COMPUTATIONAL THINKING SKILLS

GUIDE ON DESIGNING ACTIVITIES

Abstraction

1. Hide details of an idea, problem, or solution that are not relevant, to focus on a manageable number of aspects.
2. Create a representation (idea) of what you are trying to solve.
3. Choose a way to represent an artefact, to allow it to be manipulated in useful ways.

Pattern Recognition

1. Analyse the data and look for patterns that make sense.
2. Find the similarities or patterns among small, decomposed problems.
3. Make predictions about what will happen next.
4. Transfer ideas and solutions from one problem area to another.

Decomposition

Break down a complex problem / artefact into smaller parts that can be understood, solved, developed and evaluated separately.

Evaluation

1. Assess a solution and see if it can be generalized via automation or extension.
2. Assess whether an artefact does the right thing (functional correctness).
3. Design and run test plans and interpret the results (testing).
4. Use rigorous argument to check the usability or performance of an artefact (analytical evaluation).
5. Use methods involving observing an artefact in use to assess its usability (empirical evaluation).

Algorithmic Thinking

1. Create step-by-step instructions for solving the problem or completing a task.
2. Explicitly state the algorithm steps.
3. Identify different effective algorithms for a given problem.
4. Find the most efficient algorithm