ﺍﻟﻤﺪﺭﺳـــﺔ ﺍﻷﻣﺮﻳﻜـــﻴﺔ ﺍﻟﺪﻭﻟـــﻴﺔ
American International School

HIGH SCHOOL

International Baccalaureate
Baccalauréat International
Bachillerato Internacional

Design Technology
AY 25-26

STRAND	STANDARDS/SKILLS (International Society for Technology in Education – ISTE)	9th	10th	11th/12th
Empowered Learner	EL1.12.2 build networks and customize their learning environments in ways that support the learning process.	Understand how to use digital collaboration tools to connect with peers and experts.	Understand how to leverage engineering software and simulation tools to collaborate on projects. Understand how to utilize coding communities and version control systems (like CodeHS) to collaborate on projects.	Understand how to use AutoCAD's collaboration tools to work on shared design files.
	EL1.12.4 understand the fundamental concepts of technology operations, demonstrate the ability to choose, use and troubleshoot current technologies and are able to transfer their knowledge to explore emerging technologies.	Understand the basic operations of digital design tools and software (e.g., CAD, 3D modeling, graphic design software).	Understand engineering software fundamentals, including CAD and simulation tools.	Understand basic AutoCAD operations and advanced customization features. Choose the Right Tools: Select appropriate AutoCAD tools and features, such as 2D drawing, 3D modeling, and annotation, depending on the specific task or project requirements. .
	EL1.12.3 use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.	Understand how to use digital platforms (e.g. Teams) to collect and analyze feedback on design projects.	Learn to use engineering software to present simulations and prototypes for feedback (e.g., CAD reviews, Teams). Understand methods of code sharing and collaboration through platforms like CodeHS or code review errors.	Understand how to share drawings and designs through cloud-based CAD platforms, Teams for feedback. Understand how to use AutoCAD to create 3D models of their designs, providing a more dynamic demonstration of their learning. They can then present these models through virtual walk-throughs or animations to highlight their understanding and growth.
	EL1.12.1 articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.	Understand how to assess whether their goals were met and how technology supported their efforts. This might involve analyzing project outcomes like product functionality, user feedback, or design success and how these results connect to their original learning goals.	Start identifying specific technical skills they aim to improve, such as enhancing their problem-solving abilities or mastering design software. They will leverage technologies such as CAD tools and simulation software to support their learning strategies. Through continuous reflection on their designs, prototypes, and solutions, students will assess their learning, identify challenges, and make improvements to refine their engineering skills. Start articulating personal goals around mastering programming concepts such as coding syntax, algorithms, or debugging.	Start setting goals to master specific features of the software, such as 2D drawing, 3D modeling, or design automation. They will develop strategies by utilizing AutoCAD tools, tutorials, and digital resources to achieve these goals. Continuous reflection will help students assess the quality and efficiency of their designs, identify areas for improvement, and adapt their workflow to enhance their overall proficiency in AutoCAD.

Digital Citizen	DC1.12.2 engage in positive, safe, legal and ethical behavior when using technology, including social interactions online or when using networked devices.	Collaborate respectfully when sharing design ideas through digital platforms. Use secure cloud storage for design files to prevent unauthorized access. Respect intellectual property by properly attributing any resources or references used. Ensure the designs are original and do not infringe on copyrights or patents.	Communicate effectively and respectfully when sharing CAD models and technical documentation. Protect project data and design prototypes by using secure file management systems. Follow licensing agreements for software like Fusion360 or SolidWorks. Maintain honesty and transparency when documenting and presenting engineering solutions. Follow collaborative coding practices by giving credit to contributors. Secure code to prevent vulnerabilities that could harm users or systems. Avoid plagiarism by writing original code or citing any borrowed scripts. Use open-source code responsibly and acknowledge authors where applicable.	Share CAD drawings and models in a professional and respectful manner. Protect CAD files with secure access controls and backup regularly. Adhere to software licensing terms and avoid sharing copyrighted files without permission. Clearly differentiate between original work and referenced templates or drawings.
	DC1.12.3 demonstrates an understanding of and respect for the rights and obligations of using and sharing intellectual property.	Students often use resources such as images, templates, models, and technical data. Demonstrating understanding means: - Citing sources when incorporating existing designs or inspiration. - Ensuring that any third-party resources used in projects are legally acquired and appropriately licensed. - Respecting copyright by not reproducing or distributing designs without permission. - Applying Creative Commons licenses when sharing their own creations.	Students often involve CAD models, technical diagrams, and innovative product ideas. Demonstrating respect for intellectual property in this context includes: - Protecting original engineering solutions with patents when applicable. - Properly acknowledging any collaborative input or references used in the engineering process. - Avoiding the unauthorized replication of patented mechanisms or models. - Understanding the implications of reverse engineering and its legal limitations. Intellectual property considerations revolve around code usage, algorithms, and software development. Demonstrating understanding includes: - Using open-source libraries responsibly by following license agreements. - Giving credit to the original authors when using or adapting code snippets. - Avoid plagiarism by writing original code and citing borrowed solutions. - Respecting software licenses when integrating external code into projects.	Designs and drawings can be shared, modified, or reused. Respecting intellectual property involves: - Properly attributing templates, blocks, or referenced drawings that are not original. - Understanding the licensing terms of AutoCAD itself and any third-party plugins. - Not sharing or distributing CAD files that contain proprietary information without permission. - Acknowledging collaborative input and giving credit to peers when designs are shared or built upon.

Knowledge Constructor	KC1.12.1 plan and employ effective research strategies to locate information and other resources for their intellectual or creative pursuits.	Students will know how to define the research question or problem they need to solve. Discuss the importance of setting goals for their research (e.g., what specific information or resources they need, what methods they'll use). Encourage creating a step-by-step plan for gathering data or resources, including identifying reliable sources.	Students will know how to define the research question or problem they need to solve. Discuss the importance of setting goals for their research (e.g., what specific information or resources they need, what methods they'll use). Encourage creating a step-by-step plan for gathering data or resources, including identifying reliable sources. Provide examples of how research can inspire creativity and improve the quality of their work.	Identifying the technical drawing requirements and design objectives. Searching for CAD standards, drawing conventions, and industry best practices. Accessing AutoCAD tutorials, official guides, and repositories of CAD blocks and templates. Adapting templates or existing drawings while ensuring compliance with standards. Properly attributing any reused CAD components and maintaining a clear record of resources used.
	KC1.12.2 evaluates the accuracy, perspective, credibility and relevance of information, media, data or other resources.	Students will know how to evaluate the credibility of sources (e.g., distinguishing between primary and secondary sources, verifying authorship and date of publication). Provide students with various sources (some credible, some not) and have them assess the reliability and relevance for their design projects.	Students will know how to evaluate the credibility of sources (e.g., distinguishing between primary and secondary sources, verifying authorship and date of publication). Provide students with various sources (some credible, some not) and have them assess the reliability and relevance for their design projects.	Students will know how to synthesize research findings and incorporate them into their design thinking and process.
	KC1.12.4 build knowledge by actively exploring real-world issues and problems, developing ideas and theories, and pursuing answers and solutions.	Have students research and analyze issues, such as sustainability in design, accessibility, or technological innovation, and relate them to their projects. Encourage prototyping, testing, and refining their solutions to address the problems they've explored, using both research and creativity.	Promote brainstorming sessions, mind-mapping, and the development of innovative ideas to solve the problems they've identified. Encourage prototyping, testing, and refining their solutions to address the problems they've explored, using both research and creativity.	Encourage prototyping, testing, and refining their solutions to address the problems they've explored, using both research and creativity. Encourage brainstorming sessions, sketching, and the development of creative design concepts to solve the problems they've identified in AutoCAD.
Innovator Designer	ID1.12.1 know and use a deliberate design process for generating ideas, testing theories, creating innovative artifacts or solving authentic problems.	In design technology, this often involves stages like research, concept development, prototyping, testing, evaluation, and refinement. brainstorming and exploring a variety of ideas before settling in a direction. This is where creativity and problem-solving come into play.	Identify and follow a design process that guides the work from initial ideas to the final product. This could include using methods like Design Thinking, the Engineering Design Process, or any other systematic approach to design.	During the idea generation phase, use techniques like mind mapping, sketching, or concept boards to come up with different potential solutions. The focus is on creativity and innovation.
	ID1.12.2 select and use digital tools to plan and manage a design process that considers design constraints and calculated risks.	Use digital tools to monitor progress, adjust plans as needed, and communicate with the team. For example, update tasks on Teams under the design channel and track progress using the plan.	When using digital tools, make sure they allow to account for these constraints in the designs. For example, using CAD tools/ Fusion 360, can set specific parameters (like material types, weight limits, or size restrictions) to ensure the design stays within given constraints. Use digital tools to monitor progress, adjust plans as needed, and communicate with the team.	Using digital tools, make sure they allow us to account for these constraints in designs. For example, using CAD tools / AutoCAD, can set specific parameters (like material types, weight limits, or size restrictions) to ensure the design stays within given constraints.

			For example, update tasks on CodeHS and track progress.	
	ID1.12.3 develop, test and refine prototypes as part of a cyclical design process.	Once created a prototype, put it to the test. For example, if designing a chair, test its stability, comfort, and durability by using it for an extended period. During testing, gather feedback from users or experts (if applicable). This will help to understand the real-world performance and usability of the design.	test specific features of the prototype (e.g., a mechanism, material strength, usability) to see if they meet the design specifications. During the testing phase, we invite the guest speaker to provide feedback to the students.	Use digital tools (like CAD software) or physical materials to create the first prototype. For example, if designing a new product, start by building a basic working version that represents the core features. If it's 3A design, we can test the design by working with different layers and materials from different views. Based on testing results, identify any weaknesses or limitations in the design. We make changes based on the analysis. For example, if the prototype doesn't hold up under pressure, might change materials or reinforce weak points.
Computational thinker	CT1.12.1 formulates problem definitions suited for technology-assisted methods such as data analysis, abstract models and algorithmic thinking in exploring and finding solutions.	Clearly stating the problem by identifying the design challenge and user needs. Technology-Assisted Methods: - Analyzing user feedback or testing data to identify areas of improvement. - Creating prototypes or simulations to predict how the design will perform. - Developing step-by-step plans to test and evaluate design iterations.	Identifying engineering challenges by defining performance metrics and constraints. Technology-Assisted Methods: - Gathering data on material properties and stress tests to ensure durability. - Creating CAD models and running simulations to predict mechanical performance. - Developing flowcharts to map out production processes and automate quality control. Clearly stating the programming challenge and desired output. Technology-Assisted Methods: - Processing datasets to identify patterns or anomalies. - Designing flowcharts or pseudocode to break down the problem logically. - Creating algorithms to automate repetitive tasks or solve complex problems.	Defining drafting and design challenges with clear specifications. Technology-Assisted Methods: - Measuring dimensions and tolerances accurately to meet design requirements. - Building 3D models to visualize spatial relationships and structural integrity. - Using parametric modeling to automate repetitive design elements.
	CT1.12.2 collect data or identify relevant data sets, use digital tools to analyze them, and represent data in various ways to facilitate problem-solving and decision-making.	Gathering user feedback through surveys, interviews, or product testing. - Data Analysis: Using spreadsheet software (like Excel or Google Sheets) to analyze user satisfaction data. Utilizing statistical methods to identify trends or preferences. - Data Representation: Creating bar graphs, pie charts, and histograms to visualize user preferences.	Measuring mechanical properties (like stress, strain, and load capacity) during testing. - Data Analysis: Using simulation software to analyze structural integrity. Applying data analysis tools to compare material performance under different conditions. - Data Representation: Generating 3D visualizations and stress distribution graphs. Creating technical reports that include data tables and annotated diagrams.	Extracting measurements from CAD drawings or collecting spatial data from GIS integrations. - Data Analysis: Analyzing geometric data to calculate areas, volumes, or material quantities. Comparing design iterations to identify improvements or optimizations. - Data Representation: Generating annotated diagrams, technical drawings, and schematics. Creating data-driven models that automatically update dimensions or calculations.

		Designing infographics to present survey results or testing outcomes.		
	CT1.12.3 break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving.	Decomposing a design challenge into elements such as functionality, aesthetics, user needs, and materials. - Identifying design specifications, material properties, and user requirements. - Sketching and Prototyping: Creating visual representations or physical models to convey ideas. - Using CAD to create 3D modelling software to create detailed models of the final product.	Decomposing an engineering challenge into components like load capacity, material strength, safety features, and cost efficiency. - Gathering data from technical specifications, performance metrics, and material tests. - Mathematical Models: Calculating stress, strain, and load distribution. - Creating 3D simulations to predict structural performance. Dividing a programming task into functions, modules, and algorithms. - Identifying inputs, outputs, and constraints. - Flowcharts and Pseudocode: Structuring the logic before coding. - Designing algorithms to solve each sub-problem.	Breaking down a complex drawing into layers, components, and annotations. - Identifying crucial measurements, dimensions, and spatial relationships. - 2D and 3D CAD Models: Visualizing individual components and their assembly. - Adding notes and dimensions to clarify design intent.
	CT1.12.4 understands how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.	Understanding how automated manufacturing (e.g., 3D printing) works. Developing a step-by-step workflow to automate production process. Testing and troubleshooting automated processes. Students create a flowchart outlining the steps of an automated assembly line, then simulate or model the process using software.	Breaking down problems into clear, logical steps that can be translated into code. Writing and testing programs that automate tasks and debugging them as needed.	Breaking down complex drawing tasks into simpler steps that can be automated using AutoCAD's tools.
Global Collaborator	GC1.12.1 choose the appropriate platforms and tools for meeting the desired objectives of their creation or communication.	- Identify the design requirements and objectives. - Research and evaluate different digital and physical tools (e.g., CAD software, modeling tools, prototyping equipment). - Justify the choice of tools based on accuracy, efficiency, and relevance to the task.	- Analyze project requirements to identify the most suitable engineering software or hardware tools. - Evaluate platforms for simulation, testing, and prototyping (e.g., MATLAB, CAD software, Arduino IDE). - Consider precision, accuracy, and compatibility with the design objectives. - Understand the purpose and features of different programming languages (e.g., HTML, CSS for data analysis, JavaScript for web apps). - Justify the choice of language or platform based on the	Analyze project specifications to identify suitable AutoCAD features (e.g., 2D drafting, 3D modeling, annotation tools). Choose between different CAD formats and presentation methods. Justify choices based on precision, clarity, and professional standards. The effectiveness of chosen tools in communicating the design clearly and accurately.

			project's technical and functional requirements.	
	GC1.12.2 create original works or responsibly repurpose or remix digital resources into new creations.	- Designing unique products or prototypes from scratch based on specific user needs or requirements. - Using existing designs, parts, or materials in new ways to create something original. - Ensuring that any resources or designs taken from the internet or other creators are used with permission or are in the public domain, and properly attributing them when necessary.	- Developing innovative mechanical or electrical systems and designs that solve complex engineering problems. - Incorporating existing engineering designs or technologies (like open-source mechanical parts) to develop new products. - Following intellectual property laws, obtaining licenses for third-party designs, and attributing sources when remixing designs or using others' work. - Writing original programs, applications, or websites that provide innovative solutions to real-world problems. - Using open-source code or libraries to build upon existing work, such as remixing a code snippet from GitHub to create a more personalized or functional version. Ensuring that all reused code follows open-source licensing agreements, providing proper attribution, and respecting copyrights.	- Designing original blueprints, floor plans, or 3D models that fulfill specific project requirements. - Using existing CAD blocks, templates, or 3D models from design libraries, while modifying them to suit the project. - Ensuring any CAD blocks or resources reused are properly licensed or free for commercial use, and crediting the original creator when necessary.
	GC1.12.3 communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations, models or simulations.	- Students might use tools like CAD software to create 3D models, prototypes, or simulations that represent their design ideas. - Students will need to present their ideas to peers, teachers, or potential customers using clear, understandable visuals. - Creating presentations that use 3D models, product sketches, or flow diagrams to help explain the design process and decisions clearly.	- Students might use engineering simulation software, CAD tools, and other digital modeling applications to present technical designs. - Students will need to communicate design concepts clearly using visual representations like engineering drawings, schematics, or CAD models that break down the system's functionality. - Running simulations of mechanical systems or stress tests to predict performance	- students will design detailed 2D drawings or 3D models that visually communicate architectural or engineering designs. - The goal is to use AutoCAD to present complex designs clearly with accurate technical drawings, layouts, and visualizations. - Students can represent designs as realistic 3D models or use AutoCAD's rendering tools to create photorealistic images of the design for better communication.

			and visually presenting this data - Programming students may create visualizations of algorithms or data flow, use interactive simulations to demonstrate how programs work, or design models to represent data structures. - Students will need to break down and present their code or algorithms using flowcharts, code visualizations, or interactive simulations. - Students may build simulations of programs or models to show how their code interacts with inputs, outputs, and real-world scenarios.	