Talent Definitions

The diverse individual goals supported by Arts & Bots and the Creative Robotics MSP will allow for engagement with student of numerous talents and interests. These include talents in aesthetic and visual communication, computational thinking, engineering design and others. The STEM Talent Identification and Cultivation focus of the MSP project places special interest on the identification of Computational Thinking and Engineering Design talents.

Computational Thinking Talent

Computational thinking (CT) is a set of problem solving skills and techniques which incorporate attitudes and skills that allow real world problems to be solved with methods from computing and computer science (Wing 2006). CT revolves around restructuring and modeling problems so that they can be solved through logical, algorithmic thinking (ISTE & CSTA 2011). CT exercises students' skills in handling complexity, ambiguity, and open-ended problems; persistence in working with difficult problems; and communicating and working with others to achieve a common goal (ISTE & CSTA 2011).

We define ***exceptional computational thinking talent*** as demonstrating above average abilities:

- thinking through different levels of abstraction
- formulating logical data organizations and algorithmic processes
- comparing the efficiency and effectiveness of feasible solutions

PS	Problem-Solving

PS.1	**Problem Breakdown**			
Prominent in:	★★ Planning	Building	Programming	Testing
Description	The student can take a large problem and divide it into smaller problems that are each more manageable and, when each is solved, the complex problem becomes easier.			
Example	For example, a student is looking to have a robot perform a complex and long set of actions. Instead of coding it all up into one giant *sequence*, the student thinks of natural, smaller "abilities" for the robot, making a *subsequence* for each of these and testing them individually, then finally making a *sequence* that uses all these smaller *subsequences* together.			

PS.2	**Redefine problems**			
Prominent in:	★★ Planning	Building	Programming	Testing
Description	The student recognizes that a given problem cannot be solved with available resources. She can take that problem and express it in a different way so that available tools (such as the motors and sensors that are available) are more applicable.			
Example	For example, the student is trying to get a robot dog's tail to wag back and forth continuously but cannot get the servo to move slowly enough. Instead of thinking about the problem in terms of a servo moving slowly, continuously, the student can create a program that sets the position of the tail to a series of closely spaced positions, and over ten or twenty such moves, it gives the impression of a slowly wagging tail.			

PS.3	Strategic decision-making			
Prominent in:	★ Planning	★★ Building	★★ Programming	Testing
Description	The student compares and weighs possible strategies and solutions, and she is able to make a justifiable decision concerning how to proceed.			
Example	For example, a student wants her robot to blink its eye twenty times, and considers two ways of doing that: making a *sequence* that lists out the blinking *expression* twenty times in a row, or using a counter loop. The student considers which is more work to implement and which is easier to change later. The student chooses to use the loop, because it will be easier to change the number of blinks later on.			

AB Abstraction

AB.1	Modelling			
Prominent in:	★ Planning	★ Building	★★ Programming	Testing
Description	The student is able to create a model or simulation to represent a complex system in order to better understand the system. The model represents key elements of the system while ignoring superfluous details.			
Example	For example, the student is trying to create a robotic arm that demonstrates the movement of bones, muscles and tendons. The student first models the elbow joint of the robot using just the servo and two rectangles of cardboard to test the servo motions, prior to creating bone shaped pieces and adding decorative muscles to the final system.			

AB.2	Pattern Recognition			
Prominent in:	Planning	★ Building	★★ Programming	Testing
Description	The student is able to consider multiple tasks and recognize the common features that the tasks share.			
Example	For example, a student is programming a complex robot behavior with numerous *subsequences*. The student recognizes that a desired action is similar to an existing *subsequence*. She saves a copy of the existing *subsequence* to use as a template and then modifies the *subsequence* as needed without recreating the shared actions.			

AB.3	Modularity			
Prominent in:	Planning	Building	★★ Programming	Testing
Description	A student can recognize which components may be useful for reuse and is able to create solutions that are generalizable for multiple tasks.			
Example	For example, the student is making a robotic mask which expresses different emotions using eye color and mouth position. One solution would be create complete *expressions* for "Jealous", "Sad", "Angry" and "Tired" which each contain both eye and mouth settings. A modular solution would be to create separate eye color and mouth position *expressions* which can be combined in different ways to create a wide variety of expressions.			

AT	Algorithmic Thinking

AT.1	**Algorithm Design**
Prominent in:	★ Planning Building ★★ Programming Testing
Description	The student is able to recognize that complex behaviors are, like a cooking recipe, made of simpler steps put together in a specific way. At design time, they are able to identify and name the smaller steps that happen sequentially in order to recreate the overall behavior. Beyond using *expressions* to make *sequences*, an exceptional student will combine *subsequences* to create larger and more elaborate final *sequences*.
Example	For example, while designing a robotic lobster that will talk about ocean currents, the student recognizes that the overall ocean current behavior includes distinct actions for making the speech, snapping the claws, moving eyestalks and arching the lobster's back. He is able to describe the relative timing of such behaviors, the *sequence* of the actions and their relationships to one-another even before having implemented any of them.

AT.2	**Incremental development and evaluation**
Prominent in:	Planning ★★ Building ★★ Programming Testing
Description	The student is able to solve complex challenges by breaking the problem down and implementing simple, manageable parts. One-by-one the student tests and perfects each part and eventually combines them into the full solutions.
Example	For example, a student is creating a robotic theater with a four act play. She designs, tests, and refines each act separately before combining them to create the complete play.

Engineering Design Talent

Engineering design is the process of developing a concrete solution for an ill-defined problem within technical feasibility constraints (Cross 1982, Brown 2008). Engineering design experience develops students' skills in real world problem solving, synthesizing new thoughts and concepts, and communicating mental imagery of designs and concepts through graphical representations (Cross 1982). Our vision of engineering design in K-12 education is a combination of systems engineering concepts, the engineering design process (comparable to the scientific method) and design thinking as popularized by the Stanford d.School and IDEO (Brown 2008).

We define ***exceptional engineering design talent*** as demonstrating above average abilities:

- transforming ambiguous and complex problem spaces into concrete design goals
- developing new concepts and creative solutions for solving design problems
- communicating design concepts using graphical representations and other nonverbal means
- actualizing designs into real-world prototypes, devices or systems

DP	Defining the Problem

DP.1	Defining the Problem			
Prominent in:	★★ Planning	Building	Programming	Testing
Description	The student can identify criteria for success, constraints and resource limits for a given problem.			
Example	For example, a student is given an open-ended task to make a robot that encourages their peers to recycle cans. The student can recognize the capabilities of materials that they have available to them, they consider the time restraints on the class and determine how they will measure their robot's success at the task.			

ID	Intentional Design

ID.1	Deliberate Planning			
Prominent in:	★★ Planning	Building	Programming	Testing
Description	The student first comes up with a complete plan and strategy for constructing and programming the robot that they intend to create based on the criteria and constraints and considers how to follow this plan it before beginning construction and programming.			
Example	For example, the student will make sketch designs, flowchart behaviors, make lists and/or take notes—all before actually cutting cardboard, heating glue, et cetera.			

ID.2	Following a Plan			
Prominent in:	Planning	★★ Building	★ Programming	Testing
Description	The student has a roadmap for creating the robot and works to follow it despite challenges, rather than changing plans haphazardly while building.			

Example	For example, a student starts with a plan to build a rabbit stand on its rear legs, but then discovers that the rabbit falls over when the servos move. A student who does <u>not</u> stick to the plan may decide, "Well, now I'm building a rabbit that is lying down" without having felt a loyalty to the plan and desire to understand whether she really can get the rabbit to stay standing up. A student who follows the plan will consider the cause of the problem and address it by modifying their robot with a larger base so that the final rabbit matches their planned standing design.

IN	Innovating

IN.1	**Generating Multiple Solutions**			
Prominent in:	★★ Planning	Building	Programming	Testing
Description	The student, before constructing or programming the robot or subcomponent, brainstorms two or more possible solutions for each challenge or need instead of just beginning to create the first solution that comes to mind.			
Example	For example, the student might sketch three different ways of attaching a cardboard arm to a servo (e.g., screwing the cardboard to the servo head; gluing the arm to the servo head; or pressing the servo head through a cut in the cardboard and gluing the edges). This talent can also be applied when facing a new problem discovered during the creation of a robot - when an initial design is ineffective, the student considers multiple improvements before proceeding.			

IN.2	**Solution Evaluation**			
Prominent in:	★★ Planning	Building	Programming	Testing
Description	The student, presented with multiple possible solutions, considers carefully the strengths and weaknesses of each potential solution and is able to describe her reason for making a choice. By considering the constraints, criteria and resources for the project, the student can select the solution which meets the success criteria within the given constraints and resources.			
Example	For example, the student considers two ways to have her robot express emotions, she could use servos to control the shape of the robot's mouth or she could use tri-color LEDs to make the eyes a color symbolic of the emotion. She considers that the mouth movements would be too challenging to implement in the remaining project time, while the tri-color eyes will be able to be completed quickly and will still meet the emotion expression criteria, so she chooses to use the tri-color LEDs.			

IN.3	**"Outside the Box"**			
Prominent in:	★★ Planning	★★ Building	★ Programming	Testing
Description	The student demonstrates the ability to come up with possibly risky, very novel solutions to problems. These solutions might incorporate innovative uses of materials, creative mechanisms or generating a solution unlike any examples shown in class.			
Example	For example, the student is constructing a robot that needs to hold a very large sign. The student develops a solution with a helium balloon connected to the sign, allowing the robot to lift the sign with very little force. This is very unconventional thinking and shows out of the box design ability.			

RT	Refining and Testing

RT.1	Systematic Diagnosis			
Prominent in:	Planning	★ Building	★★ Programming	★★ Testing
Description	The student discovers that her robot is not working as expected, either during construction, programming or when completed, and utilizes a methodical process of elimination to determine the source of the problem.			
Example	For example, an exceptional student with a robot that falls over when the arms move, will carefully consider why it is falling and preform a series of tests to determine the issue. She determines that the robot shifts its weight too far forward in certain poses and makes a wedge to put under the robot to shift the center of gravity so that it stops falling over. Students who do not use diagnosis may perform nonsystematic tests (e.g. varying multiple variables at once), try to fix problems through arbitrary modifications, or try to work around the problem by modifying success criteria.			

RT.2	Trade-offs Consideration			
Prominent in:	Planning	★★ Building	★ Programming	Testing
Description	The student is able to recognize when important goals of her robot are at risk of not being accomplished due to resource limitations. The student can prioritize the success criteria and reduce or eliminate features of low priority in order to reach the high priority goals.			
Example	For instance, the student is making a storytelling humanoid robot and she would like to also have it walk across the table. After a few hours of design work, she realizes that the walking goal is much harder to achieve than the primary storytelling goal, and so decides to spend her remaining time making a better stationary storytelling robot since the walking function was disproportionately challenging given the added value. Notice that this thoughtful process and decision is quite different from a student who gives up on the walking goal because they found it difficult without weighing the impact of that decision.			

RT.3	Thorough Testing			
Prominent in:	Planning	★ Building	★ Programming	★★ Testing
Description	The student is careful to test the functionality of each subcomponent of the robot, each component of the program, and the final resulting robot, comparing the test results to their design plan, their expected behavior, and the final criteria for success.			
Example	For instance, a student is building a ball-throwing robot and while building, she tests whether the robot arm can hold the ball at all and makes refinements until it is able to do so. Later, when testing the completed robot, she repeatedly measures how far the ball is thrown with multiple trials to confirm that the robot reliably meets her final criteria for success, e.g. throwing the ball ten feet or more.			

PR	Prototyping

PR.1	Design for Construction			
Prominent in:	★★ Planning	★ Building	Programming	Testing

Description	The student, while designing and constructing their robot, carefully considers how each component will be constructed. By considering the strengths and weaknesses of materials available the student is able to avoid issues that commonly occur in Arts & Bots projects.
Example	For example, the student is building a robotic draw-bridge model, and recognizes that the cardboard for the bridge bends easily between the ridges of the cardboard. In his initial design he includes a brace underneath the bridge surface with an extra cardboard support perpendicular to the cardboard ridges to correct that material weakness. Also the student will consider the tools available to construct the robot and will plan for how those tools will be used. For example the student building the draw bridge designs the bridge structure such that there are no tight spaces where the glue gun cannot reach and designs doors into the robot so that it is easy to reach the Hummingbird board using a screwdriver in order to wire the robotic components.

PR.2	Making It Real			
Prominent in:	Planning	★★ Building	Programming	Testing

Description	The student is able to take an idea and create a physical model which accurately reflects the original idea. The model is carefully crafted, constructed with attention to detail, and in the end successfully and elegantly meets the initial design criteria.
Example	For example, a student envisions the construction of robotic model of a tree. The student approaches this task with great attentional to detail, first selecting a specific tree species and researching the proper leaf shape, bark texture and branching pattern that identifies that species. She then selects the appropriate materials for replicating those features given the resources available and carefully constructs the tree. In the end, she is satisfied that the robotic tree model successfully matched the design idea as she envisioned it.

CO	Communicating Design

CO.1	Clear Communication of Ideas			
Prominent in:	★★ Planning	★★ Building	★ Programming	★ Testing

Description	The student is able to clearly communicate her design ideas to teammates, teachers and others.
Example	For example, a student has an idea for an elaborate string-pulley mechanism to move a component on his robot and the student is able to explain the mechanism through sketches, diagrams and sentences to accurately and precisely convey the idea to his teammates.

Complementary Student Dispositions

These dispositions are complementary to both the Computational Thinking and Engineering Design talents. Each disposition is broadly applicable to diverse disciplines and enhances the student's ability to utilize the talents described in the components above.

Disp.1	Confidence Dealing with Complexity
Description	The student faces complex challenges with confidence and is able to formulate a plan of action for how to proceed.
Ask Yourself	Does a complex challenge paralyze the student, or does he work quickly to break down the problem into bit-size pieces that can each be dealt with in a reasonable time?

Disp.2	Persistence in Working on Difficult Problems
Description	The student demonstrates a tolerance for early failure, and a willingness and excitement to try again.
Ask Yourself	Does the student recognize that it is alright, when making a complex robot or robot behavior, to try several times and run into dead ends, then back out and try again in order to eventually discover a strategy that is successful?

Disp.3	Flexibility
Description	The student is able to adapt to unforeseen complications and discoveries throughout the project.
Ask Yourself	Is the student able to respond to surprises and lessons learned during the building or programming process and use these to beneficially enhance details and goals in ways that are appropriate?

Disp.4	Tolerance for Ambiguity
Description	The student is able to successfully define and follow her own plan when the presented challenge is ambiguous or goals are ill-defined.
Ask Yourself	Is the student able to take a high-level challenge that does not prescribe exactly how the problem should be solved, and prosper? When there is not a single right answer to the problem or easy-to-follow step-by-step directions, can the student define a goal and chart a path to success on her own?