Conditionals:
More Fun with “If-Then” and Logic Operators
(Project 5)
Goals

• Great job so far! There are a lot of things you already know how to do! In this tutorial and in the following tutorials you will get a chance to start using what you have learned. Some familiar steps will have less detailed instructions. Look at hints or at previous tutorials if you need help, or ask a friend!

• In this tutorial you will:
  − Review: How your program controls the SPHERES satellites
  − **Practice** programming with “If_Then” statements
  − Use the logic operators “==” and “!=
  − Learn about the “debug” feature
• The program inside your loop is called once per second by the SPHERES control system.

• During each second:

```c
1. void loop() {
2. }
```
Create A New Project and Declare Variables

- Let’s create a new “if-then” project to learn more about the SPHERES Control System
- Open the IDE
- Select “New Project”
  - Project name: **Project5**
  - Editor: Graphical Editor
  - Game: FreeMode
- Open the **Init** page
- Create a variable (pink block) called “counter”
  - Type: int
  - Name: counter
  - Initial value: 0
- Go back to the **Main** page
If-Then with the Logic Operator “==”

- “==” is a Logic operator that means “equals”
- Create the following “if-then” statement in your loop:
  
  If counter == 10 then . . .

(See example to the right)

- Here are some hints:
  - Drag the “if-then” block from the Logic accordion into the loop
  - Drag the “__==__” block from the Logic accordion to the end of the “if-then”
  - Drag the “--Select—” block from the Variables accordion to the first empty space in the “__==__” block and toggle to counter
  - Drag a number from the Math accordion to the second empty space and enter 10
If-Then with the Logic Operator “==” (cont.)

- Add “counter = counter + 1” after the “if-then” statement (see example to the right):
  - Drag the “--Select-- = 0” block from the Variables accordion into the loop after the “if-then” block.
  - Toggle to counter on the left side
  - Replace the number with the “__ +__” block from the Math accordion
  - Place a “--Select--" block from the Variable accordion in the first empty space and toggle to counter
  - Place a number (1) in the second space from the Math accordion
- Next add a debug statement, which prints out messages.
  - Find the “DEBUG” block in the Debug accordion
Adding Debug Statements

- Drag and Drop the “DEBUG” block into the “if-then” block.
- Type in the following message into between the quotation marks in the “DEBUG” block:
  
  “counter == 10”

Notes:
- You will get errors without the quotation marks.
- There are no spaces between the equals signs.

- You have created the following “if-then” statement:
  
  If counter == 10 then
  print the debug statement:
  “counter==10”
Viewing Debug Statements

- Since you have not used any SPHERES controls in your program, the Satellite will not move, instead, watch for your a debug message as follows:
  - Compile, Simulate
    - Maximum time: 45 seconds
  - **Before** you play your simulation
    - Be sure a grey console box is present on your screen (Toggle the “show console” icon, if not)
  - Watch for your debug message in this box when you run the simulation
    - Your debug message for blue SPHERES 1 (SPH1) will appear after 10 seconds (see example)
  - View simulation
• “!=” is a Logic operator that means “not equal”

• What happens if you add the following if-then statement to your loop?

  If counter != 10 then....

• To do this:
  – Drag a new “if-then” block from the Logic accordion and insert it into the loop after the first “if-then” statement and before the “counter = counter + 1” block as shown.
Adding the logic operator “!=” (cont.)

- Change the new “if-then” block statement to:

  If counter “!=” 10 then...
  (If counter is not equal to 10 then...)
  
  - Drag the “__==__” block from the Logic accordion, and use the dropdown in it to change to !=
  
  - Place a variable block (“--Select--”) from the Variables accordion into the first space and select counter
  
  - Place a number block from the Math accordion into the second space and enter 10
Adding the logic operator "!=" (cont.)

• Drag a “DEBUG” block into the new “if-then” statement
• This time write the message:
  
  “not equal”  
  (don’t forget to use the quotation marks)
• Look at the program you created. What do you expect to happen when you run the simulation? Let’s find out.

• Compile, Simulate
  – Maximum Time: 45 seconds

• Before you play your simulation
  – Click the “view console” icon on the bottom of your screen
• Remember that your program is being read by the SPHERES Control System every second.

• A “not equal” message was printed each second that the counter did not equal 10 (starting from counter==0)

• A “counter == 10” message was printed for each second that the counter was equal to 10
  – Can you find that message?

• DEBUG statements can help you check if your program is running the way you expect it to run.
Compare: Your program - versus - C Code

```
void loop() {
  if (counter == 10) {
    DEBUG("counter == 10");
  } else if (counter != 10) {
    DEBUG("not equal");
  }
  counter = counter + 1;
}
```
More on Debugging

• You can also use the debug statement to print out the value of a variable:

  – To do this use the following format inside the DEBUG block:

    “text text text symbol”, variable

  – The symbol used depends on the data type:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td>%d</td>
</tr>
<tr>
<td>float</td>
<td>%f</td>
</tr>
</tbody>
</table>

• For example to print out the value of the counter:

  – In the DEBUG block write: “counter equals %d”, counter

(Use symbol %d since “counter” is an integer (int))
• Change the words in the 2\textsuperscript{nd} DEBUG statement to:

  “counter equals %d”,counter

  – Pay careful attention to the location of the quotation marks and include the comma!

• Look at the program you created. What do you expect to happen when you run the simulation?

• Let’s find out. Compile, Simulate
  – Maximum Time 45 seconds
• Remember that your program is being read every second.
• A “counter equals (number)” message was printed for each second that the counter did not equal 10 (starting from counter==0)
• A “counter == 10” message was printed for each second that the counter was equal to 10
  – Can you find that message?
• DEBUG statements which print variables are also very helpful for checking if your program is running the way you expect it to run.
• Congratulations!

• You are becoming a pro at conditional statements!

• You have learned two more logic operators “==” and “!=”

• You wrote a program that shows the SPHERES Control System reads your loop once per second.

• You have learned how to use Debug statements!