Cognitive Walkthrough Example Form

Briefly describe the system being evaluated:

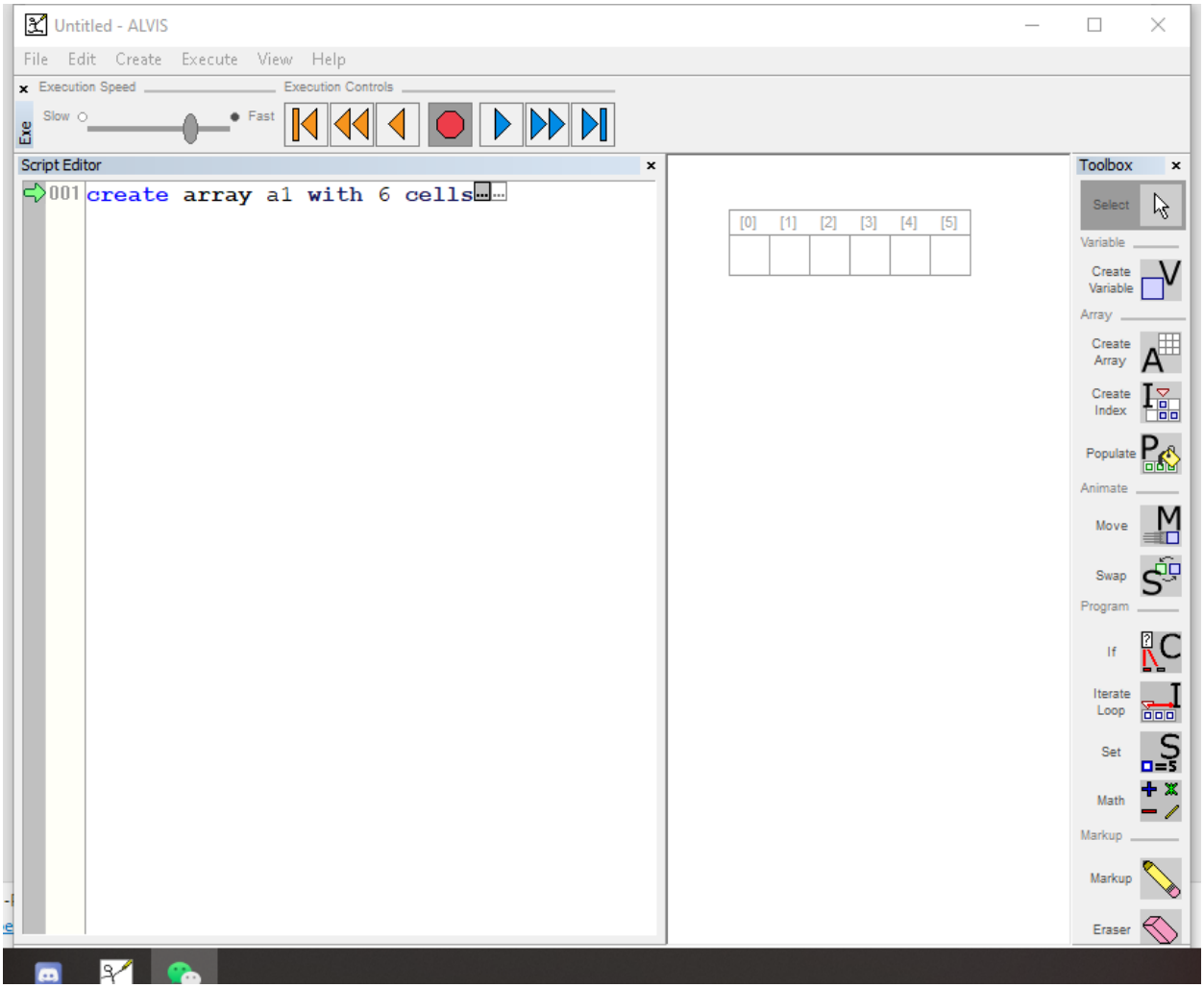
The Algorithm Visualization Storyboarder (ALVIS)

Briefly describe the target users of this system (background, experience, etc.):

First-semester computer science students who are learning to program algorithms.
Briefly describe the task(s) to be evaluated:

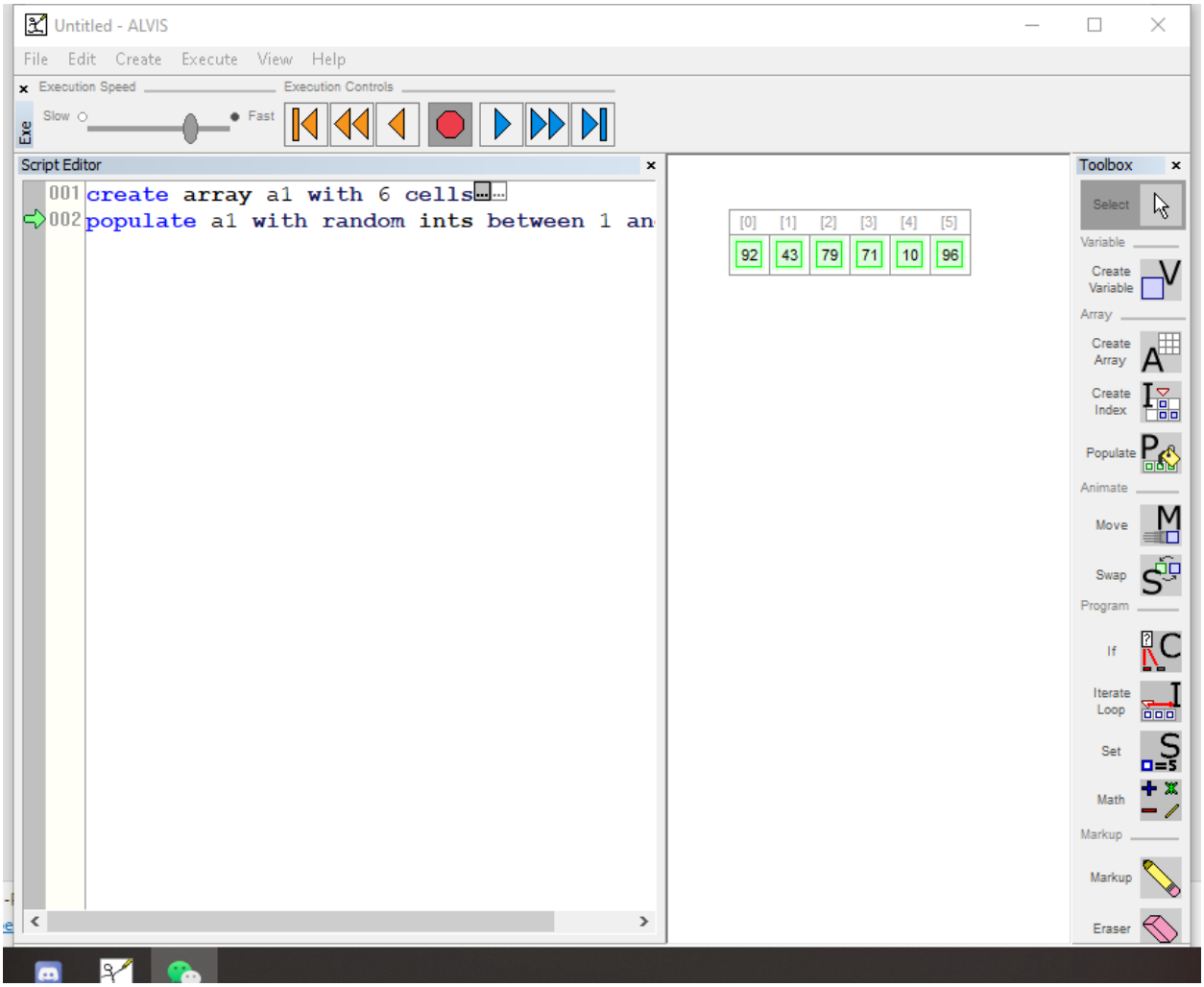
Task 1: Create array

Task Steps for Task 1	**Will the user know what to do next to make progress?**	**Will the user notice how to perform the correct action?**	**Will the user interpret the system response correctly?**
1.2 Click and Drag out an array	The pop-up box that appears in previous step reads "click in the animation window and drag the mouse to create and size an array." (After clicking, the mouse will change, but you need to prompt the user to drag and drop to create the size you want.) The user will therefore have an idea what to do next.	This is questionable, since the instructions are actually not quite right. A more precise instruction would be "Position the mouse at location where you'd like the array, press and hold the left mouse button, drag out an array, and finally release the mouse button." Further, the user may have trouble knowing what the "animation window" is, since it lacks a signifier.	The user will see the array appear, along with a "create array" statement. This will likely serve as confirmation that the array was created. However, the array may not be the correct size (6), which will require corrective action on the part of the user (see next step)



Task 2: Populate array

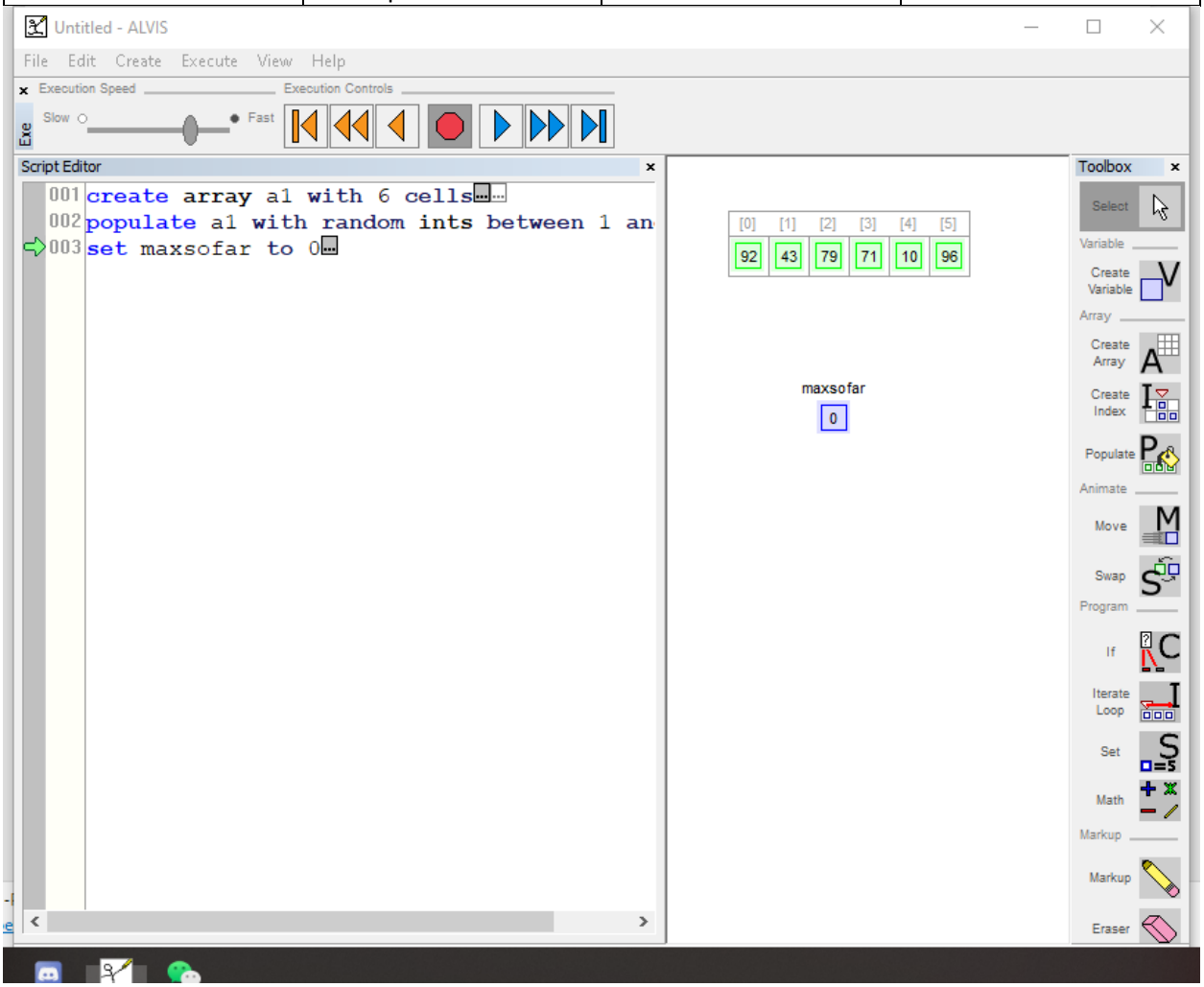
Task Steps for Task 2	Will the user know what to do next to make progress?	Will the user notice how to perform the correct action?	Will the user interpret the system response correctly?
2.1 Click the populate tool	The pop-up box that appears in previous step reads “Populate an array with variables by clicking on it.” (After clicking, the mouse will change, and it will take effect only when the mouse is placed in the list.) The user will therefore have an idea what to do next.	The correct description should be "After clicking the Populate button, select the function, and then click the existing array list to be filled to realize the function". Therefore, users may need to think and try where to click.	After clicking, the random variables will be directly displayed in the array list, and there will be corresponding declarations in the left window. If necessary, the range of the random variable can be set later.



Task 3: Set variable “maxsofar”

Task Steps for Task 3	Will the user know what to do next to make progress?	Will the user notice how to perform the correct action?	Will the user interpret the system response correctly?
3.1 Click and select a location to create a variable	The pop-up box that appeared in the previous step is displayed as "Click in the animation window, and a variable will be created at the location". (After clicking, the mouse will change, indicating that the operation is being performed.) Therefore,	This operation is relatively simple, and it is easy to understand if there is a basis for creating an array list. When you move the mouse to the created variable, "Double-click to modify" will be displayed, prompting the user to double-click to edit.	On the left, there is a declaration of "created variable and what is the value of the variable", indicating that the creation was successful.

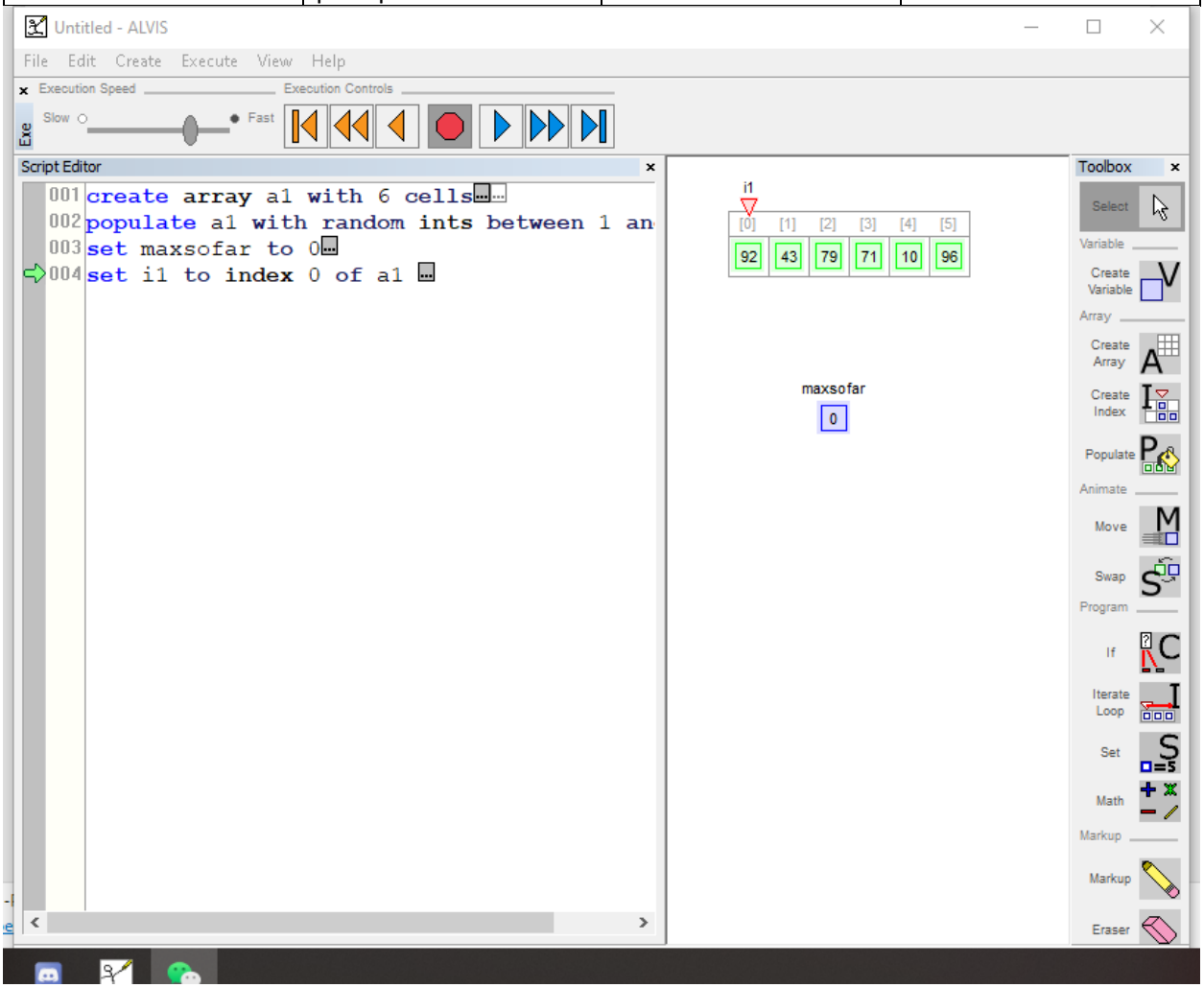
	the user will know the next operation.		



Task 4: Create array index

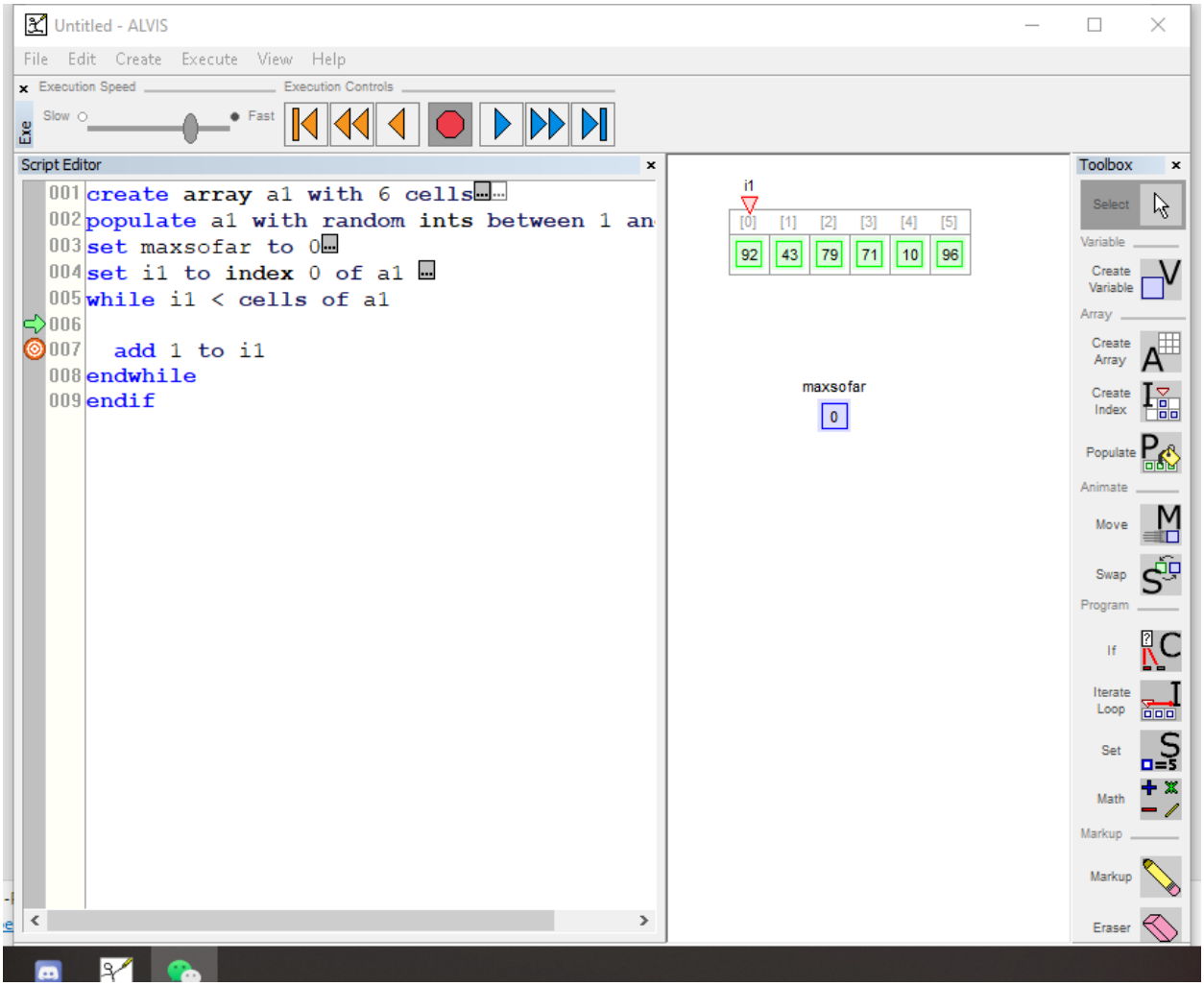
Task Steps for Task 4	**Will the user know what to do next to make progress?**	**Will the user notice how to perform the correct action?**	**Will the user interpret the system response correctly?**
4.1 Choose “Create Index” tool	The pop-up box that appeared in the previous step is displayed as "Double-click to set tool properties" (after clicking, the mouse will change, and it will only take effect when the mouse is placed in the	It is best to have a more detailed introduction, such as "index needs to be placed on the variable"。	The added index will be displayed on the left. (No follow-up operation will not see any specific effects.)

	variable of the array list.) But what to do afterwards is not prompt		



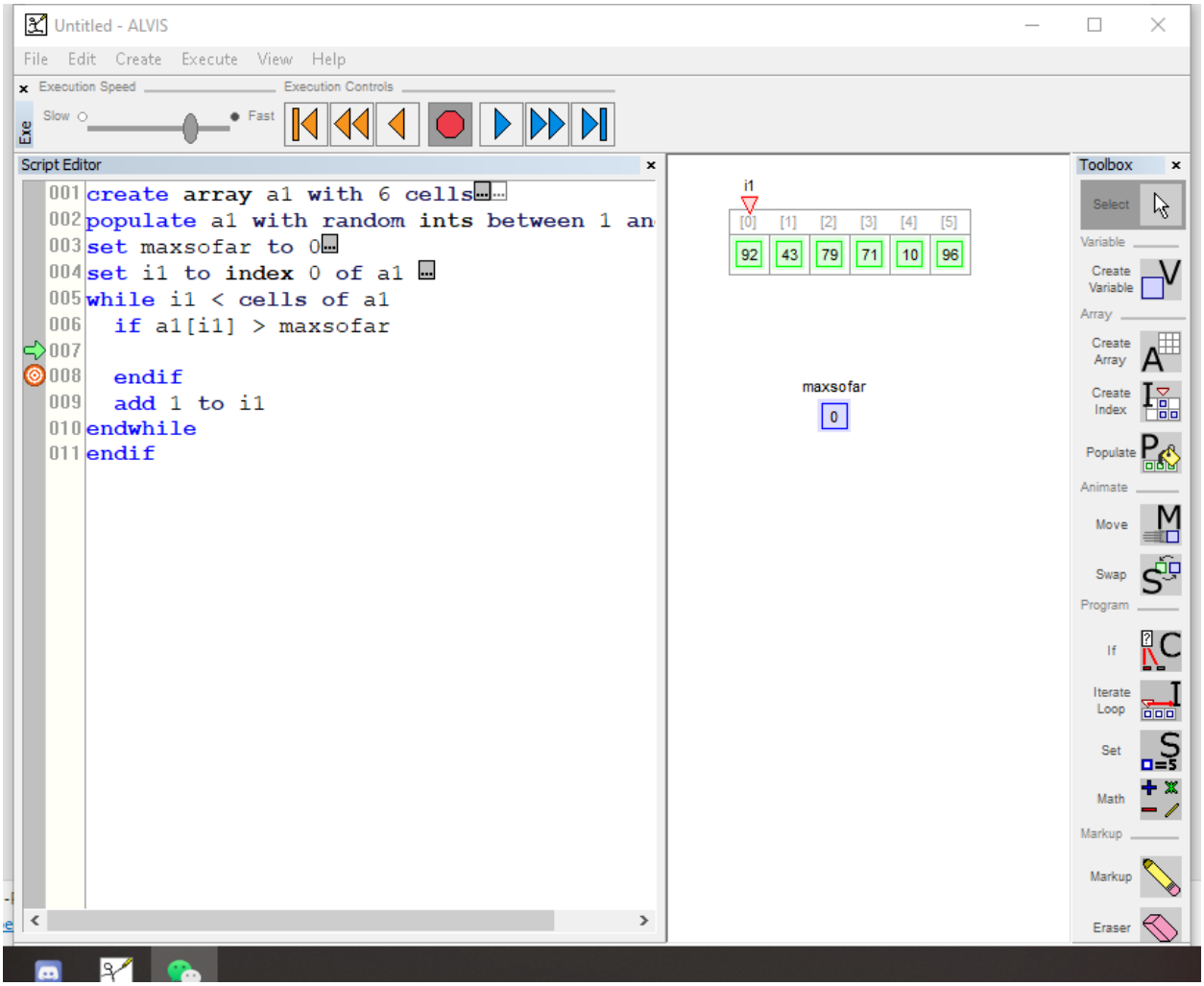
Task 5: Create loop of index

Task Steps for Task 5	Will the user know what to do next to make progress?	Will the user notice how to perform the correct action?	Will the user interpret the system response correctly?
5.1 Choose “Iterate Loop” tool	The pop-up box that appears in previous step reads “Click on an index, and drag it to the last cell of iteration.” The user will therefore have an idea what to do next.	Some wrong choices cannot be executed, the starting point is fixed, and the ending point is easy to judge	The statement on the left indicates whether the loop creation is successful.



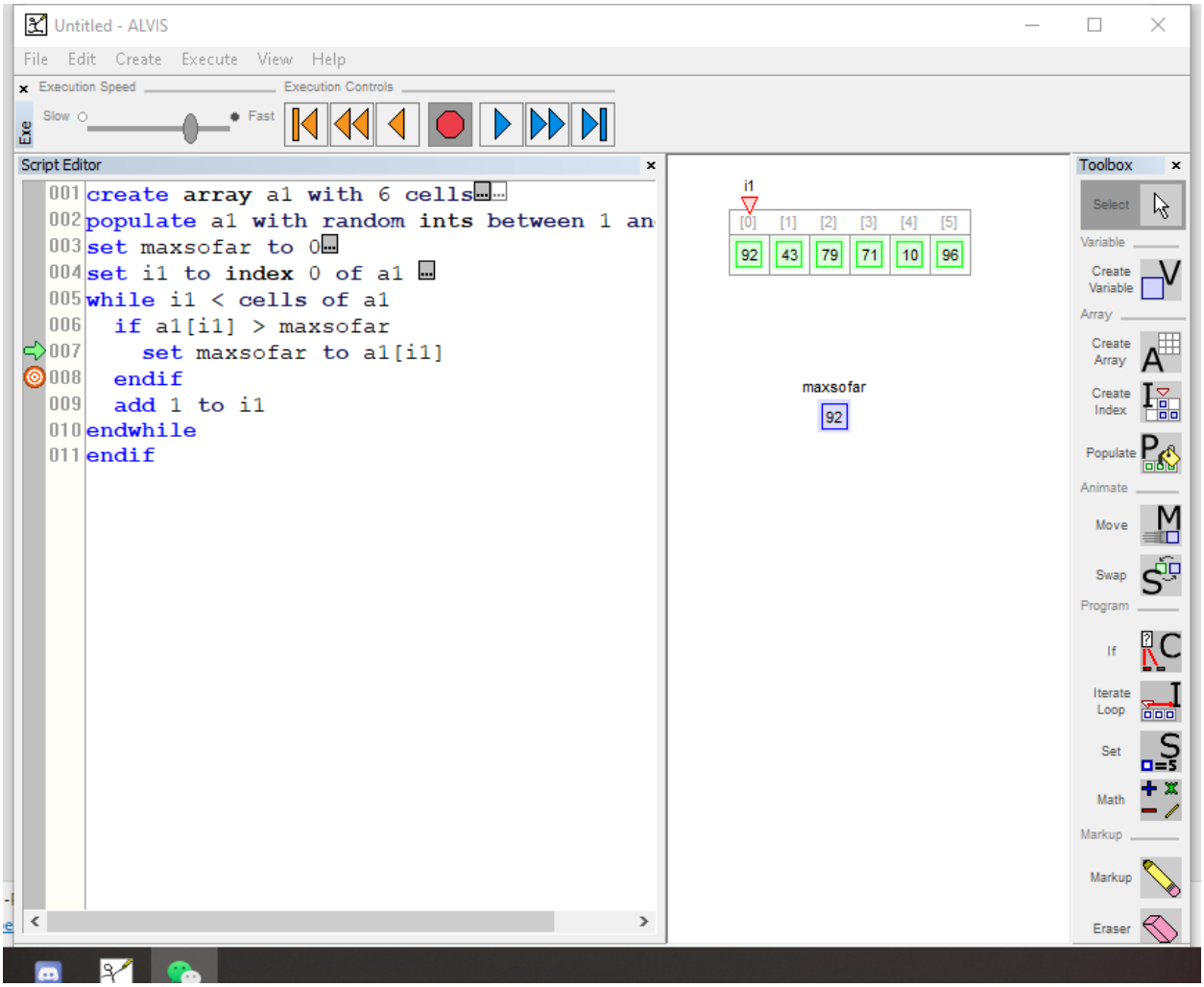
Task 6: Set If else

Task Steps for Task 6	Will the user know what to do next to make progress?	Will the user notice how to perform the correct action?	Will the user interpret the system response correctly?
6.1 Choose “If” tool	The pop-up box that appears in previous step reads “Chick on a variable or array element to place on the left-hand side of the if statement.” The user will therefore have an idea what to do next.	Each step is explained, so it is not difficult to execute.	The left side will show that the statement was created successfully



Task 7: Set variable

Task Steps for Task 6	Will the user know what to do next to make progress?	Will the user notice how to perform the correct action?	Will the user interpret the system response correctly?
7.1 Choose “Set” tool	The pop-up box that appears in previous step reads “Click on a variable or array element whose value is to be set.” The user will therefore have an idea what to do next.	The variables in the list have multiple choices and are not very exhaustive. It is easy for users to make mistakes and fail to get the desired results.	If user can operate it correctly, they can get the desired result.



Discussion of successes and failures:

Successes:

- If the steps are correct, or the method of use is mastered, this design can achieve its purpose through these tools.
- The code on the left allows the user to know whether the drawing is correct.

Failures:

- Users need to master certain knowledge, and the tool is not easy to operate without being familiar with it.

Design suggestions:

No shortcut keys are currently found, which makes drawing too much dependent on the mouse. It can add related instructional videos to make it easier for users to get started. And if the process is accidentally reversed, accurate error reporting cannot be provided.