

Title: Week-at-a-Glance (March 17–21, 2025)

Class: Introduction to Hardware Technology

Topic: Computational Thinking in Hardware Technology - Week 1

Dates: Monday, March 17 – Friday, March 21, 2025

Day	Lesson Focus	Academic Standards	Learning Targets	Success Criteria	Literacy Focus	Work Session	Assessment/Evaluation
Monday	Introduction to Computational Thinking in Hardware	IT-IHT-1: Demonstrate employability skills required by business and industry.	<ul style="list-style-type: none">- Define computational thinking (CT) and its relevance to hardware technology.- Identify the four components of CT: decomposition, pattern recognition, abstraction, and algorithms.	<ul style="list-style-type: none">- Can explain the importance of CT in hardware contexts.- Can list and describe the four components of CT.	Vocabulary: Computational Thinking, Decomposition, Pattern Recognition, Abstraction, Algorithm	<ul style="list-style-type: none">- Lecture on CT principles and their application in hardware technology.- Group discussion on real-life hardware problems solvable through CT.	<ul style="list-style-type: none">- Participation in class discussion.- Exit ticket: Summarize CT and its components in hardware contexts.
Tuesday	Decomposition: Breaking Down Hardware Problems	IT-IHT-6: Utilize computational thinking procedures to analyze and solve	<ul style="list-style-type: none">- Understand decomposition in CT.- Apply decomposition to dissect complex	<ul style="list-style-type: none">- Can break down a hardware problem into smaller tasks.- Can explain how decomposition	Vocabulary: Decomposition, Sub-problems, Task Analysis	<ul style="list-style-type: none">- Interactive activity: Decompose a complex hardware issue (e.g., computer	<ul style="list-style-type: none">- Completion of decomposition activity.- Reflection: Describe how decomposition was applied in the

		hardware problems.	hardware issues into manageable parts.	aids in hardware troubleshooting.		not booting) into smaller tasks. - Discuss how decomposition facilitates efficient hardware troubleshooting.	activity and its effectiveness.
Wednesday	Pattern Recognition: Identifying Hardware Issue Trends	IT-IHT-6: Utilize computational thinking procedures to analyze and solve hardware problems.	- Grasp pattern recognition in CT. - Identify patterns in hardware failures to predict and solve issues.	- Can recognize patterns in hardware issues. - Can use identified patterns to inform troubleshooting strategies.	Vocabulary: Pattern Recognition, Trends, Data Analysis	- Analyze hardware failure logs to identify common issues. - Group discussion on how recognizing patterns can lead to more efficient hardware problem-solving.	- Participation in data analysis activity. - Quiz on identifying patterns and explaining their significance.
Thursday	Abstraction: Focusing on Critical Hardware Details	IT-IHT-6: Utilize computational thinking procedures to analyze and solve hardware problems.	- Comprehend abstraction in CT. - Learn to focus on essential hardware details while ignoring	- Can abstract key information from complex hardware scenarios. - Can explain how abstraction simplifies hardware	Vocabulary: Abstraction, Essential Details, Simplification	- Case study analysis: Identify the core hardware problem in a complex scenario by filtering out	- Completion of case study analysis. - Group presentation on the importance of abstraction in hardware problem-solving.

			irrelevant information.	problem-solving by reducing complexity.		unnecessary details. - Discussion on the role of abstraction in hardware troubleshooting.	
Friday	Algorithms: Creating Step-by-Step Hardware Solutions	IT-IHT-6: Utilize computational thinking procedures to analyze and solve hardware problems.	<ul style="list-style-type: none"> - Understand what algorithms are and their role in CT. - Develop simple algorithms to solve specific hardware problems. 	<ul style="list-style-type: none"> - Can create clear, step-by-step instructions (algorithms) for solving a hardware problem. - Can explain the importance of precision and clarity in algorithm development. 	Vocabulary: Algorithm, Step-by-Step Instructions, Flowchart	<ul style="list-style-type: none"> - Workshop: Write algorithms for hardware troubleshooting tasks (e.g., diagnosing a non-functional printer). - Introduction to flowcharting as a tool for visualizing troubleshooting steps. 	<ul style="list-style-type: none"> - Submission of written algorithms. - Peer review: Evaluate the clarity and effectiveness of classmates' algorithms.