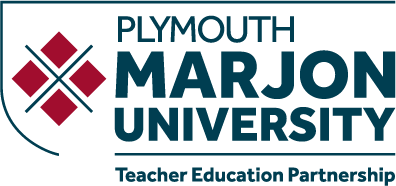
****

**Subject Knowledge Self-Assessment:**

**Computer Science**

|  |  |
| --- | --- |
| Name: | Route: |
| Host school: | Complementary school: |

The self-assessment requires you to reflect upon your own personal strengths knowledge and skills of using the range and content then on your experience of enabling pupils to grasp these key elements. You will be asked to self-assess your knowledge and skills and experience using the following grading:

|  |  |  |  |
| --- | --- | --- | --- |
| 1 = Significant area of strength | 2 = A firm grasp of this aspect | 3 = A developing understanding | 4 = limited grasp of this area |

In the comments section please give a brief justification of why you have graded yourself as you have.

At the end of each term (PP 1, 2 and 4) discuss with your subject mentor the development of your subject knowledge and pedagogy and re-grade and add additional justification.

# Introduction

## Computing interview subject knowledge audit

This audit aims to do several things:

* to help you know what subject knowledge means for your subject
* to help you identify your strengths and areas for development in subject knowledge
* to enable you to bridge the gaps in your subject knowledge before you start the course
* to give your course tutor and school mentors a starting point from which to monitor your progress in developing subject knowledge and guide you in developing your knowledge
* to play a part in helping you to provide evidence of your achievement against the Profile Characteristics that relate to subject knowledge.

## What is subject knowledge?

There are different ways to think about subject knowledge in the context of teacher education. For the purpose of this audit, the focus is mainly on the information, concepts, processes and skills in the subject itself. While we use software to teach some of these aspects, we do not expect to cover simple software skills which you can learn yourself once you have the underlying knowledge and understanding.

### The need to know

No teacher knows everything about their subject. There is no defined list of topics or other information that must be known by subject practitioners. But it is important that you have the fundamental concepts, as set out in this audit, in place. You will certainly discover that one of the best ways to increase subject knowledge is to teach it to someone else. This has become increasingly important in the light of the recent changes to Computing and the need for a whole new range of subject knowledge.

## Subject knowledge in Computing

Subject knowledge in the National Curriculum Programme of Study is not in the form of specific factual content as is found in most other subjects. It is rather more related to skills, knowledge and understanding of concepts. Clearly programming/coding has become important in the last few years but again, you will experience coding but will need to be able to develop the ability to transfer this knowledge to new programming languages.

## Improving my subject knowledge

Although we will cover as much Computing as is possible in the one year PGCE course, we also have to address how to teach the subject. You need to appreciate that you will take responsibility for learning new material and raising your level of subject knowledge or software skills, especially when working in schools on placements. Your subject tutors, mentors and other Bath Spa Teachers can help with resources and in other ways, but this is an aspect of the course in which you need to develop independent strategies for learning.

# Subject Knowledge audit for Computing

## Levels of subject knowledge

### The four levels

The audit identifies four levels of subject knowledge. These range from Level 4 as the lowest level, though to Level 1 as the highest level. The exact definitions of these levels depend to some extent on the curriculum needs of different subjects.

**Enter your level of knowledge at each of the suggested instances during the course using Levels 1-4 (see below).** Use the final column for notes to describe the origin of your knowledge, for example, from a degree module, work experience, AS/A2 level, independent study etc., to describe actions you take to improve your knowledge and to refer to where evidence of your achievement can be found. You can add anything that is not addressed in the audit to the most relevant page although the last page allows you to add any valuable subject knowledge not addressed anywhere else in the audit.

Level 1: your skills, knowledge and understanding is current and to a standard that would make you feel secure to teach this area, with some revision.

Level 2: your skills, knowledge and understanding is good and you would be prepared to teach this area with some further development.

Level 3: your skills, knowledge and understanding is developing but you do not yet feel confident to teach this area.

Level 4: you have never studied the topic or your knowledge is not sufficient to teach it to any standard.

## Entries in your audit

You need to update this audit at intervals during the course.

* At interview
* At the very start of the year
* At the end of term 1 (PP1) December
* At the end of the term 2 (PP2) in December
* At PP3 in May
* At the end of the course PP4 and submit this final version as part of your Indicative evidence for your Professional Conversation.

**COMPUTING SUBJECT KNOWLEDGE AND SOFTWARE SKILLS**

| **Elements of Subject knowledge** | **At interview** | **Initial Tutorial** | **PRP1** | **PRP2** | **PRP 3** | **Final Tutorial** | **Dates as above for the columns**  **Specific examples of experience in this area** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Give a simple description of what an algorithm is |  |  |  |  |  |  |  |
| Write algorithms for everyday task such as making a cup of tea, getting dressed |  |  |  |  |  |  |  |
| Explain why computers need precise instructions. |  |  |  |  |  |  |  |
| Write algorithms containing repetition (loops), selection (if) and procedures (sub algorithms) |  |  |  |  |  |  |  |
| Understand why sub algorithms can deduce amount of coding/file size |  |  |  |  |  |  |  |
| Understand the link between and differences between algorithms and computer programs. |  |  |  |  |  |  |  |
| Recognise algorithmic code in computer programs |  |  |  |  |  |  |  |

| **CS2 - Programming** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Elements of Subject knowledge** | **At interview** | **Start of course** | **PRP2 (Oct 13)** | **End TP3 (Feb 14)** | **PRP 4 (April 14)** | **PRP 5 (End of course)** | **Specific examples of experience in this area** |
| Understand why computer hardware needs computer software (programs) |  |  |  |  |  |  |  |
| Recognise the similarities and differences between a programming language such as Python and a mark-up language such as HTML |  |  |  |  |  |  |  |
| Recognise and use simple HTML tags such a <b>, <p>, <a href=……..>, <img src=…>. |  |  |  |  |  |  |  |
| Create simple HTML pages using a text editor that contain text, images and hyperlinks. |  |  |  |  |  |  |  |
| Create a series of linked pages using a text editor. |  |  |  |  |  |  |  |
| Create simple game using a visual program such as Scratch using the concepts of computer programming as game play and game design. |  |  |  |  |  |  |  |
| Use a control language (eg: Logo) to draw simple geometric shapes that incorporate REPEAT. |  |  |  |  |  |  |  |
| Know what computer programming syntax is. |  |  |  |  |  |  |  |
| Create two simple programs (suitable for Key Stage 3) using two different programming languages |  |  |  |  |  |  |  |
| Explain and use programming concepts such as selection, repetition, procedures, variables, and relational operators |  |  |  |  |  |  |  |
| Use programs to simulate environments and to test a hypothesis. |  |  |  |  |  |  |  |
| Analyse strengths and weaknesses in own use of code |  |  |  |  |  |  |  |
| Detect errors in syntax and its meaning and make corrections. |  |  |  |  |  |  |  |
| Be able to describe the stages in software development and refer to one model of system development. |  |  |  |  |  |  |  |
| Know what happens in the testing phase and the types of testing that might be used for different applications. |  |  |  |  |  |  |  |
| **CS3 - Data** | | | | | | | |
| **Elements of Subject knowledge** | **At interview** | **Start of course** | **PRP2 (Oct 13)** | **End TP3 (Feb 14)** | **PRP 4 (April 14)** | **PRP 5 (End of course)** | **Specific examples of experience in this area** |
| Represent numbers in binary form - decipher binary numbers. |  |  |  |  |  |  |  |
| Use a coding system such as ASCII to code/decode symbols |  |  |  |  |  |  |  |
| Name some audio and graphic file types and give some characteristics/uses of them. |  |  |  |  |  |  |  |
| Know that binary can be used to represent both numbers and symbols (text) |  |  |  |  |  |  |  |
| Understand that software controls whether a binary number is interpreted as a number or text. |  |  |  |  |  |  |  |
| Explain some reasons why different formats are available for sound, video and graphics |  |  |  |  |  |  |  |
| Describe possible sources of data errors (human and machine) and how to minimise them |  |  |  |  |  |  |  |
| Know what the Computer Misuse act, Data Protection Act and Copyright legislation are meant to cover/prevent. Be able to explain key aspects of the legislation and describe the possible consequences to those who contravene them. |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **CS4 - Computers and Social informatics** | | | | | | | |
| **Elements of Subject knowledge** | **At interview** | **Start of course** | **PRP2 (Oct 13)** | **End TP3 (Feb 14)** | **PRP 4 (April 14)** | **PRP 5 (End of course)** | **Specific examples of experience**  **in this area** |
| Explain what a computer is |  |  |  |  |  |  |  |
| Describe a range of microprocessor based devices |  |  |  |  |  |  |  |
| Explain and describe the key parts of a computer - CPU, memory, hard disk, input and output devices - and their key roles in a computer system |  |  |  |  |  |  |  |
| Understand the need to specify computer systems for different applications and what might happen if this is not done when acquiring computer systems. |  |  |  |  |  |  |  |
| Understand the role of the operating system and why different ones have developed. |  |  |  |  |  |  |  |
| Trouble shoot simple problems on computer systems - both hardware and software. |  |  |  |  |  |  |  |
| Use a range of system utilities to prevent or to find and fix problems or to increase performance. |  |  |  |  |  |  |  |
| Be able to install hardware (input devices, RAM) and install operating system, utilities software and generic software (for example Office-type applications) |  |  |  |  |  |  |  |
| Know what computer multi-tasking is, why and how to turn off background applications. |  |  |  |  |  |  |  |
| Be aware of Moore’s law and take part in discussion on the future of computer hardware. |  |  |  |  |  |  |  |
| Be able to describe a range of social and ethical issues (positive and negative) raised by computers - including local and global issues. |  |  |  |  |  |  |  |
| Understand the economic and ergonomic needs for good human-computer interface design |  |  |  |  |  |  |  |
| Be able to describe a number of occupations and career pathways that could be taken by those wanting a computer-based career. |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **CS5 - Communication and the internet** | | | | | | | |
| **Elements of Subject knowledge** | **At interview** | **Start of course** | **PRP2 (Oct 13)** | **End TP3 (Feb 14)** | **PRP 4 (April 14)** | **PRP 5 (End of course)** | **Specific examples of experience**  **in this area** |
| Know what the internet and world wide web are and the relationship between the two. |  |  |  |  |  |  |  |
| Create a simple web site incorporating information about the web and using text, video and graphics as well as links to key web pages. |  |  |  |  |  |  |  |
| Know what data packets and protocols are and why they are required. |  |  |  |  |  |  |  |
| Know what a search engine is, what data they hold and how searches can be influenced for commercial gain. |  |  |  |  |  |  |  |
| Know what happens when a page is requested from the WWW, from the server where the page is stored to the web browser where it was requested. |  |  |  |  |  |  |  |
| Know why information on the WWW does always follow the same route from one location to another - explain why this is key to the web’s success. |  |  |  |  |  |  |  |
| Know the role Tim Berners-Lee had in the development of the WWW. |  |  |  |  |  |  |  |
| Understand the need for using advanced search methods to reduce the number of unwanted ‘hits’ - use such methods. |  |  |  |  |  |  |  |
| Be able to locate relevant resources to support personal educational needs. |  |  |  |  |  |  |  |
| Explain the personal and economic dangers of using the internet, including for young people, and steps needed to reduce risks. |  |  |  |  |  |  |  |
| Explain what ergonomics are and why they are important for personal well-being when using computer hardware. |  |  |  |  |  |  |  |
| Describe some steps computer users can take to ensure their physical well-being when using computers. |  |  |  |  |  |  |  |

# ADDITIONAL Subject/SOFTWARE SKILLS

**Describe any additional experience/understanding you have with the following (please expand space to fit your information):**

* Use of Scratch and Logo
* Web development (e.g. Adobe Dreamweaver, MS FrontPage, MS Expression Web)
* Use of Web 2.00 tools such as Google sites to share resources
* Use of video editing software
* Use of sound manipulation/recording software
* Programming software (for example Pythion)
* Use of text editor to create HTML pages.
* Any other aspect that you consider to be useful prior experience in this subject

# 