Making Choices

Making Choices

Conditionals questions –

- Does something happen differently at different times, or only happen sometimes?

  → **need conditionals**

- Imagine that you see only 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 the 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**

On-the-Spot Decisions

> can decide what to do from where the sketch is at the moment

Questions –

- 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? → condition for each if / else if part
  - this is a boolean expression in terms of the animation and system variables already in your sketch

---

Syntax

- if statement – "to do or not to do" (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
}
```

boolean condition is an expression that evaluates to true or false – typically a comparison involving ==, !=, <, <=, >, >=

statements are any legal Processing statements e.g. drawing commands, function calls, animation variable updates, conditionals, …

Syntax

- else if – choose from more than two alternatives (do nothing is not an option)
  - any number of else ifs can be included

```
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
}
```

boolean condition is an expression that evaluates to true or false – typically a comparison involving ==, !=, <, <=, >, >=

statements are any legal Processing statements e.g. drawing commands, function calls, animation variable updates, conditionals, …

Syntax

- else if – choose from more than two alternatives (do nothing is an option)
  - any number of else ifs can be included

```
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
}
```

boolean condition is an expression that evaluates to true or false – typically a comparison involving ==, !=, <, <=, >, >=

statements are any legal Processing statements e.g. drawing commands, function calls, animation variable updates, conditionals, …