Video ​Submit one video in .mp4, .wmv, .avi, or .mov format that demonstrates the running of at least one

significant feature of your program. Your video must not exceed 1 minute in length and must not exceed 30MB

in size

Prompt 2a. ​Provide a written response or audio narration in your video that:

● identifies the programming language;

● identifies the purpose of your program; and

● explains what the video illustrates.

(Must not exceed 150 words)

Advice: ​For resources on how to make your video head to https://studio.code.org/s/csp-create/stage/1/puzzle/2. Here’s

the most important things to remember for your video and prompt 2a.

● Video Runs Continuously: ​Your video must run continuously and show your actual code running. It can’t just

be a series of screenshots.

● Show One Feature: ​Your program does NOT need to be complete so long as you can demonstrate one major

feature that’s running.

● Describe the Purpose: ​The purpose of your program is the intended goal or objective of the program. In other

words, it’s “what” the program is supposed to do. If you made a game, an app, or some other kind of project,

just quickly describe “what” kind of program it is and how it would be used / played.

● Connection to Video: ​Make sure that you can connect the purpose of your program to what is shown in the

video. If you only have one feature working then describe the purpose of the feature.

Sentence Starters

● My program is written in JavaScript using the App Lab programming environment.

● The purpose of my program is... (describe what it was meant to do)

● In the video you can see (how a piece of functionality works that’s directly tied to the purpose).

Draft Your Response Here:

(must not exceed 150 words)

Response 2a Checklist

Video

❏ Video runs continuously (it cannot be a series of screenshots)

❏ Video is less than 60 seconds long and less than 30MB in size

❏ Video demonstrates one running feature of the program

Written Response

❏ Response identifies the programming language used

❏ Identifies the purpose of the program

❏ Describes the feature(s) shown in the video and their connection to the purpose of the program

❏ May be audio commentary in your video. Carefully follow this checklist even if you use audio commentary.

14

2b. ​Describe the incremental and iterative development process of your program, focusing on two distinct

points in that process. Describe the difficulties and / or opportunities you encountered and how they were

resolved or incorporated. In your description clearly indicate whether the development described was

collaborative or independent. At least one of these points must refer to independent program development.

(Must not exceed 200 words)

Advice: ​There are many individual pieces of information you need to fit in this response. Use the checklist at the

bottom carefully to make sure you don’t miss any. Here’s the most important pieces to remember:

● Don’t Forget the Overall Process: ​Your response should reference your development process as a whole,

NOT just the two points in time. Your response should reflect an iterative process of identifying and solving

problems.

● Difficulties / Opportunities: ​You need to describe two distinct difficulties / opportunities you encountered

○ A difficulty​ is likely either a bug in your code or a difficult design problem you needed to work out.

○ An opportunity​ is likely an idea or realization you had as you developed your code.

● Feedback / Testing / Reflection: ​ You should clearly describe HOW you identified the two difficulties /

opportunities - was it through testing your program out? Personal reflection? Through feedback from a peer?

● Independent or Collaborative: ​If you developed your program completely independently, you need to say so -

don’t assume the reader will know. If you developed your program collaboratively, some parts need to be done

on your own. For this response make sure:

○ You clearly indicate which parts were done collaboratively

○ You clearly state which of the difficulties/opportunities described here was done independently on your

own (could be both, but at least one).

Sentence Starters

● I completed this project (independently / collaboratively).

● I began developing my program by (describe beginning of process, what did you do / decide first)

● Then I worked iteratively to (how you continued building your project)

● Early on through (feedback / testing / reflection) I identified a (problem / opportunity) which was ...

● I solved this problem (independently / with my partner) by ...

Draft Your Response Here:

(must not exceed 200 words)

Response 2b Checklist

Overall Development

❏ Response describes the overall development process, not only two key points.

❏ Response indicates whether you completed the project independently or with a partner. (note: this indication can

be incorporated throughout your response and in comments within your code as well).

First Difficulty / Opportunity

❏ Response describes one difficulty / opportunity encountered early in the development process

❏ Response describes source of difficulty / opportunity as either feedback, testing, or reflection

❏ Response indicates how it was incorporated / solved, including whether you wrote the code independently.

Second Difficulty / Opportunity

❏ Response describes one difficulty / opportunity encountered later in the development process

❏ Response describes source of difficulty / opportunity as either feedback, testing, or reflection

❏ Response indicates how it was incorporated / solved, including whether you wrote the code independently.

15

❏ If first Difficulty / Opportunity WAS NOT solved independently, then this one must be

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 words)

Advice: ​Review the “Is It a Good Algorithm” section above for lots of helpful tips on how to choose your algorithm.

Here’s the most important points.

● You Wrote It: ​You need to have written the code of your algorithm entirely on your own (not with a partner)

● Copy and Paste It: ​You must paste your actual algorithm code as part of this response.

● A Parent and Two Children: ​Your main algorithm (the “parent”) needs to have two sub-algorithms (the

“children”). See Example Algorithms 4 and 5 for ideas on how this might look.

● Mathematical / Logic Concepts: ​At least one sub-algorithm needs to use mathematical and/or logical

concepts.

● Break Into Functions: ​To make it easier to refer to individual parts of your algorithm give the parent and child

algorithms their own functions. Example Algorithm 4 is written in this way.

● Describe “how”, not just “what”: ​You need to talk about how your code works, not just what the user will see

when it runs. Do this by referring to the actual variables names, programming constructs, strings, and so on,

that are visible in your code snippet. For example:

“The algorithm I selected is signInUser() which handles the user login process in my app

which has two key parts: checkName() and startHomeScreen(). checkName has an

if-statement that checks to see whether the name entered in the usernameTxt textbox is

equal to ‘MrSillyMan’ or ‘MsFunnyGal’. If it is then it sets the accessGranted variable to

true, otherwise false. The startHomeScreen() function checks the accessGranted variable

and returns the login screen if false, otherwise it proceeds to show the home screen for

the user.”

Sentence Starters

● The main algorithm that I selected is (main-algorithm name).

● This algorithm has two key parts, (sub-algorithm 1) and (sub-algorithm 2).

● (Sub-algorithm 1) is designed to (what it does). It does this by (how the code of sub-algorithm 1 works).

● My (main-algorithm) combines (sub-algorithm 1) and (sub-algorithm 2) to (what main-algorithm does).

● Together these algorithms help achieve the purpose of my program by (how these algorithms are tied to

program purpose).

Draft Your Response Here:

[don’t forget - paste the algorithm code snippet here - the same one you put an oval around in the whole program code]

[write your response]

(must not exceed 250 words)

Response 2c Checklist

Overall

❏ You wrote all algorithm code yourself

16

❏ Response includes copy-pasted versions of code for main and sub-algorithms with ovals around them

❏ Response identifies the main algorithm and at least two sub-algorithms

Sub-algorithm 1

❏ Clearly identifies the code for the algorithm (where in the code, function name, line numbers, etc)

❏ Explains what the algorithm does independently

❏ Describes how the code of the algorithm works

❏ Uses mathematical or logical concepts

Sub-algorithm 2

❏ Clearly identifies the code for the algorithm (where in the code, function name, line numbers, etc).

❏ Explains what the algorithm does independently

❏ Describes how the code of the algorithm works

❏ Uses mathematical or logical concepts

Main Algorithm

❏ Clearly identifies the code for the algorithm (where in the code, function name, line numbers, etc).

❏ Describes how main algorithm combines sub-algorithms

❏ Explains how main algorithm helps to achieve the overall purpose of the program

17

2d.​ Capture and paste a program code segment that contains an abstraction you developed individually on

your own (marked with a rectangle​ in section 3 ​below). This abstraction must integrate mathematical and

logical concepts. Explain how your abstraction helped manage the complexity of your program. (Must not

exceed 200 words)

Advice: ​Review the “Is It a Good Abstraction?” section above for tips on how to choose your abstraction. Here’s the

most important points.

● Choose a Function: ​Unless you feel confident about another abstraction, choose a function - not an onEvent,

but a function you defined and named yourself.

● You Wrote It: ​You need to have written the code of your abstraction entirely on your own (not with a partner)

● Copy and Paste It: ​You must paste your actual abstraction code as part of this question submission.

● Manages Complexity: ​Make sure you can describe how your abstraction helps manage complexity in your

program.

● Make a contrasting argument: ​explain how your program would be more complex to read, write, or reason

about if you had not created your abstraction.

● Mathematical and Logical Concepts: ​While you should aim to include these in your abstraction, this is NOT

explicitly assessed by the Scoring Guidelines for 2018.

Sentence Starters

● The abstraction I selected is (name of abstraction)

● My abstraction manages complexity in my program by...

● Without my abstraction my program would be more difficult to (read / write / understand) because...

Draft Your Response Here:

[Don’t forget - paste your abstraction code snippet here - the same one you put a rectangle around in the program

code]

[Write your response]

(must not exceed 250 words)

Response 2d Checklist

Overall

❏ You wrote all abstraction code yourself (it’s not an onEvent block, but a function you defined and named)

❏ Response includes copy-pasted versions of code for abstraction with a rectangle around it

❏ Response identifies the abstraction by name

❏ You explicitly describe HOW the abstraction manages complexity (e.g. by explaining how your code would be

more complex to write or reason about without the abstraction)

18

3. Program Code

Capture and paste your entire program code in this section.

› Mark with an oval the segment of program code that implements the algorithm you created for your

program that integrates other algorithms and integrates mathematical and/or logical concepts.

› Mark with a rectangle the segment of program code that represents an abstraction you developed.

› Include comments or acknowledgments for program code that has been written by someone else.

Advice: ​For resources on how to make a PDF of your program code head to

https://studio.code.org/s/csp-create/stage/1/puzzle/2. Here’s the most important things to remember:

● Making Your PDF: ​Use CodePrint to make a PDF of your program. It’s designed specifically for the Create PT.

You can find it from the link given above.

● Marking Your Algorithm: ​Make sure you place an oval around all parts of your algorithm (parent and

children).

● Marking Your Abstraction: ​Place your rectangle around the code where you create your abstraction (e.g.

define a function) not where you use the abstraction (e.g. call a function).

● Commenting and Collaboration: ​You may work with a partner on the Create PT, but you must clearly indicate

which parts you completed independently and which you completed together by using comments. For example:

// I completed the section below with my collaborative partner

// I completed this section independently

// I have extended code found at [URL]. The code below is my additions

Remember that your algorithm and abstraction need to be created entirely independently.

● Citing Images: ​If you use code or images made by someone else (for example that you found online) then you

should cite those resources as well. Again you can use comments.

// The images used in this app came from:

// [1] bird image - http://name-of-site.com/path/to/image.jpg

// [2] flower image - http://site.com/path/to/flower.jpg