

ITAP 1 – How do I start lessons in a calm and purposeful manner?

To begin to be able to:

- Start a lesson in a calm and controlled manner
- Obtain silence and pupils' attention
- Respond effectively to disruption during teacher talk

We recommend the teaching episodes in ITAP 1 are around pupils entering, engaging with entrance tasks (if appropriate) and student teachers completing the review of the task, obtaining and maintaining silence in the process

ITAP 1 – Computing Guidance:

- We want student teachers to take responsibility for the start of lessons, welcoming pupils into the computer room, directing them to allocated seats and ensuring a starter task is visible immediately. They should minimise 'dead time' by having software, resources and equipment prepared in advance and establish clear routines around screens, equipment and movement around the room.
- Once pupils are working, student teachers should practise gaining whole-class attention using strategies such as countdowns, locking screens, asking pupils to move away from keyboards or turn towards the teacher.
- They should circulate during practical work, monitor pupils' screens and respond calmly and consistently to off-task behaviour in line with school policy.

ITAP 2 – How do I model and explain knowledge, processes and skills in my subject?

To begin to be able to:

- Model and explain one aspect of my subject
- Break down complex concepts or skills into bite-size chunks
- Practise the ‘hand-over’ from teacher to pupil activity

ITAP 2 – Computing Guidance:

- Student teachers should practise modelling a Computing skill or process, such as writing a short program, debugging code, using software or constructing an algorithm. Demonstrations should be broken into small, manageable stages, with pupils watching before attempting each stage themselves.
- Student teachers should explain their thinking as they model, use worked examples and check understanding before handing responsibility over to pupils. Technical demonstrations should always be rehearsed beforehand to ensure that software, files and instructions work correctly in the teaching room.

ITAP 3 – How do I ask and respond to questions in my subject?

To begin to be able to:

- Ask effective questions to check pupil understanding
- Respond effectively to hesitant or incorrect pupil responses
- Explore different modes of pupil responses (e.g. written responses)

ITAP 3 – Computing Guidance:

- Student teachers should plan and use a range of questioning techniques to check pupils' Computing knowledge and understanding. This could include cold calling, hinge questions, mini-whiteboards, online quizzes, voting, think-pair-share and asking pupils to explain or extend another pupil's answer.
- Questions should also be used during programming and demonstrations, for example asking pupils to predict output, identify errors, explain what a line of code does or suggest the next step. Student teachers should use pupils' responses to identify misconceptions and decide whether to continue, re-explain or re-model

ITAP 4 – How do I deploy a subject-specific pedagogy effectively?

To begin to be able to:

- Use subject-specific pedagogy to support pupil knowledge, understanding and skills
- Promote the use of appropriate subject-specific vocabulary
- Recognise and respond to common pupil misconceptions associated with learning my subject

Subject leads to identify the specific pedagogy and send general mentors the brief overview of supporting literature

ITAP 4 – Computing Guidance:

- The focus for Computer Science will be **PRIMM – Predict, Run, Investigate, Modify, Make**. Student teachers should plan a programming episode where pupils first examine and understand existing code before gradually moving towards modifying and creating programs themselves.
- Student teachers should use questioning and discussion particularly during the Predict and Investigate stages, then provide appropriate scaffolding during Modify before giving pupils greater independence during Make.
- PRIMM techniques can be used across a *series* of lessons and **not necessarily every aspect needs to be covered**. However, the techniques are useful to incorporate into their programming teaching where appropriate.