

# WHAT EVERY CHILD SHOULD LEARN

Simon Peyton Jones

Computing at School Working Group

May 2013

The thinking behind  
the major policy  
changes in the UK's  
education in  
computing,  
for ages 6-18

# What every child should learn

At school, we teach both **disciplines** and **technologies & skills**

## Disciplines

- Principles, ideas
- Knowledge, laws
- Techniques, methods
- Broadly applicable
- Dates slowly

Physics, chemistry,  
mathematics, English

## Technologies & skills

- Artefacts
- Machines
- Programs
- Products
- Organisations
- Business processes
- Dates quickly

Budgeting, presentation  
skills, metalwork, textiles

# But in computing...

Discipline

Technologies and skills

- Principles
- Ideas
- Laws
- Broadly applicable
- But needs application
- Dates slowly

- Spreadsheets
- Databases
- Powerpoint
- Digital media
- Using the web
- Safety on the internet
- Plan communication projects

# The UK picture

Computer Science  
Barely taught

ICT  
Statutory, dominant

- Spreadsheets
- Databases
- Powerpoint
- Digital media
- Using the web
- Safety on the internet
- Plan communication projects

- Principles
- Ideas
- Laws
- Broadly applicable
- But needs application
- Dates slowly

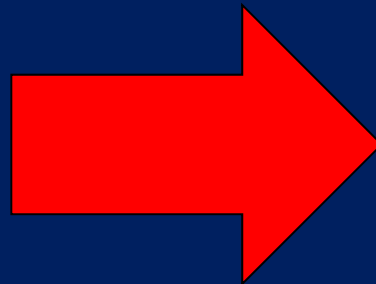
No age-16 qualification at all  
(2009)

Range of 14+ different  
age-16 qualifications

# A radical shift of perception

What most people think

Computer science is a niche university subject for socially-challenged males



The reality

Computer science is a foundational discipline, like maths or physics, that every child should learn, from primary school onwards

What EVERY child should learn  
from primary school (traditional)

Maths

Science

English

Art

What EVERY child should learn  
from primary school (21<sup>st</sup> century)

Maths

Science

Computer Science

English

Art



# What is "Computer Science"?

- **What students should *know*:**  
languages, algorithms, data structures and representation, architecture, programs, communication and coordination.
- **What students should be able to *do*:**  
computational thinking, abstraction, modelling, design, problem solving, programming.

# Computational thinking (Jeannette Wing)

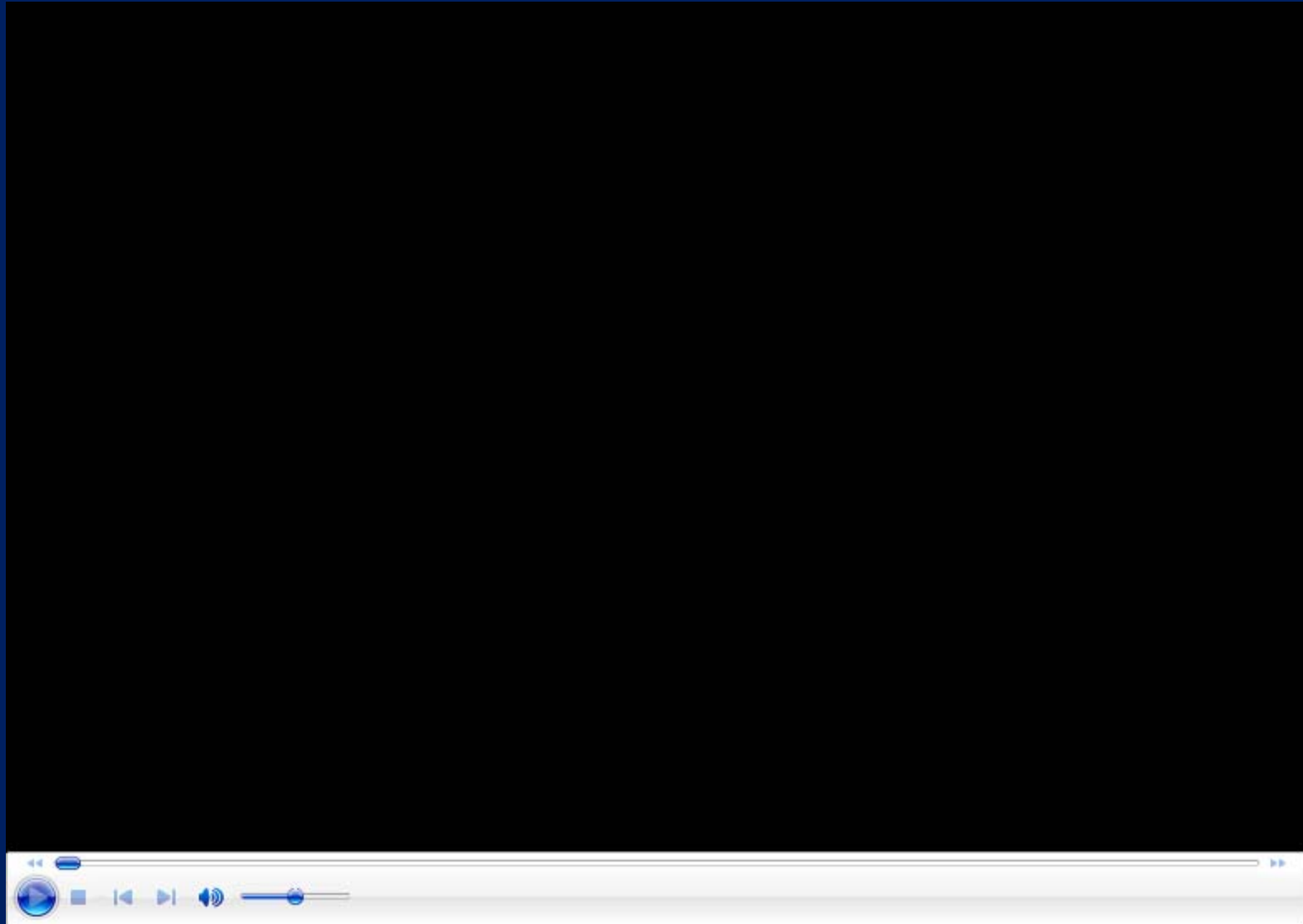
Computational thinking is the process of *recognising* aspects of computation in the world that surrounds us, and *applying* tools and techniques from computing to understand and reason about both natural and artificial systems and processes.

- Computational thinking is something **people** do, not something **computers** do
- Computational thinking is ubiquitous; it is useful in every profession, and in daily life

# What is "Computer Science"?

- **Foundational**
  - Not just "coding/programming" (although that too)
  - Not just to get a good job (although that too)
  - Not just for geeks, or even future software professionals
- The "**fourth science**"
- **Ubiquitous**, like maths: biology, ecology, design, engineering, astronomy, medicine,...
- Primarily rooted in **ideas** rather than **technology** hence using the term "**computer science**" rather than "**information technology**"
- A **quintessentially STEM subject** (involving Science, Technology, Engineering, and Mathematics)

# What is "Computer Science"?



# Why every child?

- We live in a world suffused with digital technology. Ignorance of how that technology works means being a slave to it. The choice is: "Program or be programmed" (Rushkoff).
- Computing teaches unique thinking and problem-solving skills: computational thinking, abstraction, and precision.
- Information and computation gives a new "lens" through which to look at the world: eg flocks of birds, cell biology, cancer propagation, and ecology.

# Why every child?

- We live in a world suffused with digital technology. Ignorance of how that technology works means being a slave to it.

- An elementary knowledge of computer science, including programming, is a fundamental digital literacy
- “lens” through which to look at the world: eg flocks of birds, cell biology, cancer propagation, and ecology.

# Slam dunk

1. Computer science is **educationally foundational**
2. Computer science equips students to meet the **huge un-met demand** from employers.
3. Computer science is **tremendous fun**: creativity, intellectual beauty, programming, robots, making things do stuff.

# The key idea

- Computer Science should be **recognised in school as a rigorous subject discipline**, like physics or history, alongside the (useful) skills of applying information technology.
- Just as every student needs to learn a bit of chemistry, even though few will become chemists, so **every student** should learn a bit of computer science (including some elementary programming) because they live in a digital world.
- From **primary school onwards** (like science).
- My true goal: to **INSPIRE** young people with the joy and beauty of computational thinking