## CodeAIR Remix Mastery Rubric

| Requirement | No evidence ←----------------------------------------------------------------→ Mastery | | |
|---|---|---|---|
| **Programming Conventions** are followed | <ul><li>Variable names aren't descriptive</li><li>Function names aren't descriptive</li><li>Code blocks inconsistently indented</li><li>Capital letters used</li><li>Code is not organized into sections</li></ul> | | <ul><li>Variable names are descriptive</li><li>Function names are descriptive</li><li>Code blocks consistently indented</li><li>Use of lower case letters and underscores</li><li>Code is organized into sections</li></ul> |
| **Documentation and Readability** | <ul><li>No comments are used.</li><li>Code is difficult to read because no blank lines were used, or too many blank lines were included.</li></ul> | | <ul><li>Frequent and descriptive comments are used regularly.</li><li>Blank lines are used to help with readability.</li></ul> |
| **Use of Variables and constants** | <ul><li>"Magic Numbers" or literal values are used in the code.</li><li>Data isn't tracked or updated (no counters, states, conversions, etc.).</li></ul> | | <ul><li>Constants are used to eliminate "magic numbers."</li><li>Variables are used for storing, keeping track of and updating data.</li><li>Global and local variables are used.</li></ul> |
| **Use of Functions** | <ul><li>No plan or algorithm to follow.</li><li>Everything in one main program.</li><li>Long sections of code.</li><li>Functions use all global or all local variables (no use of parameters or return)</li></ul> | | <ul><li>Code is divided into smaller sections that accomplish a task.</li><li>Parameters are used as needed.</li><li>Local and global variables are used as needed.</li><li>Functions return a value as needed.</li></ul> |
| **Use of libraries and modules** | <ul><li>No libraries are imported.</li><li>No custom modules are imported.</li></ul> | | <ul><li>Public libraries are imported, and functions used.</li><li>Custom modules are imported, and functions used.</li></ul> |
| **Use of Inputs** Buttons and sensors | <ul><li>No input during program run. (from sensors or buttons)</li></ul> | | <ul><li>Input is used during program run. (from sensors or buttons)</li></ul> |
| **Algorithms and Programming** | <ul><li>No algorithms identified or used.</li><li>Program does not utilize input</li><li>Data structures are not used to simplify code. (such as lists and tuples)</li><li>Debugging practices are not used and code contains errors.</li></ul> | | <ul><li>Algorithms are used to manipulate data and get results.</li><li>Data is used to inform decisions.</li><li>Data structures are used to simplify data collection and implementation. (such as lists, tuples, etc)</li><li>Debugging practices are used to correct errors in code and logic.</li></ul> |
| **Control Structures** | <ul><li>Program does not have any branching statements. (if or if/else or if/elif/else)</li><li>Program does not use any iteration. (while or for loops)</li></ul> | | <ul><li>While loops and if statements are used to control the flow of execution.</li><li>Conditional and logical operators are used appropriately.</li><li>Nested while, if statements used when needed.</li></ul> |
| **Use of Outputs** LEDs, speaker, motors | <ul><li>No output is produced; nothing happens.</li><li>Console output is unstructured.</li></ul> | | <ul><li>One or more outputs are used to convey data or perform a task. (motors, lights, speakers, etc.)</li><li>Console output is formatted.</li></ul> |
| **Collaboration** | <ul><li>Students work independently or uncooperatively on a team.</li></ul> | | <ul><li>Students work collaboratively with shared tasks in their team to complete the project.</li></ul> |
| **Synthesis / Purpose** | <ul><li>No clear purpose for the program.</li><li>Project does not incorporate learning across the mission pack.</li></ul> | | <ul><li>Program's purpose is clearly understood.</li><li>Project combines learning, concepts and code from several missions.</li></ul> |
| **Code Completion** | <ul><li>Code will not run or doesn't complete the task correctly.</li></ul> | | <ul><li>Code runs and accomplishes its task without any errors, including logic.</li></ul> |