

1.5.3 Notes

How do programs usually respond to user input?

Usually the user input is a mouse click, a touch on a screen, a key being pressed or voice recognition. The way the programs usually respond is based on how the program is written. For instance, when an app is touched on the homescreen, it will either bring up more options, or it will open the app.

How do programmers make the interface behave the way a user expects it to? How do software developers design a solution to a problem so that it can be reused for other, similar problems?

Programmers could conduct surveys with potential clients and users and then they could use the input they receive and implement it into their GUI. They would have to do this through their code. Software developers should usually make their code, generic so that you only need to change a variable in one spot, so others can easily use it. Also, leaving comments explaining what your code does helps too.

What is a solution you developed to a real-life problem that you were able to reuse in a different situation?

One solution I had was to a problem I had on my math tests. I usually didn't do too good on my tests, but then I started studying harder, and my scores improved. My studying strategy then carried over into my other subjects and my grades went up.

Consider a grade book system where parents, teachers, and students are emailed if a student misses an assignment. What is the event? What classes of objects might be subscribing to that event?

The event is the email. When the student is missing his assignment, his parents and then the student get an email. They "subscribe" to the email.

Times for Sorting (in seconds)

Insertion	Selection	Bubble	Shell	Merge	Heap	Quick	Quick 3
5.86	10.98	8.75	4.59	5.32	4.72	4.53	5.29

The fastest sorting algorithm for a random list was Quick 3.

Quicksort 3 works by dividing a list into 3 smaller pieces, that it can then quickly sort.

What is another instance of the searching-a-sorted-list problem?

When browsing through the school website, in the faculty list, all the faculty are listed in alphabetical order.

Radius Changer: The radius changer takes input from a slider and then draws a circle and changes the circles size based on the number from the slider, which represents radius.

Color changer: The color value represented by ' #A01145 ' is 160 red, 17 green and 69 blue.

Steps: 1. Create a canvas

2. Create a circle object

3. Change the color of the circle object in based on the color code sliders

I completes this challenge by first creating a canvas in the middle of the screen. Then, I had to create a circle object, so i copied the code from the earlier challenge. Lastly i had to figure out a way to change the color based on the sliders. To do this, I used the result from the sliders plus a '00' at the end. This gave me the correct color code. But there was still a problem. The circle just stayed the color of the sliders at the beginning and didn't change.

To fix this, I put the code for creating the circle inside the function that prints the color codes.

21A) The scope of `new_shape` is local to the function up. It is implied that `new_shape` will only be used within this function.

21B) The global scope is only needed in one function, because once something is global, all functions can access it.

To create the adaptation of the circle art program, I decided to create a button that can clear the canvas. To do this correctly, instead of placing the button on the root, I had to place it on the editor button. When it is pressed it calls the function callback, which then clears the canvas with the function `canvas.delete ("All")`.

23) The mirror man image stops once the man holding the mirror is too small to see inside another mirror.

Bouncing Ball

Our change to this program was to add another slider so you can adjust the radius of the ball while it is bouncing across the screen. We had difficulty with placing the slider in the correct spot because originally it was placed on top of the speed slider. To fix this we just had to change the column the radius slider was in and then the program worked.

Conclusion:

- 1) The difference between recursion and iteration is that a recursive function calls itself until a condition is met. While iteration uses a loop to complete multiple instructions a set number of times.
- 2) I will ask Mr. greenway... i don't understand the question and the lesson did not explain it.
- 3) To design a GUI, first you have to create a plan of how the GUI should look. Then you could create a data map in order to show how the GUI should work. Lastly, you need to code the GUI.

All the code is on github....

