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

*(Approx. 150 words, for all subparts of 3a combined)*

1. Describes the overall purpose of the program;

|  |
| --- |
| The Random Dog Picker app displays a random image and name of a dog based on what size is selected. |

1. Describes what functionality the video illustrates;

|  |
| --- |
| The video shows several different sizes being chosen and how a different dog shows up each time. |

1. Describes the input and output of the program shown in the video

|  |
| --- |
| The input is the selection in the dropdown, and the output is the image and name of the dog displayed on the screen. |

**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 to manage complexity in your program.

*(Approx. 200 words, for all subparts of 3b combined, excluding program code)*

1. The first program code segment must show how data has been stored in the list.

|  |
| --- |
|  |

1. The second program code segment must show the data in the same list being processed, such as creating new data from the existing data or accessing multiple elements in the list, as part of fulfilling the program’s purpose.

|  |
| --- |
|  |

Then, provide a written response that does all three of the following:

1. Identifies the name of the list being processed in this response

|  |
| --- |
| The list is filtered based on the size selected. |

1. Describes what the data contained in the list is representing in your program

|  |
| --- |
| If the size chosen is "Small" only dogs whose height is less than 16 will be randomly chosen to be displayed. If the size is "Medium" then the displayed dog needs to be between 16 and 23. If the size is "Large" then dogs that have a height bigger than 24 may be displayed. |

1. Explains how the selected list manages complexity in your program by explaining why your program code could not be written, or how it would be written differently, if you did not use the list

|  |
| --- |
| The list manages complexity because without it, dog heights would all have to be stored in their own individual variables. This would be very confusing, and would add a lot of extra lines. |

**3c.** Capture and paste two program code segments you developed during the administration of this task that contain a student-developed procedure which implements an algorithm used in your program and a call to that procedure.

*(Approx. 200 words, for all subparts of 3c combined, excluding program code)*

1. This first program code segment must be a student-developed procedure that:

* Defines the procedure’s name and return type (if necessary)
* Contains and uses one or more parameters that have an effect on the functionality of the procedure; and
* Implements an algorithm that includes sequencing, selection and iteration.

|  |
| --- |
|  |

1. The second program code segment must show where the student-developed procedure is being called in your program

|  |
| --- |
|  |

Then, provide a written response that does both of the following:

1. Describes in general what the selected procedure does and how it contributes to the overall functionality of the program

|  |
| --- |
| This function filters the lists to smaller lists from which a random name and image can be picked and then displayed. |

1. Explains in detailed steps how the algorithm implemented in the selected procedure accomplishes its task. Your explanation must be detailed enough for someone else to recreate it.

|  |
| --- |
| To do this, a for loop in lines 31-43 traverses the dogHeight list and an if else if statement checks to see if each element fits into what the user wants. If it does, the element at that index in the dogName and in the dogImage lists is added to the filtered lists. |

**3d.** Provide a written response that does all three of the following:

*(Approx. 200 words, for all subparts of 3d combined, excluding program code)*

1. 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;

First call:

|  |
| --- |
| If the user selects a small dog. |

Second call:

|  |
| --- |
| If the user selects a medium dog. |

1. Describes what condition(s) is being tested by each call to the procedure

Condition(s) tested by the first call:

|  |
| --- |
| The function filters the lists to only include small dogs. |

Condition(s) tested by the second call:

|  |
| --- |
| The function filters the list to only include medium dogs. |

1. Identifies the result of each call.

Result of the first call:

|  |
| --- |
| One of those is chosen randomly and displayed on the screen. |

Result of the second call:

|  |
| --- |
| One of those is chosen randomly and displayed on the screen. |