The Towers of Brahma

|  |  |  |  |
| --- | --- | --- | --- |
| **Subject** | Computational thinking | **Prepared By:** | Panagiotis Grontas |

|  |
| --- |
| **Overview & Purpose**  The participants will be introduced to the principles of computational thinking through the old Hindu legend about the towers of Brahma. They will learn the basic components of algorithms (sequence, selection, recursion) and problem-solving techniques (synthesis, analysis). They will apply their knowledge and experiment with actual computer code solving the problem. Finally, they will learn the basic principles of computational complexity and use them to evaluate the quality of their solution. |

|  |  |
| --- | --- |
| **Guide** | |
| **Objectives** | * Learn the basic principles of computational thinking * Apply them to a computational problem * Create an algorithm by experimenting using an online activity * Implement the algorithm in Python using an online editor * Evaluate the quality of their solution |
| **Information** | * The legend of the towers of Brahma: Reality or Fiction? * Real world value of computational thinking * Algorithm components (sequence, selection, recursion) * Analysis and synthesis * The basics of computational complexity |
| **Activity 1** | * Try to solve the problem in [https://www.mathsisfun.com/games/towerofhanoi.htm](https://www.mathsisfun.com/games/towerofhanoi.html) for 3 discs * Repeat for 4 and 5 discs * Record the numbers of moves in each case * Try to infer the sequence of actions that are performed, the constraints and the repetitions |
| **Activity 2** | * Apply the principles of problem decomposition and combination to the online game * Solve the problem for 4 discs using these techniques * Try to write them more compactly |
| **Activity 3** | * Implement the algorithm in the Python online editor at [https://repl.it/KiNR/](https://repl.it/KiNR/6)11 * Use the predefined functions * Fill in the required actions from activity 2 |
| **Take home** | * Computational thinking has little to do with computers * Algorithm design can be aligned with general problem-solving techniques * Programming languages allow us to think in code * The quality of the solution counts |