2c. Capture and paste a program code segment that implements an algorithm (marked with an oval in section 3 below) and that is fundamental for your program to achieve its intended purpose. This code segment must be an algorithm you developed individually on your own, must include two or more algorithms, and must integrate mathematical and/or logical concepts. Describe how each algorithm within your selected algorithm functions independently, as well as in combination with others, to form a new algorithm that helps to achieve the intended purpose of the program. *(Must not exceed 200 word)*

<table>
<thead>
<tr>
<th>Scoring Guidelines</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Row and Task</strong></td>
</tr>
<tr>
<td><strong>Row 4 - Code Segment in Response 2C</strong></td>
</tr>
<tr>
<td>Selected code segment implements an algorithm.</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Row 5 - Response 2C</strong></td>
</tr>
<tr>
<td>Selected code segment implements an algorithm that uses mathematical or logical concepts <strong>AND</strong></td>
</tr>
<tr>
<td>Explains how the selected algorithm functions <strong>AND</strong></td>
</tr>
<tr>
<td>Describes what the selected algorithm does in relation to the overall purpose of the program.</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
Row 6 - Response 2C

Selected code segment implements an algorithm that includes at least two or more algorithms

AND

At least one of the included algorithms uses mathematical or logical concepts

AND

Explains how one of the included algorithms functions independently.

- Responses are still eligible to earn this row, even if they do not earn row 5.
- The included algorithms can be sub-parts of the algorithm in row 5.

Do NOT award a point if any one of the following is true:
- the selected algorithm consists of a single instruction; or
- the selected algorithm consists solely of library calls to existing language functionality; or
- neither of the included algorithms nor the selected algorithm that includes two or more algorithms uses mathematical or logical concepts; or
- the code segment consisting of the algorithm is not included in the written responses section or is not explicitly identified in the program code section; or
- 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).

<table>
<thead>
<tr>
<th>Student Response A</th>
<th>Scoring Guidelines</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Video] [Written Response]</td>
<td></td>
</tr>
<tr>
<td><strong>Row 4</strong></td>
<td></td>
</tr>
<tr>
<td>The response earned a point for this row. The code that is given represents an algorithm.</td>
<td></td>
</tr>
<tr>
<td><strong>Row 5</strong></td>
<td></td>
</tr>
<tr>
<td>The response earned a point for this row. The response includes an algorithm that has math/logic (while loop, height/2). The response explains how the wolfMove algorithm works: &quot;The method moves the wolf to the blue chickens and then the chickens' opacity is set to zero.&quot; The response describes what the algorithm does with respect to the entire program. It &quot;contains the code that tells my wolf how to move.&quot;</td>
<td></td>
</tr>
<tr>
<td><strong>Row 6</strong></td>
<td></td>
</tr>
<tr>
<td>The response earned a point for this row. The response gives an algorithm (wolfMove) that uses two algorithms (resetColors and scoreCounter). The response explains how resetColors works: &quot;count the number of blue chickens and proceeds to use logic and call my &quot;setrandomcolors&quot; ... method if there are no blue chickens left.&quot; The resetColors function includes logic using if statements.</td>
<td></td>
</tr>
</tbody>
</table>
I chose to use my “wolfMove” method, which I wrote independently, that contains the code that tells my wolf how to move, as well as two other methods “resetColors” and “scoreCounter”. “wolfMove” uses math to tell the wolf how to move towards the blue chickens and sets their opacity to zero. The method moves the wolf to the blue chickens and then the chickens’ opacity is set to zero. The “resetColors” method uses math to count the number of blue chickens and proceeds to use logic and call my “setrandomcolors” (I will talk more about this method in 2D) method if there are no blue chickens left. My “scoreCounter” method uses math to count the number of chickens that have been “eaten” or disappeared.

Student Response B - [Video] [Written Response]

Row 4

The response earned a point for this row. The circled code segment provided in the response represents an algorithm.

Row 5

The response earned a point for this row. The response includes math/logic through the use of if statements. The response explains how it works overall (“The for-loop iterates over each item in the dictionary and asks the user to enter the term which corresponds to a given definition”). The response also describes what this algorithm does in relation to the entire program (“quizzing the user, and the three smaller algorithms (the if-else-statements) included within the algorithm determines the score and streak-points of the user and uses that information to print adequate motivational messages”).

Row 6

The response earned a point for this row. The response explains that the three if-else structures are the independent algorithms within the main algorithm. Each of these is explained and each uses logic (an if statement along with math).

Student Response C - [Video] [Written Response]

Row 4

The response earned the point for this row. The response explains that the three if-else structures are the independent algorithms within the main algorithm. Each of these is explained and each uses logic (an if statement along with math).
This algorithm is the central decision making part of my program. Located in the Card class and called through each card’s act method, the `turnCard` method will set the image to either a check or x. Each card is instantiated with their order value (1-9), which is shown on the picture of the card. In the Background class, I set a variable to a random integer value, which every card is instantiated with as a parameter. In the constructor, the order value and the random value are assigned to the class’s private instance variables. The order value of the card is then compared with the random value that decides the “x” card as a parameter. **This method tests if the card’s value is the same as the “x” value, then decides if it should call the `setX()` method or the `setCheck()` method.**

If the card’s value is the same as the “x” value, it calls the `setX` method, which changes the image of the card to an “x” and creates a new “Game Over” object and adds it to the middle of the background.

If the card’s value doesn’t match the “x” value, the algorithm calls the `setCheck` method to just set to the “check” image (the player can still flip more cards.) If wished, these two smaller methods could be called independently of the algorithm by a Card object.

---

**Student Response D**

**Artifact** [Written Response]

---

<table>
<thead>
<tr>
<th>Scoring Guidelines</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>The response earned a point for this row.</strong> The code segment provided is an algorithm.</td>
</tr>
<tr>
<td><strong>Row 5</strong></td>
</tr>
<tr>
<td><strong>The response earned a point for this row.</strong> The algorithm provided uses math/logic (if statement). The response explains how the algorithm works: “I set a variable to a random integer value, which every card is instantiated with as a parameter ... this method tests if the card’s value is the same as the &quot;x&quot; value, then decides if it should call the <code>setX()</code> method or the <code>setCheck()</code> method.” The response also describes what the purpose is for this algorithm in relation to the entire program: “The <code>turnCard</code> method will set the image to either a check or x.”</td>
</tr>
<tr>
<td><strong>Row 6</strong></td>
</tr>
<tr>
<td><strong>The response DID NOT earn a point for this row.</strong> The response identifies two algorithms used by the <code>turnCard</code> method, but the two methods do not use math/logic. Setting a variable to true or false is not considered use of logic, just as <code>x = 4</code> is not considered use of math.</td>
</tr>
</tbody>
</table>

**Code.org Commentary:** According to the Scoring Notes, this row could have received the point because the selected algorithm contained an if-statement. However, it’s best to use math or logic in at least one of the included algorithms to ensure the point.

The point should NOT be receive if:

- neither of the included algorithms nor the selected algorithm that includes two or more algorithms uses mathematical or logical concepts; or
This algorithm was one that was fundamentally important in aiding the program to successfully run. In this algorithm, the bowling pin is sensing whether the bowling ball is touching it or not and if it is, the costume of the bowling ball switches from a regular bowling pin to the bottom of a bowling pin to make it look as if the bowling pin has fallen. Next, the score is changed by 1, meaning that every time the bowling ball touches a bowling pin, the score is increased by one point. After that, “message7” is broadcasted so that when the bowling ball receives “message7”, it will immediately disappear, wait 1.5 seconds, and then return to its original position so that the user can continue playing the game. This algorithm functions independently because it is solely for the purpose of making sure that the bowling pin is knocked down. Also, in combination with others, this algorithm helps determine the course of the game because if the bowling pin isn’t being touched, the algorithm won’t run. This algorithm allows other algorithms to be notified that if the bowling pin is being touched, the rest of the program should continue moving forward.

Student Response E - [Video] [Written Response]
This particular algorithm is essential to the program because it allows the user to go to the next flashcard. When the next button is clicked, it displays the label font text and then doesn't show the back text. The index also determined which flashcard that you are on, in which you keep going to the next term as the next button is clicked. Furthermore, if the index is bigger than the number of items in the list then it restarts back to 1, or the first item in the list. This is the same for the other algorithm as they both use an index. One of the independent algorithms makes so that user can type in the term as the other algorithm is displaying the definition as a flashcard. Together as a combination, this makes it so that the user has a study guide environment in which they can type in the necessary term to the definition and then be able to go to the next set of terms.

The response also describes what the purpose is for this algorithm in relation to the entire program: "allows the user to go to the next flashcard" and "the user has a study guide environment in which they can type in the necessary term to the definition and then be able to go to the next set of terms."

---

**Student Response F - [Video] [Written Response]**

```javascript
onEvent("btnPlay", "click", function(event) {
    setScreen("playScreen");
    Time();
});
onEvent("Catimage", "click", function(event) {
    setPosition("Catimage", randomNumber(0,200),
               randomNumber(0,395));
    getText("Score", Score);
    Score = Score + 1;
    setText("Score", Score);
});
onEvent("btnRestart", "click", function(event) {
    setScreen("playScreen");
    seconds = 30; Score = 0;
    setText("Time", seconds);
    setText("Score", Score);
    Time();
});
onEvent("btnPause", "click", function(event) {
    setScreen("PauseScreen");
    clearInterval(myInterval);
});
onEvent("btnResume", "click", function(event) {
    setScreen("playScreen");
    Time();
});
onEvent("btnRestart1", "click", function(event) {
    setScreen("playScreen");
    seconds = 30;
    Score = 0;
    setText("Time", seconds);
    setText("Score", Score);
    Time();
});
```

I used this algorithm in order to set the code in an organized way, such that it would work efficiently and when a certain button is pressed, it would execute the code written for the button in the event handler. For example when the event handler "Catimage" was called, it would execute the code in that event handler when the Nyan Cat image was clicked, then adding adding 1 point to the score and setting the text of the score to update the list. This is the same for the other algorithm as they both use an index." The response also describes what the purpose is for this algorithm in relation to the entire program: "allows the user to go to the next flashcard" and "the user has a study guide environment in which they can type in the necessary term to the definition and then be able to go to the next set of terms."

---

**Scoring Guidelines**

<table>
<thead>
<tr>
<th>Row 4</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>The response earned a point for this row. The code given in the response is an algorithm because it involves sequencing, selection and/or iteration.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Row 5</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>The response DID NOT earn a point for this row. Although the algorithm given includes math minimally (incrementing the score) and it explains how the functions work, it does not describe what this algorithm does in relation to the overall program.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Row 6</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>The response DID NOT earn a point for this row. The response does not clearly identify two algorithms used by the selected algorithm.</td>
<td></td>
</tr>
</tbody>
</table>
everytime the Nyan Cat was clicked such that the score would keep increasing. Something else to notice is the Time function being called in several event handlers. This was done in order to set the timer to restart or to start again depending on the event handler that is executed.

Student Response G - [Video] [Written Response]

The purpose of this algorithm is to make the apple appear in a new place. Each time the snake gets an apple, it reappears somewhere else. When the snake touches the sprite that is the apple, the random number function makes it reappear anywhere between x: -230 to 230 and y: -160 to 160. This also adds to the difficulty of the game because the snake has to avoid the edges or the game will end.

Scoring Guidelines

Row 4

The response earned a point for this row. The selected block of code represents an algorithm.

Row 5

The response earned a point for this row. The included code segment contains math/logic by adding the score to 1. The response explains how the algorithm works: "When the snake touches the sprite that is the apple, the random number function makes it reappear anywhere between x: -230 to 230 and y: -160 to 160." The response describes what this algorithm does in relation to the program: "to make the apple appear in a new place."

Row 6

The response DID NOT earn a point for this row. The response does not identify two algorithms that are used within the selected algorithm nor how one of them works.

Student Response H - [Video] [Written Response]

The two algorithms within the selected algorithm are the onEvent function (to move on to the next screen) and the if…else…. statement. The response identifies a segment of code that is an algorithm.

Scoring Guidelines

Row 4

The response earned a point for this row. The response identifies a segment of code that is an algorithm.

Row 5

The response earned a point for this row. The response uses math and/or logic (an if statement). The response states what the algorithm does in relation to the program ("used when the user must type in a response"). The response explains how the algorithm works ("There is only one correct answer, in this case it is horse. If the user types in horse exactly then they got it right and they move on; anything other than horse will be counted as wrong and the user will have to restart.").

Row 6

The response DID NOT earn a point for this row. The response identifies the event handler as the first algorithm, and the if/else statement as a second algorithm. This does not meet the requirements for this row. The response needs to identify an algorithm that uses two additional algorithms that function independently.
the user to continue the quiz, or go to the game over screen when the user has to restart. The second algorithm uses logical concepts and allows for the user’s input to be checked and verified by using an if/else statement. There is only one correct answer, in this case it is horse. If the user types in horse exactly then they got it right and they move on; anything other than horse will be counted as wrong and the user will have to restart. Another small part of the algorithm is that everytime the check button is clicked, the input text box is reset to be blank for the next time around. The two algorithms together make the program run smoothly as it is supposed to.

There is only one correct answer, in this case it is horse. If the user types in horse exactly then they got it right and they move on; anything other than horse will be counted as wrong and the user will have to restart.

Another small part of the algorithm is that every time the check button is clicked, the input text box is reset to be blank for the next time around. The two algorithms together make the program run smoothly as it is supposed to.

The red oval in the image captures the algorithm which is essential for the function of the program. Within it there is an algorithm in lines 9-10, which takes the value given to countdown by the user from the dropdown textbox and subtracts one from it and then displays Create Sample I 1 of 2 the result on the countdown slot. This algorithm was developed independently. The second algorithm begins in code line 7 as setInterval, and its value is displayed in line 18 of the code as 1000 milliseconds. The value of the setInterval was also developed independently. Both algorithms are essential because in unison they allow the program to work, as without the first algorithm the code would not work as nothing would be displayed nor nothing would be subtracted so the function would eventually reach zero. While without the second algorithm result would always be one less then the value set for countdown and there be no command telling it to repeat the process in algorithm number one.

The response earned a point for this row. The selected block of code represents an algorithm.

The response earned a point for this row. The circled code contains math/logic. The response explains how the algorithm works in detail. The response describes what this algorithm does in relation to the program: "Both algorithms are essential because in unison they allow the program to work" AND then states what would happen to the program if either part is missing. By explaining what happens if each part is missing, the response is explaining what these algorithms do for the whole program.

The response DID NOT earn a point for this row. The code given in the response is not considered an algorithm because each event handler consists of a single
Within the code, the algorithm is essential to the program by each sport having its own command or action. It controls which sport you want to go to. The code represents the amount of time it takes you to answer a question, taking too long might make the game go back to the welcome screen so there's limited time to play. Picking the right answer will get you to the next question automatically. From there it's just repetitive for the rest of the game, for all sports. All buttons are on click actions, as well, just know whatever sport you pick you must have a little background knowledge on it.