Goals

In this tutorial you will learn how to use `getMyZRState` in conditional statements.

<table>
<thead>
<tr>
<th>My_ZR_State</th>
<th>Position</th>
<th>Velocity</th>
<th>Pointing vector</th>
<th>Rotation rates</th>
</tr>
</thead>
<tbody>
<tr>
<td>X: 0.0</td>
<td>Y: 0.0</td>
<td>Vx: 0.0</td>
<td>Nx: 0.0</td>
<td>ωx: 0.0</td>
</tr>
<tr>
<td>Y: 0.0</td>
<td>Vy: 0.0</td>
<td>Vy: 0.0</td>
<td>Ny: 0.0</td>
<td>ωy: 0.0</td>
</tr>
<tr>
<td>Z: 0.0</td>
<td>Vz: 0.0</td>
<td>Vz: 0.0</td>
<td>Nz: 0.0</td>
<td>ωz: 0.0</td>
</tr>
</tbody>
</table>
- **getMyZRState** retrieves the following information about the Blue satellite:
  - Position \((x,y,z)\)
  - Velocity \((vx,vy,vz)\)
  - Pointing vector \((nx,ny,nz)\)
  - Rotation rates \((\omega_x, \omega_y, \omega_z)\)

- These same values are displayed in upper right corner of the simulation window.

- The ZRState information is provided in an array of 12 floats: \([0] [1] [2] [3] [4] [5] [6] [7] [8] [9] [10] [11]\)
  (Remember, the counting starts from 0; you see 0-11, not 1-12)

- **getMyZRState** \([0], [1], [2]\) represent the \(x\), \(y\), and \(z\) coordinates of the SPHERES.
Use of getMyZRState

• You can use getMyZRState to figure out where your SPHERES satellite is relative to a specific location in the game arena.

• This means you can use ZR State information instead of a counter to decide when things happen—this is very useful in the game!

• In this tutorial you will use getMyZRState information to program the following:
  
  If the satellite has not reached positionA, then continue to positionA
  else go to positionB
Use of getMyZRState (cont.)

- First some things to consider in the example to the right:
  - Q: How does the x coordinate of the satellite change as it moves from initial position (0,0.5,0) to position A (1,0,0) in the picture?
  - A: The satellite starts with x coordinate=0 and moves towards x coordinate=1
- For this example, we will use the SPHERES x-coordinate information to decide if the satellite has reached position A.
- Since: \((\text{Initial position x-coordinate}) < (\text{Position A x-coordinate})\), we can compare the moving SPHERES x coordinate with position A’s x coordinate as follows:
  - If myZRState[0] < positionA [0], then keep moving toward position A
Use of getMyZRState (cont.)

- Because the SPHERES controller is not perfectly accurate, it is best to pick a target x coordinate that comes just before the point the satellite is moving toward (just before position A)
  - Example: x=0.97 is close to x=1.0
  - Pick target x= 0.97
  - This gives you .03 meters (3cm) margin for error

- Program outline:
  - If myZRState[0] < 0.97
  - Then go to position A (x = 1.0)
  - Else go to position B

- Let’s get started:
  - Create a new project
  - Name it “Project9” and choose “FreeMode” and “TextEditor”
• Create the following variables and arrays: (see tutorial on variables and arrays for help)
  – float positionA[3]
    • Set initial value to 1,0,0
  – float positionB[3]
    • Set initial value to 1,1,0
  – float myZRState[12]
    • Leave initial value blank
  – float target[3]
    • Leave initial value blank
• In void loop (), call GetMyZRState, and write the information to the array myZRState.
  The myZRState information will change as the satellite moves. This information will be updated each
time the loop is called, as shown.
• Assign a value to target[0]
  – Set target[0] = 0.97
Else if using myZRState

- Create an if-then statement.
- Use the condition that if the myZRState[0] is less than target[0], the code in the if statement will be executed.
- You will get the following statement:
  
  ```
  void loop(){
    //This function is called once per second.
    //Use it to control the satellite.
    api.getMyZRState(myZRState);
    target[0] = 0.97;
    if (myZRState[0] < target[0]){
      }
    else{
      }
  }
  ```

If-Then-Else using myZRState (cont.)

- Complete the conditional statement If
  myZRstate [0] < target [0] then
    setPositionTarget to positionA
  else
    setPositionTarget to positionB

- Compile and simulate
  - Load settings: Tutorial _90
  - View simulation

Your program

```c
void loop(){
    //This function is called once per second.
    //Use it to control the satellite.
    api.getMyZRState(myZRState);
    target[0] = 0.97;
    if (myZRState[0] < target[0]){
        api.setPositionTarget(positionA);
    } else{
        api.setPositionTarget(positionB);
    }
}
```

Blue satellite should move from:
initial position ➔ positionA ➔ positionB
without pausing
• If your SPHERES did not behave as expected:
  - Troubleshooting
    • Carefully check that your program matches the one shown to the right
    • Check that you have correctly initialized your variables
  - Make any corrections and simulate again

```c
void loop(){
  //This function is called once per second.
  //Use it to control the satellite.
  api.getMyZRState(myZRState);
  target[0] = 0.97;
  if (myZRState[0] < target[0]){
    api.setPositionTarget(positionA);
  } else{
    api.setPositionTarget(positionB);
  }
}
```
Review

Congratulations! You have learned how to use `getMyZRState` in conditional statements in your programs!

<table>
<thead>
<tr>
<th>Position</th>
<th>Velocity</th>
<th>Pointing vector</th>
<th>Rotation rates</th>
</tr>
</thead>
<tbody>
<tr>
<td>X: 0.0</td>
<td>Y: 0.0</td>
<td>Z: 0.0</td>
<td></td>
</tr>
<tr>
<td>Vx: 0.0</td>
<td>Vy: 0.0</td>
<td>Vz: 0.0</td>
<td></td>
</tr>
<tr>
<td>Nx: 0.0</td>
<td>Ny: 0.0</td>
<td>Nz: 0.0</td>
<td></td>
</tr>
<tr>
<td>ωx: 0.0</td>
<td>ωy: 0.0</td>
<td>ωz: 0.0</td>
<td></td>
</tr>
</tbody>
</table>