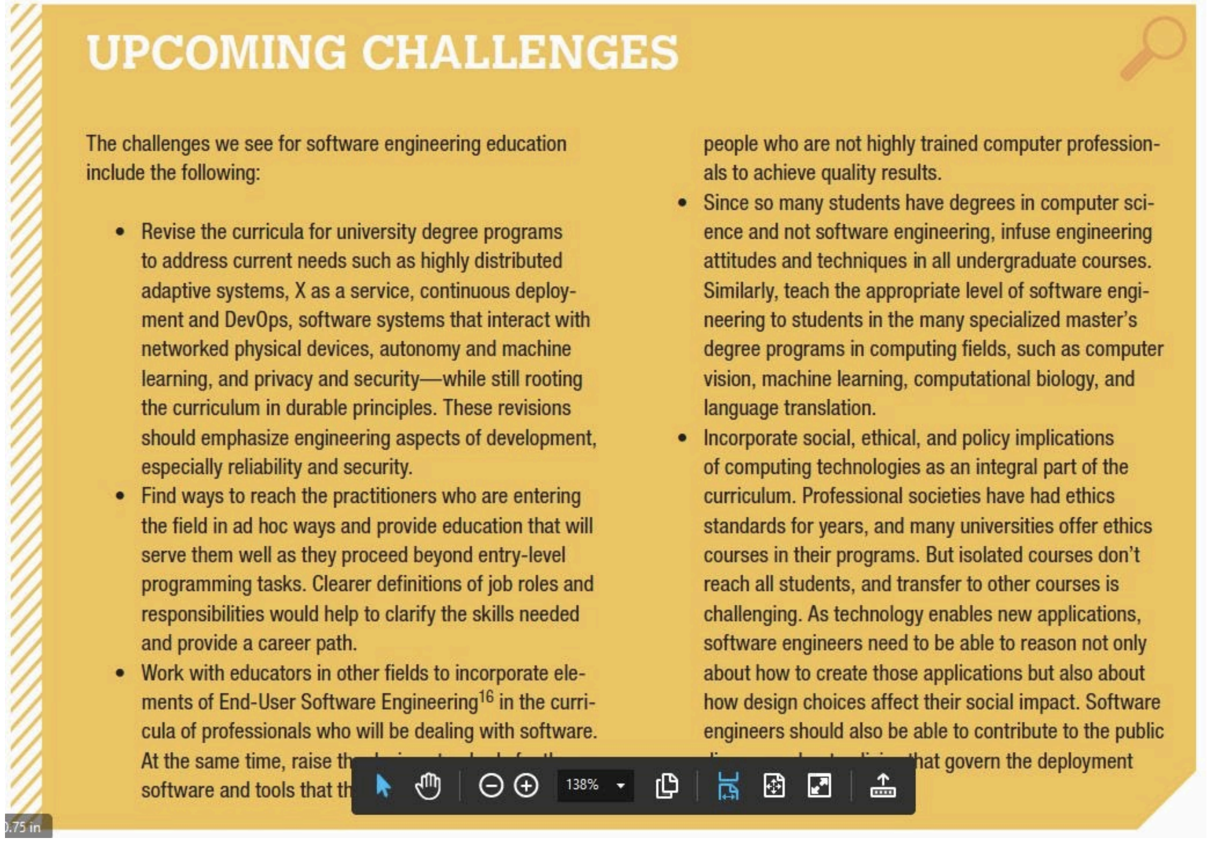
*CE=computer engineering*; ***CS***=*computer science;* ***CSEC***=*cybersecurity;* ***IS***=*information systems;* ***IT***=*information technology;* ***SE***=*software engineering*; ***DS***=*data science*

# What is software engineering

"Software engineering is considered an interdisciplinary engineering approach that draws from a few different branches of engineering."
(Marcus Smith, 2022)

1. Revise the curricula to address current needs
2. Find ways to reach the practitioners who entering the field in ad hoc ways to provide education for career path
3. Work with educators in other fields – End-user SE
4. Infuse engineering attitudes and techniques in all UG computing students
5. Incorporate social, ethical & policy implications of computing technologies in curriculum

Challenges facing software engineering education (SEE)



## UPCOMING CHALLENGES

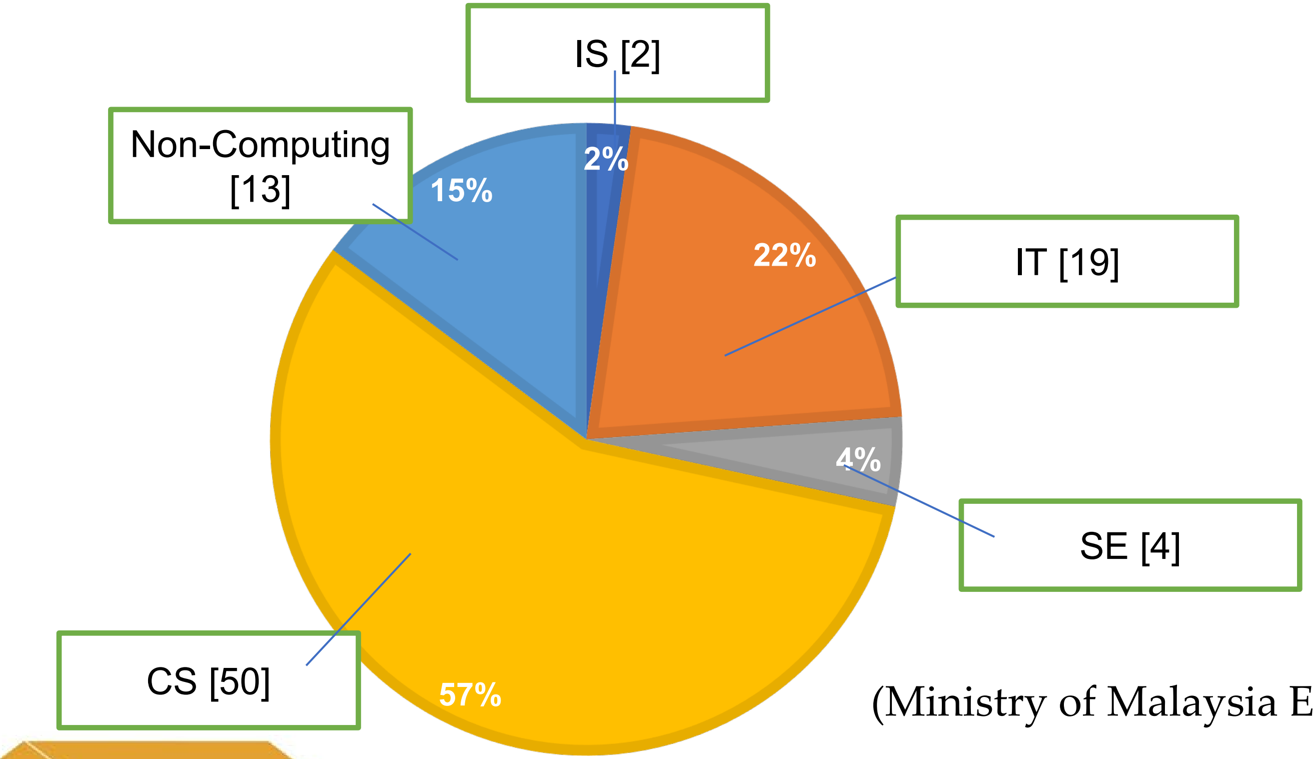
The challenges we see for software engineering education include the following:

- Revise the curricula for university degree programs to address current needs such as highly distributed adaptive systems, X as a service, continuous deployment and DevOps, software systems that interact with networked physical devices, autonomy and machine learning, and privacy and security—while still rooting the curriculum in durable principles. These revisions should emphasize engineering aspects of development, especially reliability and security.
- Find ways to reach the practitioners who are entering the field in ad hoc ways and provide education that will serve them well as they proceed beyond entry-level programming tasks. Clearer definitions of job roles and responsibilities would help to clarify the skills needed and provide a career path.
- Work with educators in other fields to incorporate elements of End-User Software Engineering[16] in the curricula of professionals who will be dealing with software. At the same time, raise th
software and tools that th
people who are not highly trained computer professionals to achieve quality results.
- Since so many students have degrees in computer science and not software engineering, infuse engineering attitudes and techniques in all undergraduate courses. Similarly, teach the appropriate level of software engineering to students in the many specialized master's degree programs in computing fields, such as computer vision, machine learning, computational biology, and language translation.
- Incorporate social, ethical, and policy implications of computing technologies as an integral part of the curriculum. Professional societies have had ethics standards for years, and many universities offer ethics courses in their programs. But isolated courses don't reach all students, and transfer to other courses is challenging. As technology enables new applications, software engineers need to be able to reason not only about how to create those applications but also about how design choices affect their social impact. Software engineers should also be able to contribute to the public
hat govern the deployment

Nancy R. Mead, David Garlan, Mary Shaw, Half a Century of Software Engineering Education: The CMU Exemplar

# CE in Higher Education

- SKPG a graduate detector review system to analyse the status job of the graduates either they are working, furthering studies or actively seeking employment.
- 88 programmes claimed as a computing programme
- 75 computing programmes are actually followed the computing disciplines

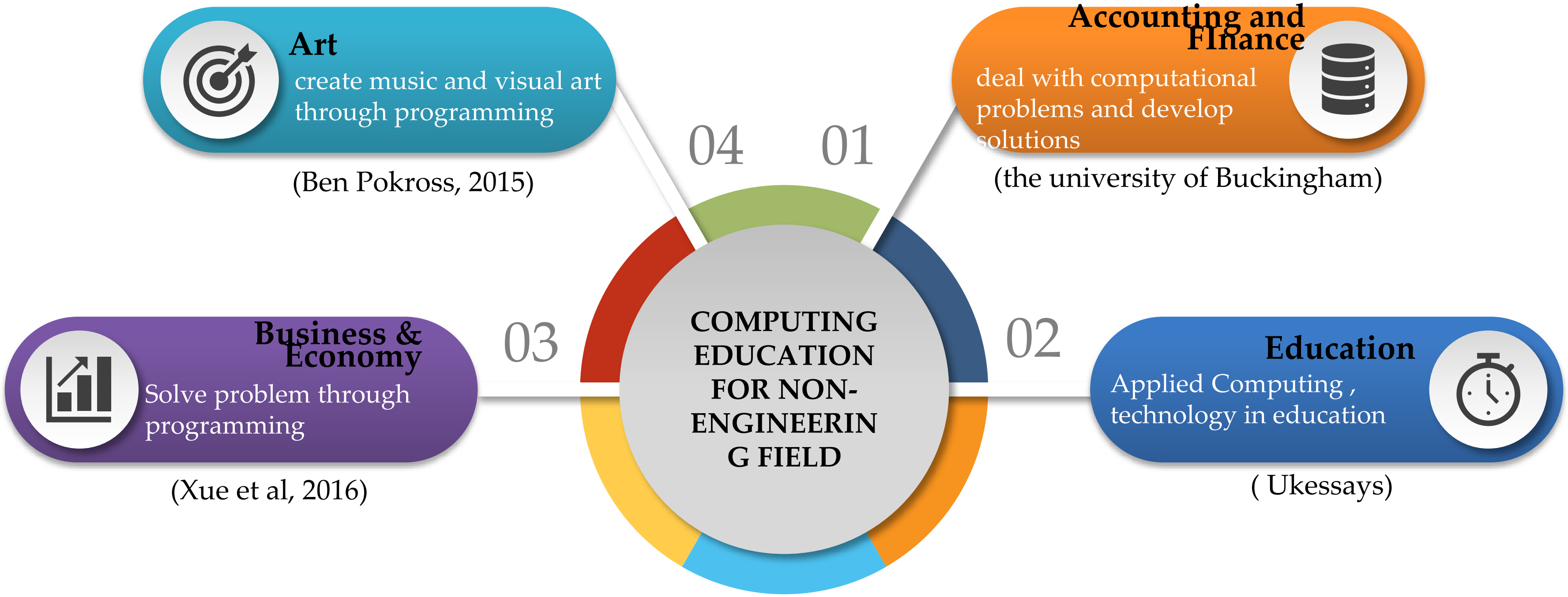


- CE have been crucial as we are in the computational world and facing 4IR
- Non-computing programmes also have applied computing in the courses

(Ministry of Malaysia Education, 2017)

# CE in Higher Education

4IR era: computing NOT for only engineering, science, and computing students



**Art**
create music and visual art through programming

(Ben Pokross, 2015)

**Accounting and FInance**
deal with computational problems and develop solutions

(the university of Buckingham)

**Education**
Applied Computing , technology in education

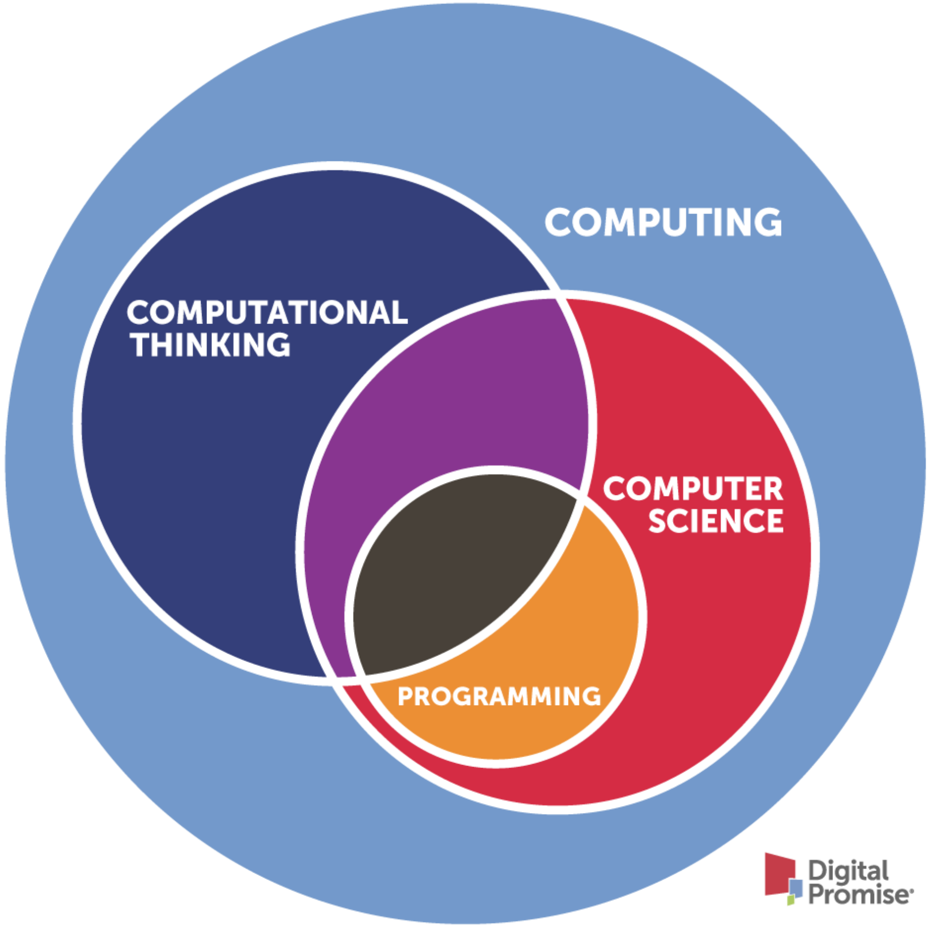
( Ukessays)

**Business & Economy**
Solve problem through programming

(Xue et al, 2016)

# Computational Thinking (CT)

- processes involved in **formulating problems** and their **solutions**
- in a form that can be **effectively** carried out by any processing agent: whether human, computer, or a combination of both.
- fundamental skill to be used by many professions



https://digitalpromise.org/initiative/computational-thinking/computational-thinking-for-next-generation-science/what-is-computational-thinking/

# CT at School

Year 1: Understand Basic Concept of CT

Form 1: Apply the CT concept to solve simple problem

Form 4: Apply the CT concept in solving programmin g task

(#MydigitalMaker)

## Examples: Integrating CT in School Curriculum

General

KS1 : Understand the concept of algorithms

KS2: Apply Logical Reasoning in solve problem

KS3: Evaluation CT Technology

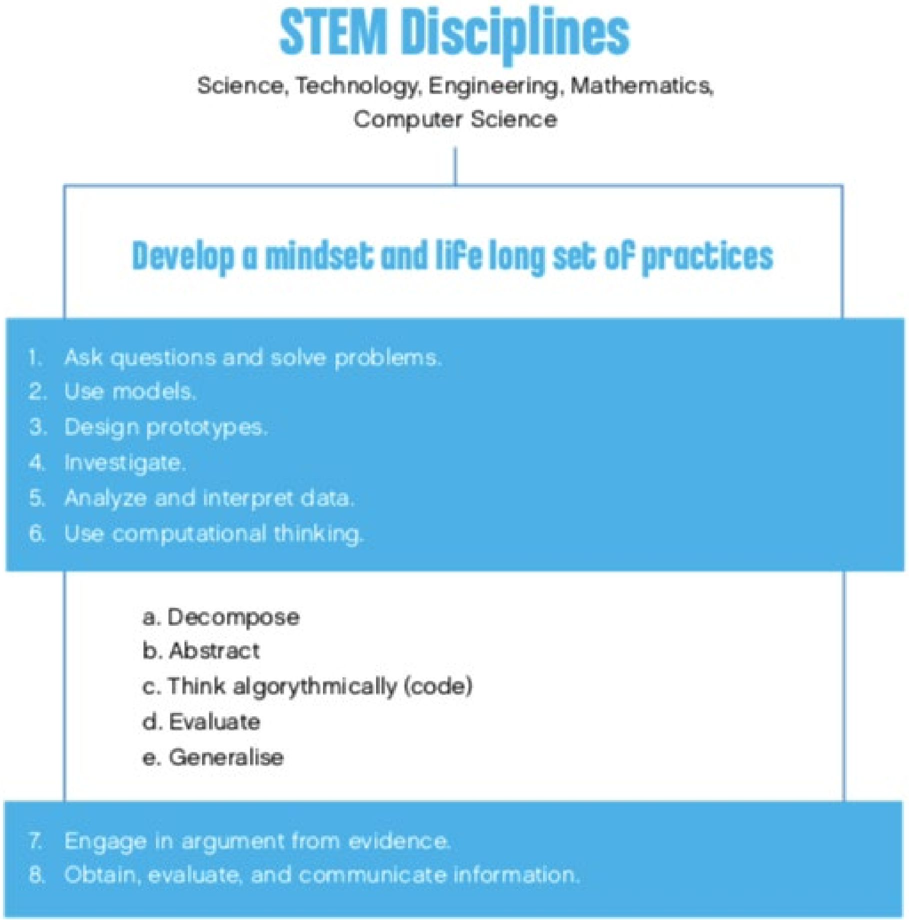
KS4: capability, creativity and knowledge Development

(Schools, National Curriculum England)

## Example: Integrating CT as Extracurricular Activity

Developing Computational Thinking with WeDo 2.0 Projects

In this chapter, you will discover how you can use WeDo 2.0 to develop computational thinking skills in a science context.


STEM Disciplines

Science, Technology, Engineering, Mathematics, Computer Science

Develop a mindset and life long set of practices

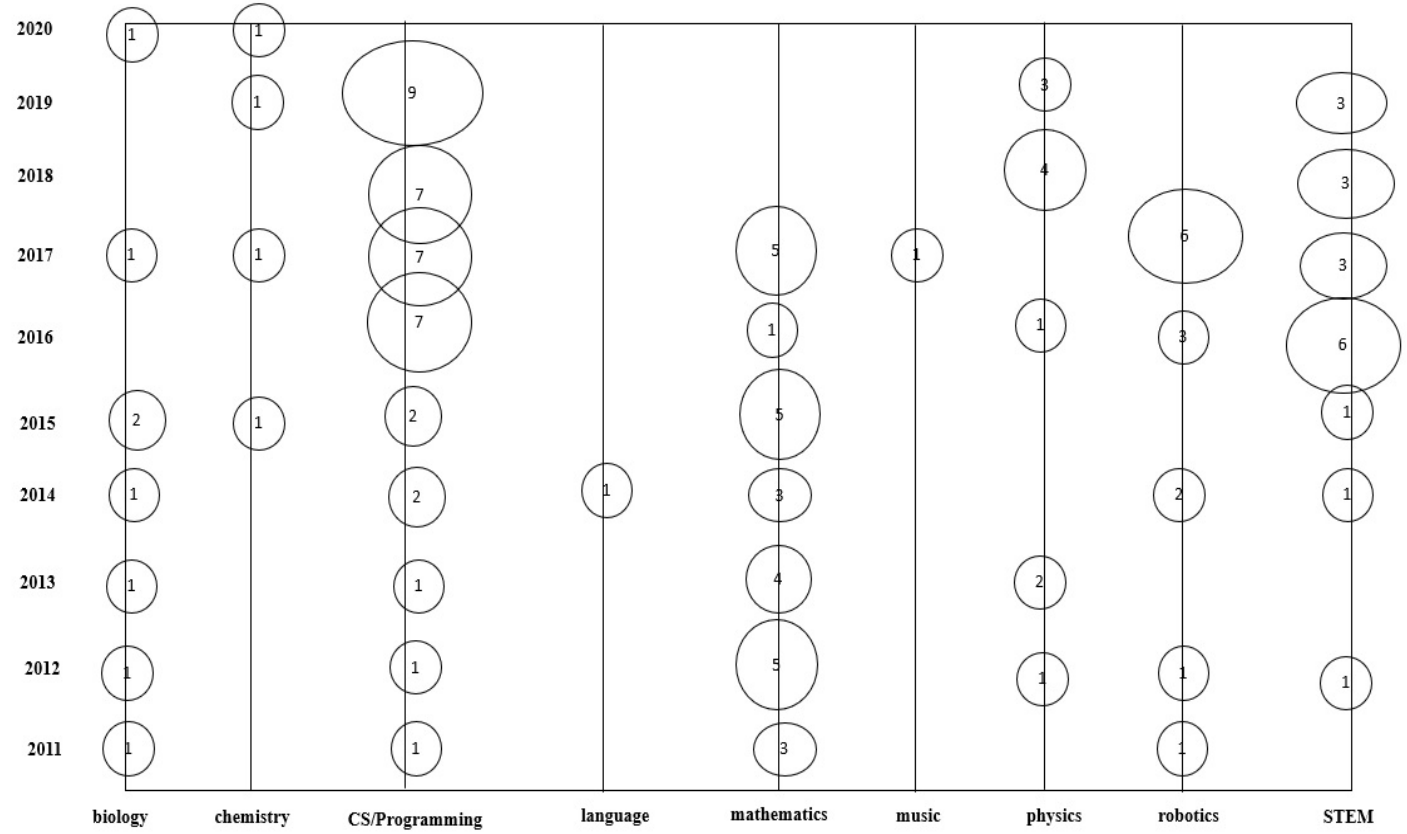
1. Ask questions and solve problems.
2. Use models.
3. Design prototypes.
4. Investigate.
5. Analyze and interpret data.
6. Use computational thinking.

a. Decompose
b. Abstract
c. Think algorythmically (code)
d. Evaluate
e. Generalise

7. Engage in argument from evidence.
8. Obtain, evaluate, and communicate information.

# CT in variety of subjects

- CT not just applicable in computer & technology field but in a range of subject areas such as art, mathematics and many more (Barr & Stephenson, 2011).



Bubble Map for CT Papers Based on Subjects

1 Math

(Snodgrass et al., 2016) combined Mathematics and scratch to show pupils how to measure time

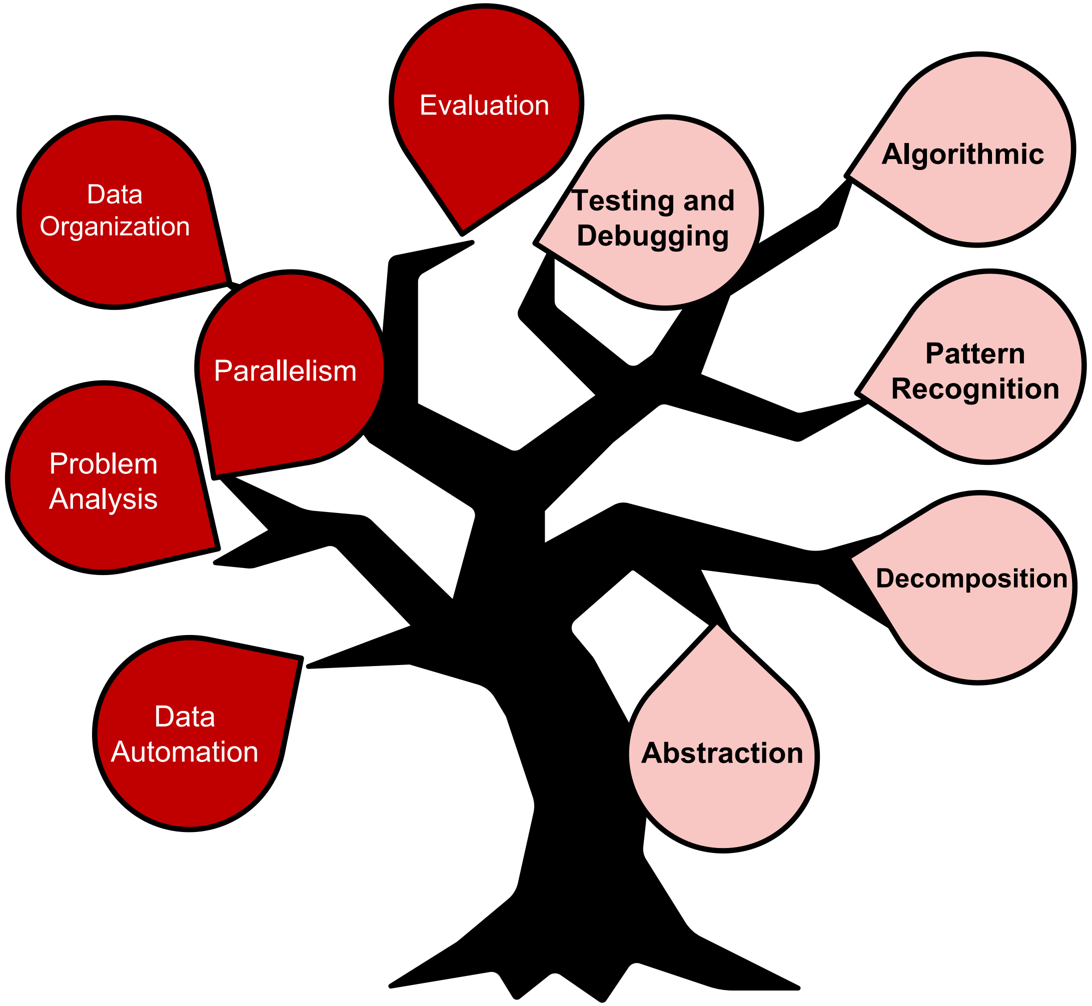
2 Biology

(Libeskind-Hadas and Bush, 2013) proposed a Bi
Comp prototype to help students solve problems
biology through CT and logic programming

3 English

(Homer et al., 2014) used a digital storybook with integrated games to facilitate the development of L1 English literacy

# CT Core Concepts



Some CT core concepts

- a strategy for **solving problem** in computing

# Literature Review CT Core Concepts

| | Kao and Hsu (2019) | Gonzalez and Murioz-Repiso (2018) | Romero et al (2017) | Duncan et al (2017) | Kalelioglu et al (2016) | Angeli et al (2016) | Repenning et al (2016) | Bers et al (2014) | Grover and Pea (2013) | Seiter and Foreman (2013) | Selby and Wollard (2013) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Abstraction | • | | • | • | • | • | • | | • | • | • |
| Decomposition | • | | • | • | • | • | | • | • | • | • |
| Algorithm | • | • | • | • | • | • | | • | • | • | • |
| Pattern | • | • | | | | | | • | • | | |
| Logical Reasoning | | | | • | | | | | • | | |
| Evaluation | | | • | • | | | | | | | • |
| Simulation | • | | | | | | | | | | |
| Problem Analysis | | | | | | | | | | | |
| Parralelisation | | | | | | | | | | | |
| Testing & Debugging | | | | | | | | | | | |
| Control Structure | | | | | | | | | | | |
| Data | | | | | | | | | | | |
| Automation | | | | | | | | | | | |
| Generalization | | | | | | | | | | | |
| | **SS** | **K** | **HE** | **HE** | **HE** | **SS** | **HE** | **K** | **HE** | **HE** | **HE** |

- There are concepts that suitable for teaching higher education and there are also concepts can be learned by all level including pre-school/kindergarten level, primary and secondary school level.
- The age factor of the learners which give impact on the ability to think in solving problem (Atmatzidao and Demetriadis, 2016).

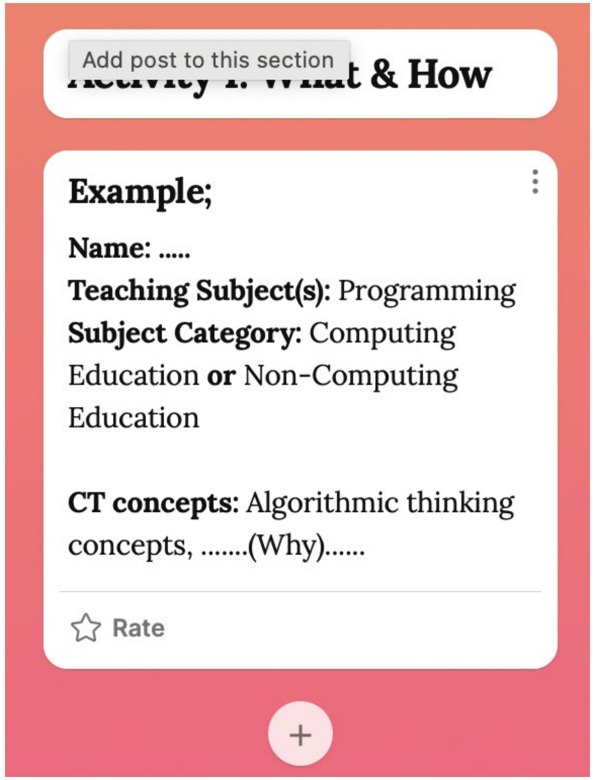
HE – Higher Education, SS – Secondary School, PS – Primary School K – Kindergarten

# Activity 1

What are the CT concepts can be presented in your course?

Upload your answer through Padlet;
Link :
https://padlet.com/nnazihah27/7ows2u306xu0hzan

QR Code:



Activity 1: What & How

**Example;**

**Name:** .....
**Teaching Subject(s):** Programming
**Subject Category:** Computing Education **or** Non-Computing Education

**CT concepts:** Algorithmic thinking concepts, .......(Why)......