

## Step 3: Conditional Behavior

---

 [coursera.org/learn/writing-running-fixing-code/supplement/XYR3z/step-3-conditional-behavior](https://coursera.org/learn/writing-running-fixing-code/supplement/XYR3z/step-3-conditional-behavior)

### Generalizing Conditional Behavior

---

Sometimes when we are generalizing, we will have steps which appear sometimes, but not others. Such a situation may be a matter in which we perform a step for some parameter values, but not for others; or in which we have steps that are almost repetitive, but some actions which appear in some repetitions but not in others. In either case, we need to figure out under what conditions we should do those steps.

It may take some work and thinking to determine the patterns for what conditions we need to perform those steps under, and what conditions we do not. As with many things in generalizing, if it is not immediately apparent, it can be quite useful to work more examples—giving you more information to generalize from. You might also find it informative to make a table of the circumstances (parameter values, information specific to each repetition, etc.) and whether or not the steps are done under those circumstances.

Once you have figured out the pattern, you can express the step in the algorithm more generally by describing the condition that should be determined, and what to do if that condition is true, and what to do if it is false. Doing so makes your algorithm a little bit more general, and may help you express a large sequence of steps as repetition, since they will now be more uniform.

### Generalizing Is an Iterative Process

---

Generalization is an iterative process—you take what you have, generalize (or rewrite it) a bit, and then try to generalize that result more. Sometimes one step of generalization opens up new avenues of generalization that were not visible before. We have already seen how recognizing repetitive patterns can lead to the opportunity to generalize in terms of how many times you do the repeated steps. You may also end up exposing the repetitive pattern of some steps only once you have figured out what the generalization of the values in those steps is.