National Centre for Computing Education

The revolution in computing at school: opportunity and challenge

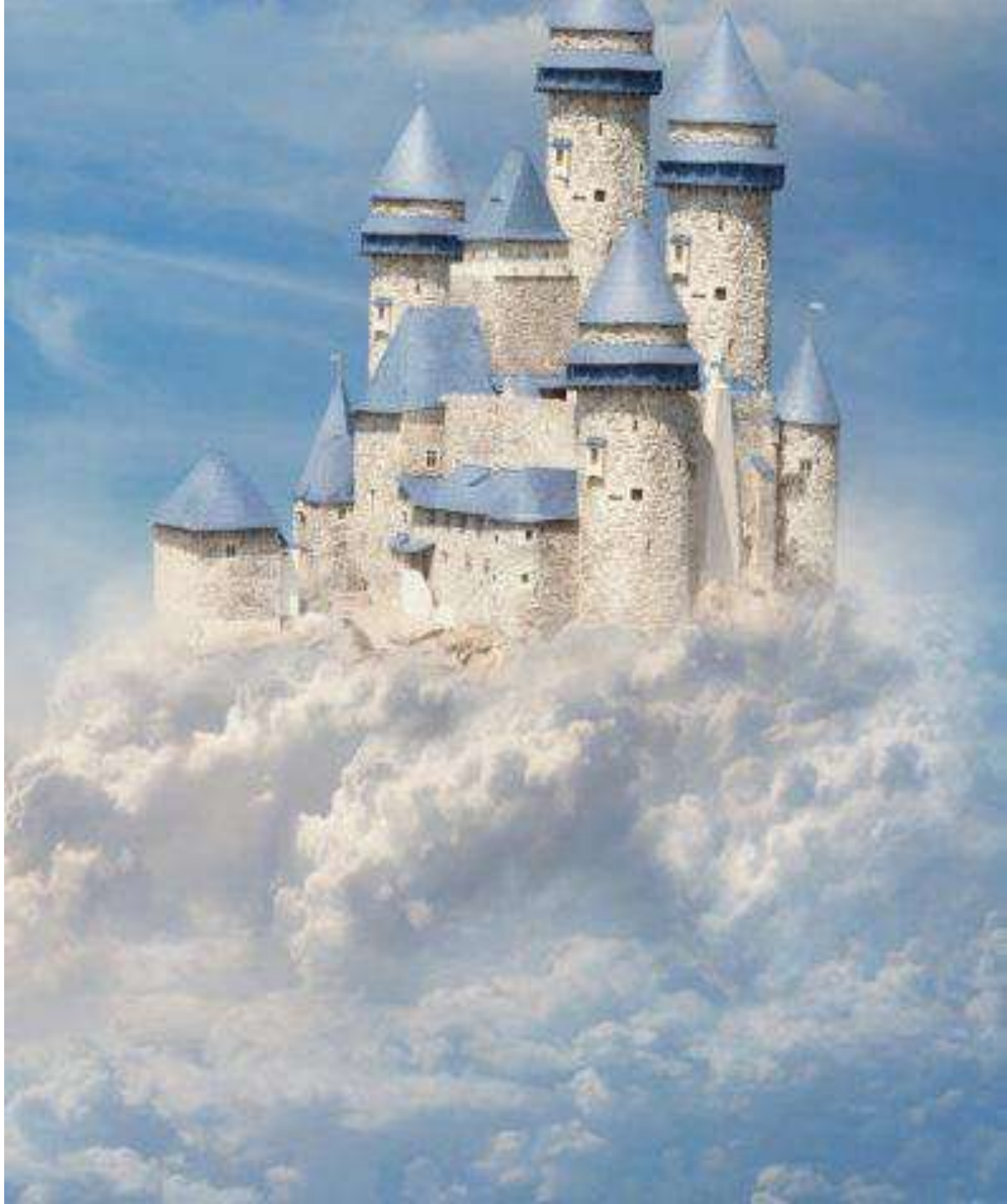
Simon Peyton Jones
Chair, National Centre for Computing Education

What do I truly care about?

Who loves their work?

“”The programmer, like the poet, works only slightly removed from pure thought-stuff. He builds his castles in the air, from air, creating by exertion of the imagination. Few media of creation are so flexible, so easy to polish and rework, so readily capable of realizing grand conceptual structures.”

Fred Brooks



Putting two and two together

"OUR VISION FOR TEACH COMPUTING"

Vision

Computer science is a foundational subject discipline, like maths and natural science, that every child should learn from primary school onwards

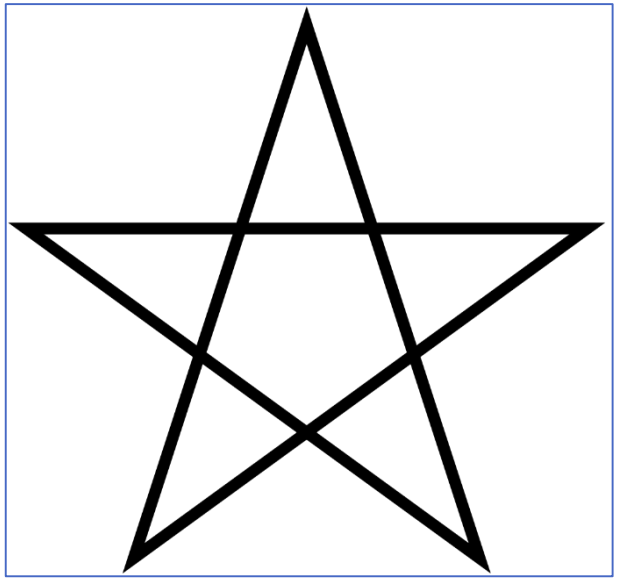
What is computer science?

What is computer science?

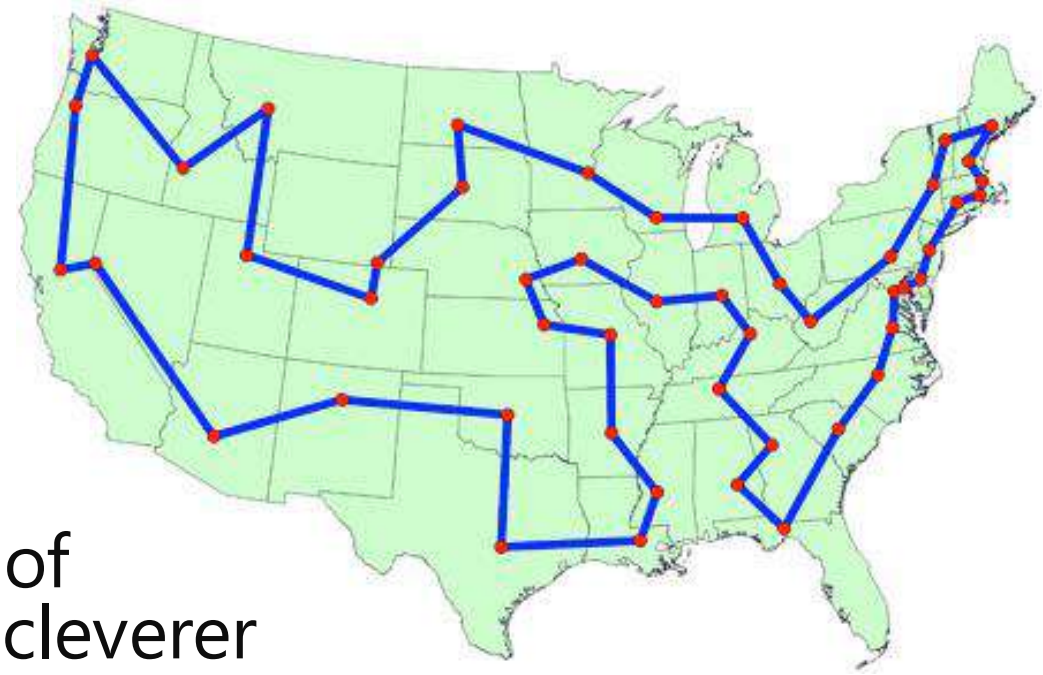
Computer science is the study of
information,
computation, and
communication,
in both natural and artificial systems

Information

- How much information is there in these two pictures?
- If I transmitted the picture to you, and the message was corrupted a bit en route, how could you know? How could you recover the original?
- Is information a measurable quantity? What would its units be?

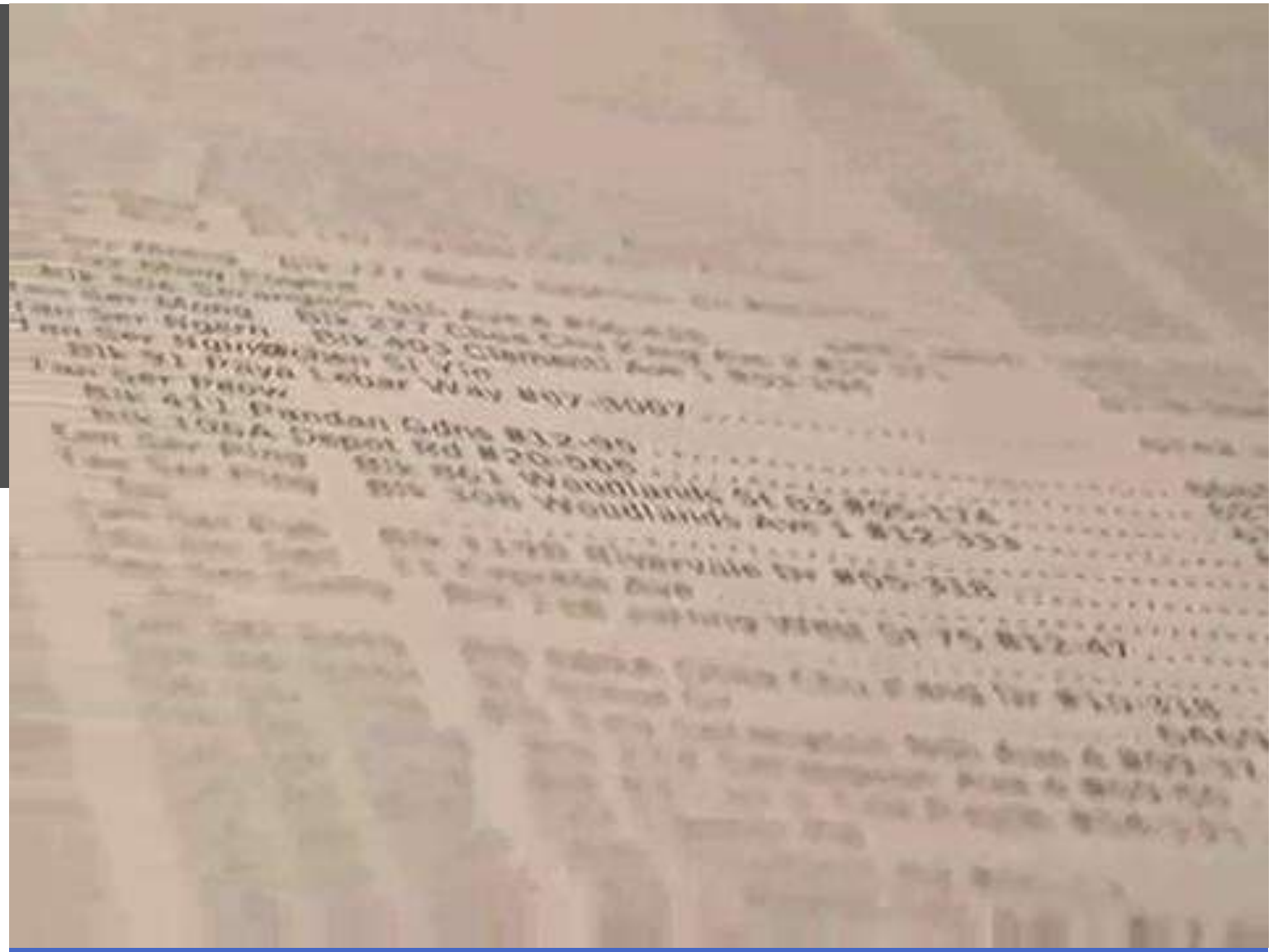


Computation



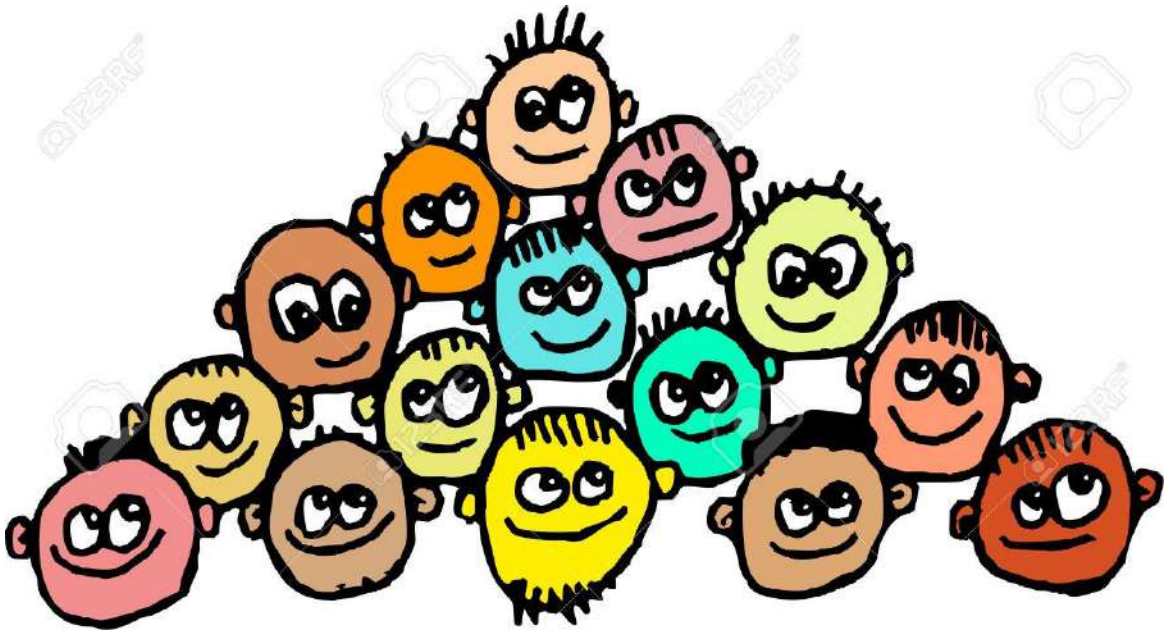
- Find the shortest route through all the red cities.
- Worst case: longer than the lifetime of the universe. (But could there be a cleverer algorithm?)
- Simulated annealing: a simple but clever algorithm that runs in seconds, and gives a result that is almost always very close to the best answer. (What does “almost always” mean? How close?)
- Simulated annealing is useful in lots of applications; it’s a re-usable idea

Look!
No computers



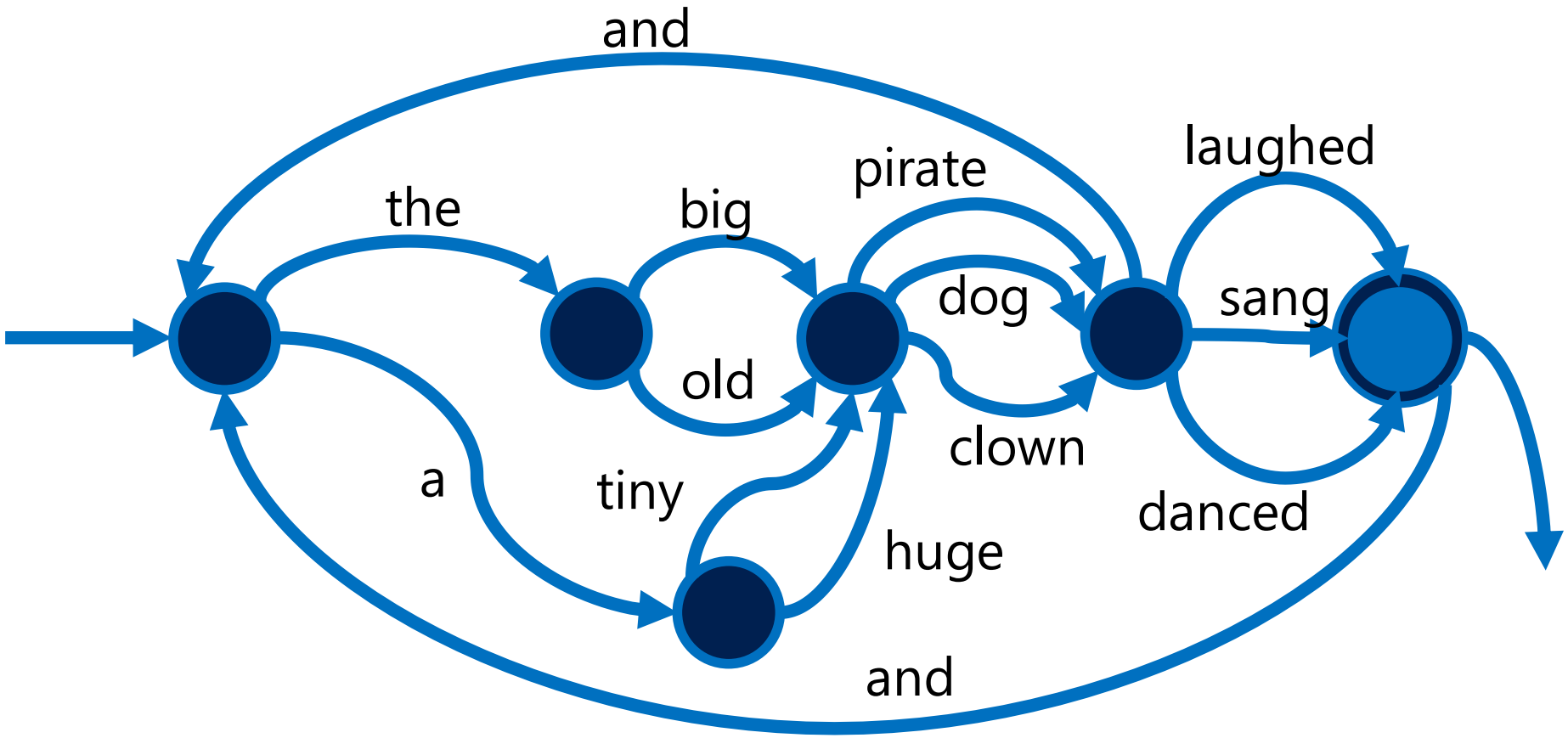
http://csunplugged.org/sorting-networks

Communication



- Bill and Jane have never met before.
- Can they have a 100% private conversation despite eavesdroppers hearing 100% of what they say?

Interdisciplinary



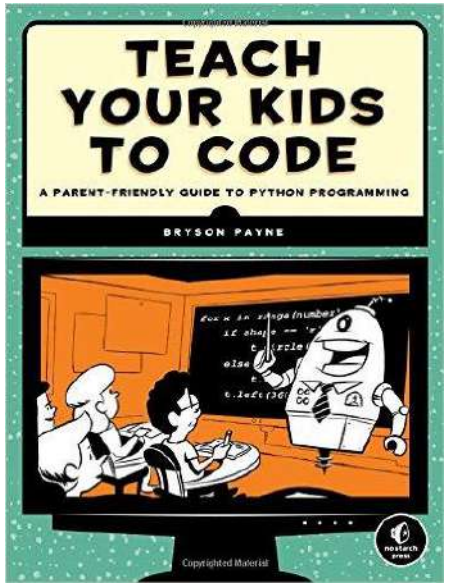
Follow the arrows to generate a sentence

Myth 1: it's all about computers

> Computer science is no more about computers than astronomy is about telescopes
>
> EJW Dijkstra

- Ideas, not technology
- "Unplugged" approaches to teaching
- Beyond technology: computation in the natural world

Myth 2: it's all about coding



Why Our Kids Must Learn to Code

Computer science and IT The Observer

Why all our kids should be taught how to code

15 Reasons Why We Should Be Teaching Our Kids To Code

BY JAYNE CLARE · APRIL 20, 2013 · BLOG · 13 COMMENTS

Myth 2: it's all about coding

- **Programming** is to **computer science**
  as
  Labwork is to **physics**

- Crucial, motivating, and "ground truth"
- But also seductive, distracting, and risks excessive focus on technology details

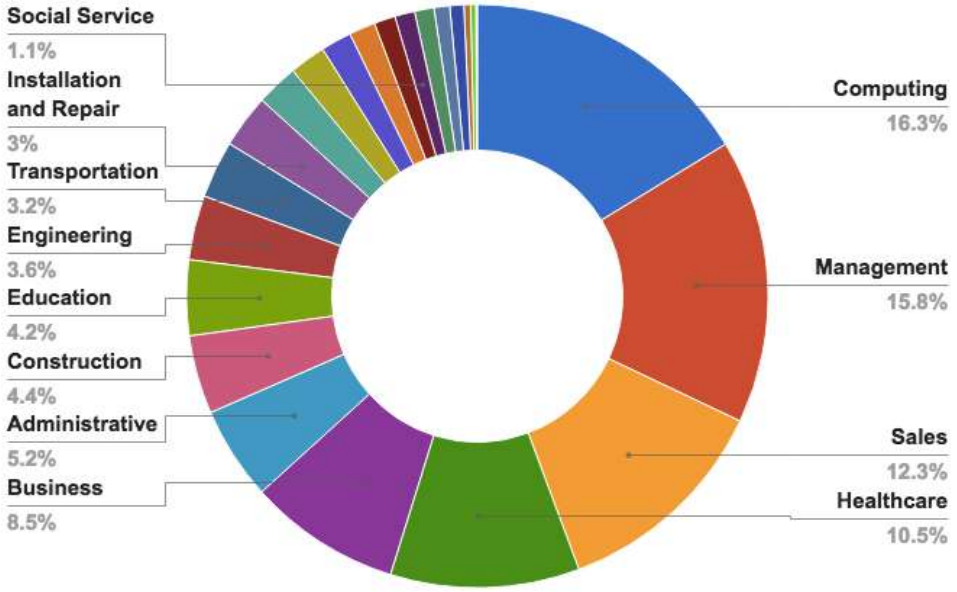
- A bad outcome: "we teach all our kids Python"

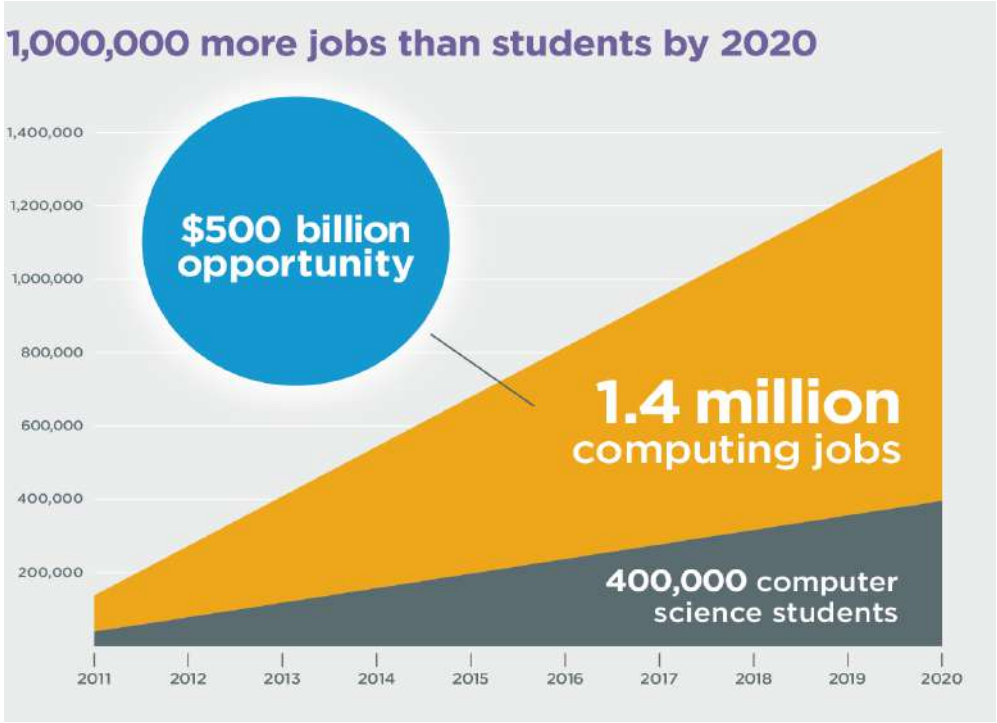
Myth 3: it's all about jobs

- It's about **understanding the world**, and being equipped for life, not just about jobs
- It's about **tomorrow's jobs**, not today's
- It's about **all jobs**, in design, engineering, retail, banking, plumbing, advertising, cookery, management — not just hard-core devs

Jobs in computing

- 70% of new job growth in STEM will be in computing:
  the STEM crisis is a computing crisis
- It's not just STEM! Almost half the graduates who use advanced CS skills (notably programming) are in non-STEM fields.
- There just aren't enough graduates to fill the need
- These are the best-paying jobs. What's not to like?





Myth 4: computer science is not for creative people

> *"The programmer, like the poet, works only slightly removed from pure thought-stuff. He builds his castles in the air, from air, creating by exertion of the imagination. Few media of creation are so flexible, so easy to polish and rework, so readily capable of realizing grand conceptual structures."* *Fred Brooks*

- Every program is a new thing, something no one has done before
- There are no limits – except the limits of our own ability to manage complexity
- Managing that complexity is inextricably tied to conceptual clarity, elegance, and modularity

The big vision

Computer science as a school subject

A foundational subject discipline that every child should learn

"Education should prepare young people for jobs that do not yet exist, using technologies that have not yet been invented, to solve problems of which we are not yet aware."

Richard Riley

A subject that will equip our children for the jobs of the future

Educational pillar

Instrumental pillar

How

can we make CS for all into a reality?

The UK journey

Eric Schmidt (Google) tells us we should be educating our children in CS!

Shut Down or Restart

CAS curriculum (35 pages)

COMPUTING AT SCHOOL
EDUCATE · ENGAGE · ENCOURAGE
In collaboration with BCS, The Chartered Institute for IT

CAS is formed

New curriculum (2 pages) published

New curriculum launches

£

NETWORK OF EXCELLENCE COMPUTER SCIENCE TEACHING

After the Reboot

National Centre for Computing Education

£££

2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019

Review of the National Curriculum in England

Department for Education

Computing

Age 6-16

Launched Sept 2014

Programmes of study for Key Stages 1-4

Aims

The National Curriculum for computing aims to ensure that all pupils:

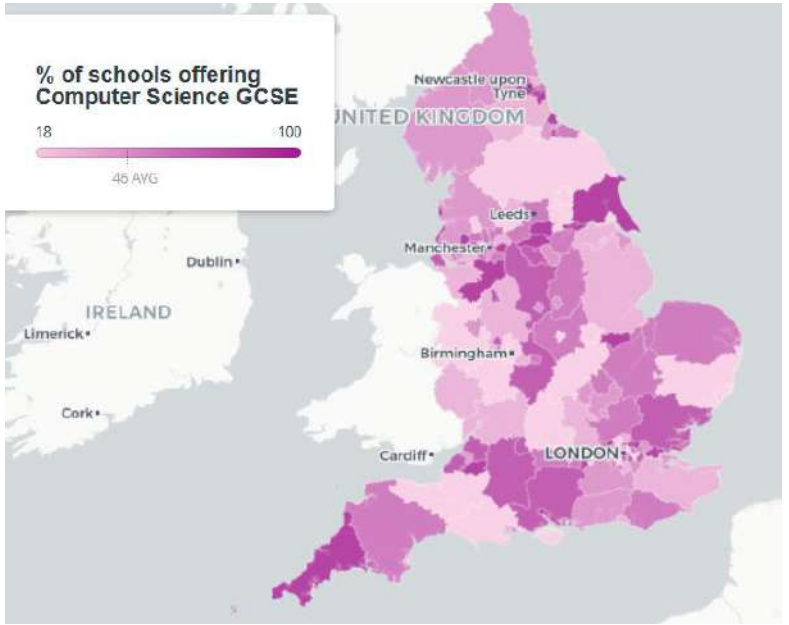
- can understand and apply the fundamental principles of computer science, including logic, algorithms, data representation, and communication
- can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems
- can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems
- are responsible, competent, confident and creative users of information and communication technology.

Myth 4: it's all about the curriculum

- Yes, you need a good curriculum
- But, turning it into a living reality, in every classroom, is a huge, huge challenge
  - Seismic change: establishing an entirely new subject discipline at school
  - Few teachers have computing qualifications
  - Very limited experience of pedagogy and assessment
  - Schools under intense pressure

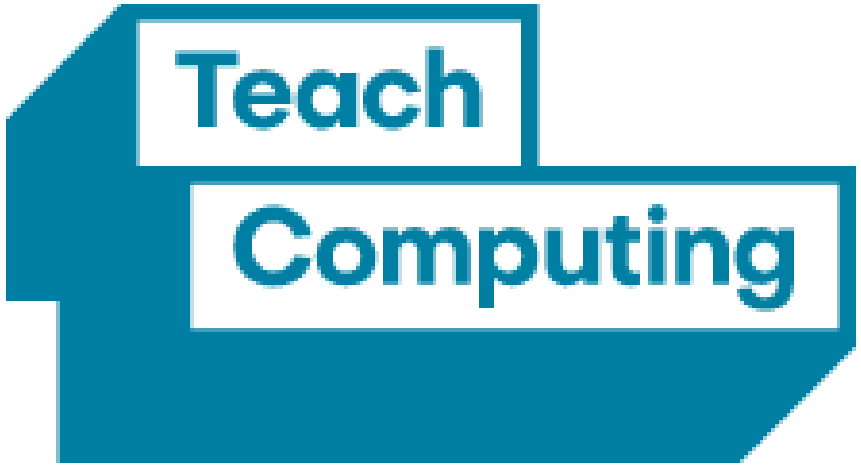
November 2017
(3 years after the launch of the new curriculum)

- Much to celebrate. Many more children are learning much deeper stuff about computing than before.
- But patchy and fragile
- Only 11% of students take GCSE Computer Science; 54% of schools do not even offer it
- Only 1 in 5 computer science students are female
- Professional development inadequate: 25% of secondary teachers have lacked the opportunity to do any; 41% had done less than nine hours



The National Centre for Computing Education

- Inspire, train, support, equip, encourage teachers to teach the most fascinating, creative, and economically important subject on the planet
- £84m over four years
- From primary through to A level
  - 3,500 secondary schools
  - 16,000 primary schools
  - 6,000,000 children



How the NCCE works

- Network of 40 school-led Computing Hubs
- Face to face and online professional development courses
- Comprehensive online Resource Repository
- Isaac Computer Science platform (A level)
- Computing at School (CAS) community of practice

https://teachcomputing.org/

Job done?

The challenge

- Establishing an entirely new school subject is a Huge Ask
- 20,000 schools is a lot
- £80m over 4 yrs = £1,000 per school per year

- **Being a teacher is hard. Being a computing teacher is harder.**
  - Everything is in flux
  - Making CS a mainstream subject, including from early years, has consequences right across the educational system
  - Pedagogy is in the early stages
  - Qualifications are in constant change
  - Huge shortage of confident teachers with subject knowledge
  - Hard to get out of school for training
  - “Its me, me, or ……. me”

Our teachers

A single inspired, equipped, valued, supported teacher will influence tens or hundreds of children every day, and thousands over their career.

Our job is to put them in a position to be that teacher.

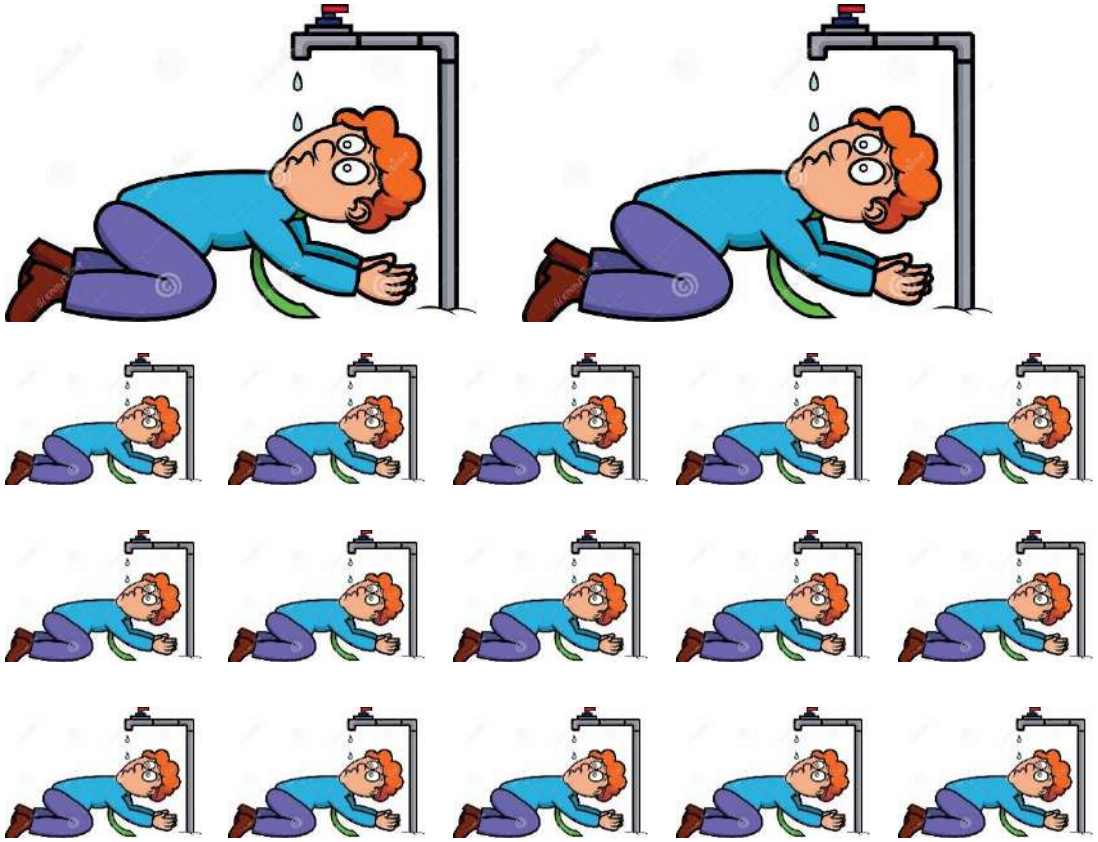
We can only do this **with** teachers, not **to** teachers.

Our employers and employees

Energy Expertise
Goodwill Money

Our schools

Passion Commitment
Expertise Thirst



Meet Terry Bennett

Head Teacher, St Paul's Primary, Whitechapel

Enough talk!

- It's a time of inspiration and opportunity
- It's also a time of anxiety, stress, and fear.
- There is everything to play for;
  but much to do
- We can
  a) wait for someone else to do something
  b) roll up our sleeves and do it ourselves

I vote for (b)

Getting involved: as an individual

- **Find a school** and get alongside the teachers. Form a relationship. Listen to them.

- **Become a STEM Ambassador** (jumping off point for everything else)
- **Become a Barefoot volunteer**
- **Start or join a Code Club or CoderDojo**

https://teachcomputing.org/get-involved
https://teachinglondoncomputing.org/volunteering/

Getting involved: as a group in a company

- We can do more together than we can individually
  - Reliable
  - Sustained over time
- Companies are remarkably supportive, when their own employees are leading
- **But you have to lead!** Don't wait for your corporate CSR outfit.

Getting involved

- Need to be **proactive**. No one will tell you what to do. You have to make the running.
- Need for **sustained input**: parachuting in for a few weeks and disappearing risks doing more harm than good.
  - Individuals: good
  - Groups of individuals: better
  - Groups with corporate blessing/backing: better still

- **Don’t reinvent the wheel.** Join/improve an existing programme rather than invent a shiny new thing.
- **Education is complicated**. We need to be humble, collaborative partners.
- **But do something!**

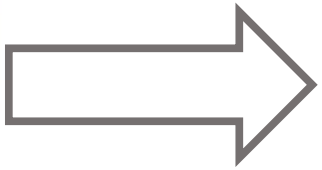
Wendy MacLeod
BCS Community Outreach

Steve Clarke
STEM Learning Education Lead

Computing in action

‘M’ realised she could replace the repeated steps by this for loop

```
function main ()
    turtle → forward(100)
    turtle → right turn(90)
    turtle → forward(100)
    turtle → right turn(90)
    turtle → forward(100)
    turtle → right turn(90)
    turtle → forward(100)
end function
```



```
function main ()
    for 0 ≤ i < 4 do
        turtle → forward(100)
        turtle → right turn(90)
    end for
end function
```

Computing in action

M's schoolteacher:

"Here I have M, self esteem going through the roof and she has associated this computing success with maths.

Over the last couple of weeks, she has solved [maths] problem after problem, met target after target – she is truly flying.

M is going great because of a positive experience in a computing workshop in London."

Engaged, curious

Empowered, informed

Creative, playful

Employed

Engaged, curious

Empowered, informed

Key links

https://teachcomputing.org/get-involved
https://teachinglondoncomputing.org/volunteering/

Creative, playful

Employed