|  | **Create PT 20-21 College Board Sample A - Score: 6/6** |  |
| --- | --- | --- |

| **Total score** | Row 1 | Row 2 | Row 3 | Row 4 | Row 5 | Row 6 |
| --- | --- | --- | --- | --- | --- | --- |
| **Sample: 3** | **1** | **1** | **1** | **1** | **1** | **1** |

1. **Program Code**

Your program must demonstrate:

* output (tactile, visual, or textual) based on input from:
  + the user (including user actions that trigger events); or
  + a device; or
  + a file
* use of at least one list (or other collection type) to represent a collection of data related to the program's purpose; and
* development of at least one procedure that uses one or more parameters to accomplish the program's intended purpose, and that implements an algorithm that includes sequencing, selection, and iteration.

Include comments or acknowledgements for any part of the submitted program code that has been written by someone other than you and/or your collaborative partner(s).

Create a PDF file that contains all your program code (including comments).

1. **Video**

Your video must demonstrate your program running, including:

* input to your program; and
* at least one aspect of the functionality of your program; and
* output produced by your program.

Your video:

* must be either .mp4, .wmv, .avi, or .mov format; and
* must not exceed 1 minute in length; and
* must not exceed 30 MB in file size.

Collaboration is not allowed during the development of your video. Your video must not contain any distinguishing information about yourself. Your video must not be narrated, but text captions are encouraged.

1. **Written Responses**

Submit one PDF file that includes your responses to each prompt below. Clearly label your responses 3a-3d in order. Your responses to all prompts combined must not exceed 750 words, exclusive of the program code. Collaboration is not allowed when answering the written responses.

**3a**. Provide a written response that:

* describes the overall purpose of the program; and
* describes what functionality the video illustrates; and
* describes the input and output shown in the video.

| **Student Response** | **Scoring Guidelines** | |
| --- | --- | --- |
| *This program was created in MIT App Inventor to address the issue of learning new languages. Here it teaches the user how to say different colors, where the user inputs what language it wants to hear, either Spanish or French, and then taps on a color, prompting the program to output the audio for that certain color. This allows users to quickly learn how to say colors in another language through interaction and output of audio. In the video, it shows an example of the user clicking on the Spanish checkbox and playing the audio for red and blue. The user can hear what it sounds like, and thus learn how to say it correctly. If the user accidentally inputs no language or both, the program will catch the error and notify the user.* | **Row and Task** | **Decision Rules** |
| **Row 1**  **Video and Written Response 3a**  **Program Purpose and Function**  **4.A, CRD-2B**   * The video demonstrates the running of the program including:   + input   + program functionality   + output   AND   * The written response:   + describes the overall purpose of the program.   + describes what functionality of the program is demonstrated in the video   + describes the input and output of the program demonstrated in the video. | **Consider ONLY the video and written response 3a when scoring this point.**  **Do NOT award a point if the following is true:**   * the video does not show a demonstration of the program running (screenshots or storyboards are not acceptable and would not be credited.) |
| **The response earned the point for this row, meeting all six criteria.**   * The video demonstrates the program receiving user color selection in both French and Spanish as input and producing as output the associated audio response of that color’s pronunciation in the selected language. This satisfies the first three criteria for the video. * The program’s purpose is to “address the issue of learning new languages.” * The functionality demonstrated in the video is “where the user inputs what language it wants to hear, either Spanish or French, and then taps on a color, prompting the program to output the audio for that certain color.” * The input and output demonstrated in the video are described as, “In the video, it shows an example of the user clicking on the Spanish checkbox and playing the audio for red and blue.” | |

**3b.** Capture and paste two program code segments you developed during the administration of this task which contain a list (or other collection type) being used in your program. The first program code segment must show how data has been stored in the list. The second program code segment must show the data in the same list being processed, such as creating new data from the existing data. Then, provide a written response that:

* identifies the name of the list being processed in this response; and
* identifies what the data contained in the list is representing in your program; and
* explains how the selected list manages complexity in your program code by explaining how your program code would be written differently without using this list.

| **Student Response** | **Scoring Guidelines** | |
| --- | --- | --- |
| *The data contained in the list Audio is the list of available color names. It represents all the colors a user can pick for the program in English. These are used to create the corresponding Spanish or French audio files based on which language is selected. When a button is pressed, it will get the English color word from the index of the color in the list. Here, the language the user has chosen does not matter. The program will then create the audio file name for the Spanish or French audio based on what language the user has check marked by manipulating the text (for example, adding“-spanish” to the end if they selected spanish) and then adding “.mp3” to the end in order to call the correct audio file. For the program to function without lists in general, each button will have to call the individual audio file, meaning we would need to have buttons for every color and language combination making the code and user interface more complex.* | **Row and Task** | **Decision Rules** |
| **Row 2 - Response 3b**  **Data Abstraction**  **3.B, AAP-1.C**  The written response:   * includes two program segments:   + one that shows how data has been stored in this list (or other collection type)   + one that shows the data in this same list being used as part of fulfilling the program’s purpose. * identifies the name of the variable representing the list being used in this response * describes what the data contained in this list is representing in the program. | **Consider ONLY written response 3b when scoring this point.**  **Requirements for program code segments:**   * The written response must include two clearly distinguishable program code segments, but these segments may be disjoint code segments or two parts of a contiguous code segment. * If the written response includes more than two code segments, use the first two code segments to determine whether or not the point is earned.   **Do NOT award a point if the following is true:**   * The use of the list is trivial and does not assist in fulfilling the program’s purpose. |
| **The response earned the point for this row, meeting all three criteria.**   * Two distinct code segments are provided: one showing storage of data in a list named Audio; and a second one showing the use of Audio to process output audio to fulfill the program’s purpose. * The name of the list is identified as Audio. * The response states that the data “represents all the colors a user can pick for the program in English. These are used to create the corresponding Spanish or French audio files based on which language is selected.” | |
| **Row 3 - Response 3b**  **Managing Complexity**  **3.C, AAP-3.C**  The written response:   * includes a program code segment that shows a list being used to manage complexity in the program. * explains how the named, selected list manages complexity in the program code by explaining why the program code could not be written, or how it would be written differently, without using this list. | **Consider ONLY written response 3b when scoring this point.**  **Responses that do not earn row 2, may still earn this row.**  **Do NOT award a point if any one or more of the following is true:**   * The code segments containing the lists are not separately included in the written response section (not included at all, or the entire program is selected without explicitly identifying the code segments containing the list). * The written response does not name the selected list (or other collection type). * The use of the list is irrelevant or not used in the program. * The explanation does not apply to the selected list. * The explanation of how the list manages complexity is implausible, inaccurate, or inconsistent with the program. * The solution without the list is implausible, inaccurate, or inconsistent with the program. * The use of the list does not result in a program that is easier to develop, meaning alternatives presented are equally complex or potentially easier. * The use of the list does not result in a program that is easier to maintain, meaning that future changes to the size of the list would cause significant modifications to the code. |
| **The response earned the point for this row, meeting both criteria.**   * The response includes a program code segment that shows the audio list being used to manage complexity in the program, because the list access and index enable the correct setting of a filename for audio file output. * The response explains how the list Audio manages complexity in the program. It states, “When a button is pressed, it will get the English color word from the index of the color in the list…. The program will then create the audio file name for the Spanish or French audio based on what language the user has check marked by manipulating the text (for example, adding ‘-spanish’ to the end if they selected spanish [sic]) and then adding ‘.mp3’ to the end in order to call the correct audio file.” The response also states that the use of lists manages complexity in the project, explaining that “without lists in general, each button will have to call the individual audio file, meaning we would need to have buttons for every color and language combination making the code and user interface more complex.” | |

**3c.** Capture and paste a procedure from your program that you developed during the administration of this task which implements an algorithm used in your program. This procedure must:

* contain and use one or more parameters that have an effect on the functionality of the procedure; and
* implements an algorithm that includes sequencing, selection, and iteration.

Then, provide a written responses that:

* describes what the selected procedure does and how it contributes to the overall functionality of the program; and
* explains how the algorithm implemented in the selected procedure accomplishes its task.

| **Student Response** | **Scoring Guidelines** | |
| --- | --- | --- |
| *This procedure helps to contribute to the overall functionality of the program by selecting the correct audio file based on what language and color the user has selected. Since this algorithm is needed every time a user presses a button, the procedure helps the overall efficiency of the program by having the code located in one location that the program repeatedly calls on. When a user presses a certain color button, the button returns an index pertaining to the main audio list of colors. Then, the procedure takes the index of the color as a parameter and selects the respective list element, which is a string that contains the certain color. Then, depending on the language, the procedure will append a language identifier (for example, “-spanish”) and then add “.mp3”. It will then use this to call a certain color from the database of audio files that is named accordingly. The procedure is able to take parameters and inputs from the user and then output them as such by constructing certain audio file names and then pulling them from the database to play. This action demonstrates selection, and sequencing is when the procedure is able to order tasks accordingly and call from a list. Iteration is used when the procedure senses that there are no languages or both languages selected, and thus repeats the audio of “Select a language!” until the user does so.* | **Row and Task** | **Decision Rules** |
| **Row 4 - Response 3c**  **Procedural Abstraction**  **3.B, AAP-3.C**  The written response:   * includes two program code segments:   + one showing a student-developed procedure with at least one parameter that has an effect on the functionality of the procedure.   + one showing where the student-developed procedure is being called. * describes what the identified procedure does and how it contributes to the overall functionality of the program. | **Consider ONLY written response 3c when scoring this point.**  **Requirements for program code segments:**   * The procedure must be student developed, but could be developed collaboratively with a partner. * If multiple procedures are included, use the first procedure to determine whether the point is earned.   **Do NOT award a point if any one or more of the following is true:**   * the code segment is an event handler; OR * the code segment consisting of the procedure is not included in the written response section; OR * the written response describes what the procedure does independently without relating it to the overall function of the program. |
| **The response earned the point for this row, meeting both criteria.**   * The response includes a student-developed procedure ButtonClicked that contains a parameter index. The parameter index is used in the procedure. Additionally, the response includes an example call to the procedure ButtonClicked that passes the argument “3” to the parameter. * The response describes the purpose of ButtonClicked by stating that it selects “the correct audio file based on what language and color the user has selected.” The response describes how ButtonClicked contributes to the overall functionality of the program by stating it “helps the overall efficiency of the program by having the code located in one location that the program repeatedly calls on.” | |
| **Row 5 - Response 3c**  **Algorithm Implementation**  **2.B, AAP-2.H, AAP-2.K**  The written response:   * includes a student-developed algorithm that includes:   + sequencing   + selection   + iteration * explains in detailed steps how the identified algorithm works in enough detail that someone else could recreate it. | **Consider ONLY written response 3c when scoring this point.**  **Responses that do not earn row 4 may still earn this row.**  **Requirements for program code segments:**   * The algorithm being described can utilize existing language functionality or library calls. * An algorithm that contains selection and iteration, also contains sequencing. * An algorithm containing sequencing, selection, and iteration that is not contained in a procedure can earn this point. * Use the first code segment, as well as any included code for procedures called within this first code segment, to determine whether the point is earned. * If this code segment calls other student-developed procedures, the procedures called from within the main procedure can be considered when evaluating whether the elements of sequencing, selection, and iteration are present as long as the code for the called procedures is included.   **Do NOT award a point if any one or more of the following is true:**   * The response only describes what the selected algorithm does without explaining how it does it. * The description of the algorithm does not match the included program code. * The code segment consisting of the selected algorithm is not included in the written response. * The algorithm is not explicitly identified (i.e., the entire program is selected as an algorithm without explicitly identifying the code segment containing the algorithm). * The use of either the selection or the iteration is trivial and does not affect the outcome of the program. |
| **The response earned the point for this row, meeting both criteria.**   * The student-developed algorithm within procedure ButtonClicked includes sequencing, selection (if...then statement), and iteration (while test...do). * The response explains how the algorithm works. It states that it “takes the index of the color as a parameter and selects the respective list element, which is a string that contains the certain color. Then depending on the language, the procedure will append a language identifier (for example, ‘- spanish’) and then add ‘.mp3.’” The response goes on to describe that iteration is used “when the procedure senses that there are no languages or both languages selected, and thus repeats the audio of ‘Select a language!’ until the user does.” | |

**3d**. Provide a written response that:

* describes two calls to the selected procedure identified in written response 3c. Each call must pass different arguments that cause a different segment of code in the algorithm to execute; and
* describes what condition(s) is being tested by each call to the procedure; and
* identifies the result of each call.

| **Student Response** | **Scoring Guidelines** | |
| --- | --- | --- |
| *The test cases are based on the conditions of what language is selected and what button for what color is pressed, which is represented by the parameter “index”. We would want to check the program for both Spanish and French. For the language Spanish, we could select one of the colors, for example Orange, and the program should give us the correct translation for Spanish. For the language French, we could select one of the colors, for example Orange, and the program should give us the correct translation for French. To further test the program, we should select another color, for example green, and the program should give us the correct translation for the pre-selected language. Each of these test cases executes different parts of the algorithm, going by the condition of which checkboxes are checked for which languages.* | **Row and Task** | **Decision Rules** |
| **Row 6 - Response 3d**  **Testing**  **4.C, CRD-2.J**  The written response:   * describe two calls to the selected procedure identified in written response 3c. Each call must pass a different argument(s) that causes a different segment of code in the algorithm to execute. * describes the condition(s) being tested by each call to the procedure. * identifies the result of each call. | **Consider ONLY written response 3d when scoring this point.**  **Responses that do not earn row 4 may still earn this row.**  **Do NOT award a point if any one or more of the following is true:**   * A procedure is not identified in written response 3c or the procedure does not have a parameter. * The written response for 3d does not apply to the procedure in 3c. * The two calls cause the same segment of code in the algorithm to execute even if the result is different. * The response describes conditions being tested that are implausible, inaccurate, or inconsistent with the program. * The identified results of either call are implausible, inaccurate, or inconsistent with the program. |
| **The response earned the point for this row, meeting all three criteria.**   * The response describes two calls to the procedure: one for Spanish, with the color orange; and one for French, with the color orange. * The response describes the conditions as being whether the user has selected Spanish or French. “For the language Spanish, we could select one of the colors, for example Orange [sic],” and “For the language French, we could select one of the colors, for example Orange [sic].” * The response describes the results being tested as the “correct translation for Spanish” and the “correct translation for French.” | |