

## Secondary Reading List

The following set of resources is designed to support new teachers of Computing and Computer Science teaching at secondary level.

The Programme of Study for Computing in the National Curriculum should be the starting point and connected with this CAS has produced a number of guides to draw out more detail and practical advice for delivering these programmes of study. Additionally, this list contains some books and websites that typically<sup>1</sup> appear on university reading lists.

### **Computing in the national curriculum: a guide for secondary teachers**

This free guidance has been developed by Computing at School to help secondary school teachers get to grips with the new curriculum for computing. The guide demystifies the programme of study and gives easy to follow support for planning, teaching and assessing computing.

NB. There is also a [primary version](#) which may be of particular interest for teachers of Year 7 looking at transition of pupils from primary to secondary.

### **QuickStart Computing**

QuickStart Computing has been developed to support primary and secondary schools with the computing programme of study and provides all teachers with the resources needed to successfully run computing CPD for colleagues and has applicability for the classroom teacher in getting to grips with teaching the subject to their pupils. It contains the essential subject knowledge, with a framework and guidance for planning, teaching and assessing progress for all pupils.

### **Decoding the new computing programmes of study**

Decoding the new programmes of study for computing is a guide to the thinking behind the new Programme of Study for Computing in England. Written by Simon Peyton Jones (Chair, Computing At School) and designed to help teachers unpack some of the rather dense language in the Programmes of Study.

### **Computational thinking - A Guide for teachers**

Computational thinking lies at the heart of the computing curriculum but it also supports learning and thinking in other areas of the curriculum and this free guide seeks to help develop a shared understanding of the teaching of computational thinking in schools. It presents a conceptual framework of computational thinking, describes pedagogic approaches for teaching and offers guides for assessment. It is complementary to the two CAS guides published in November 2013 (Primary) and

---

<sup>1</sup> This list is not exhaustive. Each university will have their own reading list. This list gives a flavour of what is available.



## Secondary Reading List

June 2014 (Secondary) in supporting the implementation of the new National Curriculum and embraces the [Barefoot Project for Primary](#) and [CAS QuickStart](#) Computing descriptions of computational thinking.

Additional resources for computational thinking can be found [here](#), a particular highlight is the [set of resources made available by Google](#)

### Subject Knowledge

#### Computer Science Field Guide

The “Computer Science Field Guide” is an online interactive resource for high school students learning about computer science, developed at the University of Canterbury in New Zealand. It is an excellent resource for getting coverage of the topics and provides many practical illustrations for teaching the content which can be turned into lesson plan resources.

#### CS Principles: Big Ideas in Programming

In a similar vein to the Computer Science Field Guide is the [Big Ideas in Programming](#) online book. There are teacher and student versions available. This book helps teachers learn how to teach programming without making them write lots of code, and uses Python.

### Teaching Programming

There are a plethora of other sites and other resources for both learning how to program and the teaching of programming including schemes of work and practical exercises. It would be difficult to list them all for all languages used in schools! There are two aspects here. First, learning how to program and, secondly, learning how to teach programming effectively. One recent resource from an experienced teacher, available through CAS is the [Self-marking python activities: try it, debug it extend it](#). A series of differentiated python activities that cater for a wider ability range without frustrating or leaving as many students behind. Each activity focuses on one programming construct or concept and takes pupils through the theory, trying out some example code with challenges to adapt it, fixing any errors they find and then extending the code.

For the pedagogy of programming [the course on PRIMM hosted by FutureLearn](#) is strongly recommended as the PRIMM approach pulls together known effective teaching practices into one, structured framework.

# Secondary Reading List

## Essential Reading

BCS Academy Glossary Working Party. (2016) *BCS Glossary of Computing*. Swindon: BCS

Beecher, K. (2017) *Computational thinking: a beginner's guide to problem-solving and programming*. Swindon: BCS.

Busch, B. and Watson, E. (2019) *The science of learning: 77 studies that every teacher needs to know*. London: Routledge.

Clarke, B. (2017) *Computer science teacher – insight into the computing classroom*. Swindon: BCS.

Harrison, A.J. (2021) *How to Teach Computer Science: Parable, practice and pedagogy*. Woodbridge: John Catt.

Johnson, S. (2021) *100 Ideas for Secondary Teachers: Outstanding Computing Lessons*. London: Routledge.

Kubica, J. (2012) *Computational fairy tales*. Kubica J.

Lau, W. (2019) *The little book of algorithms*, Woking: Nielsen

Sentence, S., Barendsen, E. and Schulte, C. (2018) *Computer science education: perspectives on teaching and learning in school*. London: Bloomsbury.

Simmons, C. and Hawkins, C. (2015) *Teaching computing*. London: Sage.

Textbooks from Awarding bodies such as OCR, AQA, Edexcel or WJEC

Turner-Bisset, R. (2012) *Expert teaching: knowledge and pedagogy to lead the profession*. London: David Fulton.

## Longer reads

Christian, B. and Griffiths, T. (2016) *Algorithms to live by: the computer science of human decisions*. London: Harper Collins.

Lau, W. (2017) *Teaching computing in secondary schools: a practical handbook paperback*. London: Routledge.

Ledeen, K., Lewis, H.R., Abelson, H. and Lewis, H. (2008. 2<sup>nd</sup> edition 2019). *Blown to bits: your life, liberty, and happiness after the digital explosion*. London: Pearson.

MacCormick, J. (2011) *Nine algorithms that changed the future: the ingenious ideas that drive today's computers*. Woodstock: Princeton University Press.



## Secondary Reading List

Papert, S. (1980), *Mindstorms: children, computers and powerful ideas*. London: Basic Books.

### Websites

CAS. (2011) *Computer science - a curriculum for schools*. Cambridge: Computing at School. <<http://www.computingatschool.org.uk/data/uploads/ComputingCurric.pdf>>

Computer Science Education Research Group. (2021) *CSUnplugged*. Available at: <https://csunplugged.org/en/>

Dweck, C. (2012) *Mindset: how you can fulfil your potential*. London: Constable & Robinson.

Jones, K. (ed.) (2020) *Challenging gender stereotypes in education. Learning matters*. London: Sage <<https://us.sagepub.com/en-us/nam/challenging-gender-stereotypes-in-education/book267690>>.

Pollard, A. (2014) *Reflective teaching in school*. London: Bloomsbury <<https://www.bloomsbury.com/uk/reflective-teaching-in-schools-9781441140609/>>.

Raspberry Pi Foundation. (2021) *Hello World Magazine*. Available at: <https://helloworld.raspberrypi.org/>.

*Teach Computing*. (2021) Available at: <https://teachcomputing.org/>

The Reach Foundation. (2021) *Oak Academy*. Available at: <https://www.thenational.academy/>

### Historical books

Papert, S. (1980) *Mindstorms: children, computers and powerful ideas*. Brighton: Harvester

### Reports

Furber, S. (2012) *Shut down or restart? The way forward for computing in UK schools*. London: Royal Society. <<https://royalsociety.org/-/media/education/computing-in-schools/2012-01-12-computing-in-schools.pdf>>

Royal Society. (2017) *After the reboot: computing education in UK schools*. London: Royal Society. <<https://royalsociety.org/~media/policy/projects/computing-education/computing-education-report.pdf>>

Kemp, P., Wong, B., & Berry, M. (2016). *The Roehampton computing education report: Data from 2015*. London: University of Roehampton <[https://www.researchgate.net/publication/311595274\\_The\\_Roehampton\\_Annual\\_Computing\\_Education\\_Report\\_2015\\_data\\_from\\_England](https://www.researchgate.net/publication/311595274_The_Roehampton_Annual_Computing_Education_Report_2015_data_from_England)>

## Secondary Reading List

### Research Articles

Wong, B and Kemp, PEJ. (2017) *Technical boys and creative girls: the career aspirations of digitally skilled youths*

<<https://www.tandfonline.com/doi/full/10.1080/0305764X.2017.1325443> >

Fisher, DH. (2019) *Algorithms in everyday life*, Vanderbilt University

<<https://www.vanderbilt.edu/olli/class-materials/InnovativeThinkingSession1.pdf>>

*This resource has been made possible through contributions from individuals within the CAS community and also universities. Thanks to the following universities for contributing – The University of Southampton, University of Reading, Manchester Metropolitan University.*