

## Non GCSE Action Plan Support **Guides**

### Infrastructure

#### Overview

Computer Science is not just about programming or using computers. Be clear that programming begins with logical thought and solving problems, and that translating algorithms into computer programs is secondary to the difficult work in writing the algorithm/pseudocode in the first place. The sorting out of the logic is done away from the computer and even writing the code can be done away from the keyboard. Additionally, a large proportion of computing (particularly at GCSE) is about learning the theory and facts to develop understanding about the use of technology, which can be done in a traditional non-computing classroom. If technical support is an issue then browser-based solutions e.g. Edublocks, Micro:bit Python, Repl.it and Trinket exist whereby Python coding can be accomplished and shared for assessing and feedback. Consider unplugged activities to develop logical algorithmic thinking. Cheap computers e.g. Raspberry Pi that use limited hardware can be substituted for expensive hardware for most activities and, it might be argued, provide fewer distractions from learning. If money is a barrier see the advice in the *Budget* document.

### Subject Matter Expert Support

#### Suggested Support

- Help plan how many rooms/computers/resources are needed.
- Support in developing curriculum that does not always rely on high quality infrastructure
- Create a relationship with school and other schools who have overcome infrastructure issue possibly through CAS CoP and/or STEM Communities.
- Support in the planning and embedding of suitable software needs
- Support with creative hardware available from hub Kits (Rasp Pi, Microbit etc)
- Support in developing unplugged activities.
- Support in using the TeachComputing Curriculum activities in creative unplugged ways.
- Support in using online environments for programming e.g. repl.it, Edublocks, trinket, micro:bit Python

#### Suggested Actions

- Develop parts of the curriculum that can be taught unplugged or without a computer
- Use the materials in the Teach Computing Curriculum to adapt lessons to avoid the use of computers where possible.
- Isolate the materials in the TCC where you definitely need computers and/or infrastructure for students to make progress.
- Liaise with SLT and technology providers to discuss the provision of infrastructure to support CS curriculum and its requirements.
- Save money and support expertise by using online programming environments through a browser.
- Use alternative hardware e.g. Raspberry Pi computers.
- Setting up a local classroom network with a local router/switch and connected computers.
- Courses and certification can train staff to deliver KS3 curriculum confidently.

### CPD and Professional Learning

#### Courses

##### GCSE

[Supporting GCSE computer science students at grades 1-3](#)

##### Online courses

[Impact of Technology: How To Lead Classroom Discussions](#)  
[Creating an Inclusive Classroom: Approaches to Supporting Learners with SEND in Computing](#)  
[Programming Pedagogy in Secondary Schools:](#)

##### F2F/Remote

[Adapted teaching and effective learning interventions in secondary computing](#)  
[Behaviour for learning in a computing environment - short course](#)  
[Collaboration in KS3 programming](#)  
[Enriching secondary computing with STEM Ambassadors in your region - short course](#)

#### Pathways

Our learning pathways are designed for teachers at different levels and provide a set of recommended courses to help you get started with the

[Computer Science Accelerator programme:](#)

- [New to computing](#)
- [New to algorithms and programming](#)
- [New to computer systems](#)
- [New to GCSE computer science](#)

Resources		Reading	
<b><u>Teach Computing Curriculum</u></b> Everything you need to teach computing at key stages 1 to 4. Resources include lesson plans, slides, activity sheets, homework, and assessments		<a href="#">Assessing Your School ICT Infrastructure</a> Government guidance	
<b><u>Isaac Computer Science</u></b> The free online platform for students and teachers. Use it in the classroom. Use it for homework. Use it for revision.		<a href="#">Technology in education</a> Types of infrastructures a school may have	
<b><u>Pedagogy Resources</u></b> Effective pedagogy is at the heart of good teaching and learning; successful computing teachers combine their knowledge of the subject with evidence-based teaching practices.			
Wider offer - STEM Learning			
<b><u>STEM Ambassadors</u></b>		<b><u>STEM Community</u></b>	
A STEM Ambassador is a voluntary role undertaken by someone who has a passion and/or professional knowledge of STEM subjects.		A community of over 20,000 teachers; technicians; TA's and governors, all involved in the STEM education of young people supporting and collaborating with each other to improve outcomes	
<b>How it could work with this challenge</b> Perhaps STEM ambassadors could support both an offline curriculum and how much computer science occurs away from a computer.		<b>How it could work with this challenge</b> Discuss with other teachers how they overcome infrastructure problems,	