Project 1

| **Category** | **Score** | **Actual Score** | **Explanation** |
| --- | --- | --- | --- |
| **Program Purpose and Function** | 0 | 1 | Video fully demonstrates the program functionality  Specifies that the purpose is to be used to make a decision  I found the video to be a poor demonstration but it was sufficient for collegeboard |
| **Data Abstraction** | 0 | 0 | Two code segments, but they do not show any complexity or function, as the second does not access a list  The use of the list identifies the components of it |
| **Managing Complexity** | 0 | 0 | A list is shown, but it does not make the program less complex  The use of the list does not result in a program that is easier to develop  Program uses numbers to make a choice |
| **Procedural Abstraction** | 0 | 0 | The response includes a student-developed procedure “rpsGame” with one parameter  Many if/else statements, only says “smooth execution” |
| **Algorithm Implementation** | 1 | 1 | Identifies the calls used  The response  does not specify how the procedure contributes to the overall program, only stating that it  “allows for the program to execute smoothly.” |
| **Testing** | 1 | 1 | Calls are identified and the correct call results are labeled  Program works as intended |

Project 2

| **Category** | **Score** | **Actual Score** | **Explanation** |
| --- | --- | --- | --- |
| **Program Purpose and Function** | 1 | 1 | The video and written response both clearly demonstrate the program functionality and the purpose  Clearly shown input and output; words and the hangman game |
| **Data Abstraction** | 1 | 1 | Two code segments provided  A list is named (“letofguessword”) and identified along with its contents |
| **Managing Complexity** | 1 | 1 | Student explains the importance of the list and how it would not be inconvenient if it wasn't implemented  “If the program were to run without the "letOfGuessWord" list, it would  make everything very inefficient because I would have to make 5 new variables to store the  individual letters” |
| **Procedural Abstraction** | 1 | 1 | The procedure “guesswords” was clearly implemented by the student  Its function is explained in the written response |
| **Algorithm Implementation** | 1 | 1 | The calls are clearly identified and described within the program “guesswords”  Selection and if statements are included |
| **Testing** | 1 | 1 | The results of the calls are successful  The program yields the output of the poems |

Project 3:

| **Category** | **Score** | **Actual Score** | **Explanation** |
| --- | --- | --- | --- |
| **Program purpose and function** | 0 | 1 | I assumed the written response was insufficient, but it was good  specifies the program’s purpose. The response states, “This app can help with memorization or can just be used as a way to learn something new.” |
| **Data abstraction** | 0 | 0 | Two clear code segments, but no code is shown that uses list data |
| **Managing complexity** | 0 | 0 | Explains the use of a list but not how the program would be affected without one  Answers, but is far too general |
| **Procedural abstraction** | 0 | 0 | The response includes procedure “updateScreen” but it does not have  any parameter  second code segment does not show any call to the procedure |
| **Algorithm implementation** | 1 | 0 | Doesn’t state anything about how the index is relevant  Has if/else statements so I thought it would meet the point |
| **Testing** | 0 | 0 | Mixed up the direction of the calls; had them come from instead of go to  The results of these calls are not specified either |

Project 4

| **Category** | **My Score** | **Actual Score** | **Explanation** |
| --- | --- | --- | --- |
| **Program purpose and function** | 0 | 1 | Video demonstrates the running of the program  What I thought it was missing is answered in the written response |
| **Data abstraction** | 1 | 1 | Two clear code segments  The response identifies what is stored in the list.  The data in this list  represents the type of fish and the number of a specific fish caught |
| **Managing complexity** | 1 | 1 | Uses a list of lists to manage complexity. The main list  represents fish, where each list element is also a list that stores the type of fish and the  number of that type of fish that are caught  The changes needed for the program are minimal if another variable is added |
| **Procedural abstraction** | 1 | 1 | Student-developed procedure ‘clone+movement+range’ with parameters that are used in the procedure  Includes calls as well |
| **Algorithm implementation** | 1 | 1 | The response explains in detail how the algorithm in the procedure works so it can be recreated  Thorough explanation |
| **Testing** | 0 | 0 | Does not describe specific arguments that are passed through the parameters  Rather the response explains the alternate coding segments if they were used  Describes the code, not the calls |