

Conditionals Questions

- Does something happen differently at different times, or only happen sometimes?
→ if so, need a conditional (if statement)
- Imagine that you only see a snapshot of the sketch, frozen at a moment in time – can you determine what should happen next? i.e. can you use the current values of system variables and/or animation variables to make the decision?
 - yes → on-the-spot decision
 - no, it depends on prior events / what has been going on → state machine
- For on-the-spot decisions –
 - How many alternatives are there for what happens?
 - “to do or not to do” (do something or do nothing) → if
 - two alternatives → if / else
 - more than two alternatives
 - “do nothing” is not an option → if / else if / ... / else
 - “do nothing” is an option → if / else if / ...
 - What are the alternatives? → body of each part
 - How do you decide which alternative to do? → boolean condition for each if / else if part
- For state machines – associate each alternative with a value (called a *state*) and use a *state variable* to keep track of that value
 - How many alternatives are there for what happens? → type for state variable
 - two alternatives (“do nothing” may be one) → boolean type
 - more than two alternatives (“do nothing” may be one) → int type
 - What are the alternatives for? → name for state variable
 - What are the alternatives?
 - decide on which value of the state variable corresponds to which alternative → variable comment
 - Which alternative do we start with? → initialization
 - usage – if/else if/.../else with a case for each alternative, using the state variable in the conditions
 - For each state, what triggers a change to that state? → conditions in if/else if/.../else to update the state variable

Syntax Recap

- if statement – do something or do nothing

```
if ( boolean condition ) {
    statements           // do these only if the condition is true
}
```
- if statement with else – do one of two alternatives

```
if ( boolean condition ) {
    statements           // do these only if the condition is true
} else {
    statements           // do these only if the condition is false
}
```

- else if – choose from more than two alternatives
 - any number of else ifs can be included
 - include final else if “do nothing” is not an option

```

if ( boolean condition ) {
    statements           // do these only if the condition is true
} else if ( boolean condition ) {
    statements           // do these only if all previous conditions
                          // are false and this condition is true
} else {
    statements           // do these only if all of the conditions
                          // are false
}

```

- relational operators – to compare numerical values: <, >, <=, >=, == (equal to), != (not equal to)
- logical operators – to combine boolean values: && (and – both conditions must be true at the same time), || (or – only one condition needs to be true), ! (not)

At the End of Class

- Make sure each sketch has a comment with the names of your group, and that you have saved your sketches.
- Use FileZilla to copy the entire folder for each of your sketches into your handin directory (**/classes/cs120/handin/username**). You only need to hand in one copy for the group.

Exercises

1. Create a sketch named **sketch_240925a** and copy-and-paste or type the contents of the “for #1” sketch posted on the schedule page into your sketch, then modify it so that the rectangle is red when the mouse is over it and blue otherwise. (To be over the rectangle, the mouse position has to be simultaneously right of the rectangle’s left side, left of the rectangle’s right side, above the rectangle’s bottom, and below the rectangle’s top.) Hint: this is an on-the-spot decision.
2. Create a sketch named **sketch_240925b** and copy-and-paste or type the contents of the “for #2” sketch posted on the schedule page into your sketch, then modify it so that clicking the mouse pauses and unpauses the animation. (i.e. sometimes the rectangle grows and sometimes it doesn’t) Hint: this is a state machine.
3. Save a copy of your sketch from #2 as **sketch_240925c** and modify the copy so that when the rectangle reaches the sides of the window, its size resets to 0. (Otherwise it grows.) Is this an on-the-spot decision or a state machine?

If you have time –

4. Save a copy of your sketch from #2 as **sketch_240925d** and modify the copy so that when the rectangle reaches the sides of the window, the width stops increasing but the height starts to grow instead. Once the rectangle reaches the top/bottom of the window, both width and height reset to 0 and the animation repeats. Is this an on-the-spot decision or a state machine?
5. Modify the previous sketch so that the animation resets to 0 three times, but then stops. Hint: treat the number of repetitions completed so far as (another) state variable.