

Assessment companion – England

Overview

As Apps for Good is a flexible framework, it is difficult to provide specific, prescriptive guidance on how to formally assess each topic within the course. However, in this document, we have provided links between the Key Stage 2 and Key Stage 3 Computing Curriculum and each module and mapped the relevant Computing at Schools Progression Pathways to help support summative assessment. We have also provided a series of questions for each topic in order to facilitate formative assessment.

For more information on the Computing Curriculum and CAS Progression Pathways see Assessment Guidance for England 1415

Summative assessment

Table 1 (Key Stage 2) and Table 2 (Key Stage 3) below outline which areas of the curriculum are covered in which module and links this to corresponding CAS Progression Pathways (see Assessment Guidance for England for further details) to enable you to place students in a band for summative assessment purposes.

While we recommend the use of the Progression Pathways to guide your assessment, it is important not to let it drive your approach and delivery (try to avoid instructing the students to do an activity for the sake of covering the curriculum.) The focus should be on their app development and the related learning. When following the Apps for Good modules, the table can be used to help to gauge which areas of the curriculum have been covered.

Formative assessment

Opportunities for formative assessment run throughout the Apps for Good course. In the “How it works” documents you will find a series of questions for each topic which have been designed to gauge students’ understanding of the subject matter. These questions can be used for teacher-led, self or peer assessment.

Table 1: Key Stage 2 National Curriculum for Computing mapped to AfG modules and CAS Progression Pathways

Module	Relevant National Curriculum for Computing statements	Progression Pathways (Blue)	Progression Pathways (Purple)
One – Crash course	<p>Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs (NC2c)</p> <p>Understand computer networks, including the internet; how they can provide multiple services, such as the World Wide Web, and the opportunities they offer for communication and collaboration (NC2d)</p>	<p>Use logical reasoning to detect and correct errors in programs (CS)</p> <p>Understand how computer networks can provide multiple services, such as the world wide web (CS)</p> <p>Understand the opportunities computer networks offer for communication (DL)</p>	<p>Use logical reasoning to explain how some simple algorithms work (CS)</p> <p>Use logical reasoning to detect and correct errors in algorithms (CS)</p> <p>Understand computer networks including the internet (CS)</p>
Two – Idea generation	<p>Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content (NC2e)</p> <p>Use sequence, selection, and repetition in programs; work with variables and various forms of input and output (NC2b)</p>	<p>Appreciate how search results are selected (CS)</p> <p>Use repetition in programs (CS)</p>	<p>Be discerning in evaluating digital content (DL)</p> <p>Be discerning in evaluating digital content (CS)</p> <p>Use selection in programs (CS)</p> <p>Work with variables (CS)</p>
Three – Scoping	<p>NC2e and NC2b and their corresponding progression pathways can be further developed here</p> <p>Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information (NC2f)</p>	<p>Analyse information (IT)</p> <p>Evaluate information (IT)</p> <p>Collect data (IT)</p> <p>Present data (IT)</p>	<p>Combine a variety of software to accomplish given goals (IT)</p> <p>Select use and combine software on a range of digital devices (IT)</p> <p>Analyse data (IT)</p> <p>Evaluate data (IT)</p>

<p>Four – Product development</p>	<p>Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them (NC2a)</p>	<p>Design programs that accomplish specific goals (CS) Design and create programs (CS) Debug programs that accomplish specific goals (CS) Control or simulate physical systems (NC2a)</p>	<p>Solve problems by decomposing them into smaller parts (CS) Design and create systems (IT)</p>
<p>Five – Pitch</p>	<p>Use this module and beyond to supplement areas not covered by student teams over the course of developing their app.</p>		

Table 2: Key Stage 3 National Curriculum for Computing mapped to AfG modules and CAS Progression Pathways

Module	Relevant National Curriculum for Computing statements	Progression Pathways (Red)	Progression Pathways (Black)	Progression Pathways (White)
One – Crash course	Understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem (NC3b)		Understand several key algorithms that reflect computational thinking (CS)	<i>Use logical reasoning to compare the utility of alternative algorithms for the same problem (CS)</i>
Two – Idea generation	NC3b and its corresponding progression pathways can be further developed here Undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users (NC3g) Understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming (NC3d)	Undertake creative projects with challenging goals (IT) Use multiple applications (IT) [Work with] applications across a range of devices (IT) Collect data (IT) Understand simple Boolean logic (CS)	Combine multiple applications to achieve challenging goals (IT) Analyse data (IT) Meet the needs of known users (IT)	<i>Select multiple applications to achieve challenging goals (IT)</i>
Three - Scoping	NC3g, NC3d and their corresponding progression pathways can be further developed here Use 2 or more programming languages, at least one	Use a programming language to solve computational problems (CS) Reuse digital artefacts for a	Use at least one additional programming language (that must be textual) to solve real world problems (CS)	<i>Develop modular programs that use procedures or functions (CS)</i> <i>Repurpose digital artefacts</i>

	<p>of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions (NC3c)</p> <p>Create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design and usability (NC3h)</p>	<p>given audience (DL)</p> <p>Attend to usability of digital artefacts (DL)</p>	<p>Make use of appropriate data structures (CS)</p> <p>Design modular programs that use procedures or functions (CS)</p> <p>Revise digital artefacts for a given audience (DL)</p> <p>Attend to trustworthiness of digital artefacts (DL)</p>	<p>for a given audience (DL)</p> <p>Attend to design of digital artefacts (DL)</p> <p>Create digital artefacts for a given audience (IT)</p>
Four – Product development	<p>NC3c and NC3h and their corresponding progression pathways can be further developed here</p> <p>Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems (NC3a)</p>	<p>Use computational abstractions (CS)</p> <p>Model state of real world problems (CS)</p>	<p>Evaluate computational abstractions (CS)</p> <p>Model state of physical systems (CS)</p> <p>Model behaviour of real world problems (CS)</p>	<p>Design computational abstractions (CS)</p> <p>Model behaviour of physical systems (CS)</p>
Five – Pitch	<p>Use this module and beyond to supplement areas not covered by student teams over the course of developing their app.</p>			