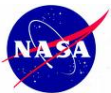


ZERO ROBOTICS

ISS PROGRAMING CHALLENGE

Applied Conditionals



Goals



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

| | My_ZR_State | | |
|-----------------|------------------|------------------|------------------|
| Position | X: 0.0 | Y: 0.0 | Z: 0.0 |
| Velocity | Vx: 0.0 | Vy: 0.0 | Vz: 0.0 |
| Pointing vector | Nx: 0.0 | Ny: 0.0 | Nz: 0.0 |
| Rotation rates | ω_x : 0.0 | ω_y : 0.0 | ω_z : 0.0 |

getMyZRState Review



- **getMyZRState** retrieves the following information about the Blue satellite

| | |
|-----------------|--|
| Position | (x,y,z) |
| Velocity | (vx,vy,vz) |
| Pointing vector | (nx,ny,nz) |
| Rotation rates | (ω_x , ω_y , ω_z) |

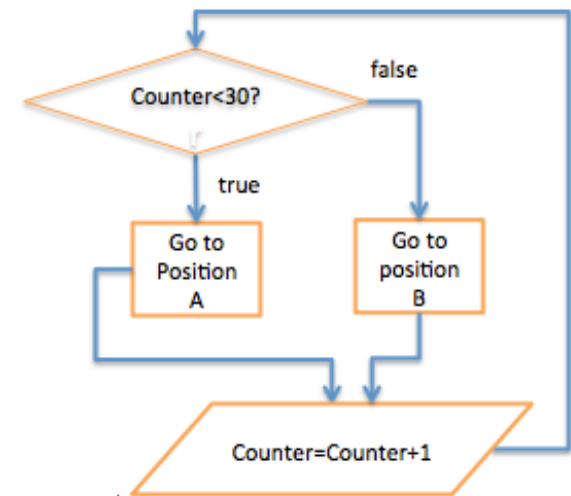
| | | |
|----------------------|------------------|------------------|
| X: 0.0 | Y: 0.0 | Z: 0.0 |
| Vx: 0.0 | Vy: 0.0 | Vz: 0.0 |
| Nx: 0.0 | Ny: 0.0 | Nz: 0.0 |
| ω_x : 0.0 | ω_y : 0.0 | ω_z : 0.0 |
| Fuel Remaining: 100% | | |

- 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] (Remember, the counting starts from 0;
[6] [7] [8] you see 0-11, not 1-12)
[9] [10] [11]
- *getMyZRState* [0], [1], [2] represent the x, y, and z coordinates of the SPHERES

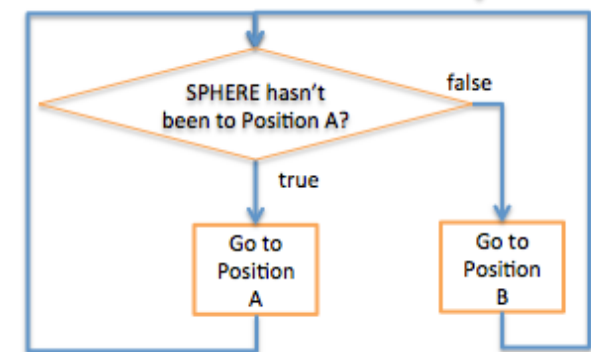
Use of getMyZRState



- 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



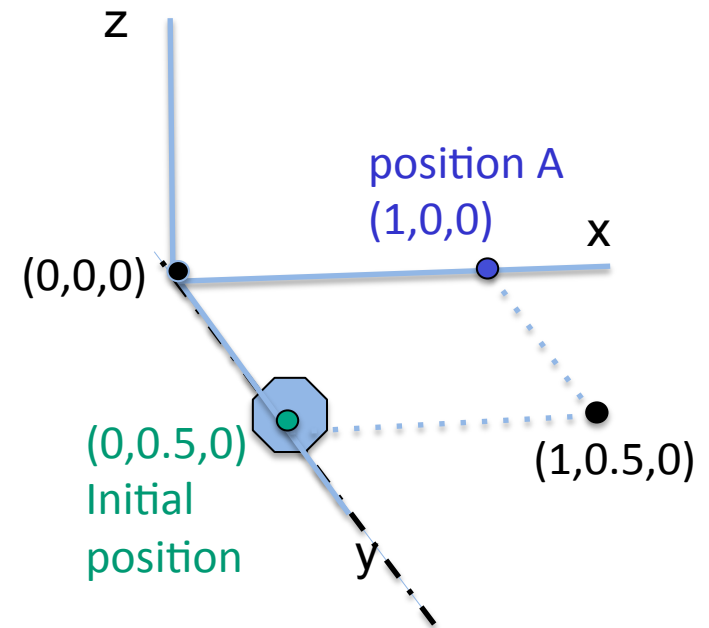
Counter ----versus--- No counter



Use of getMyZRState (cont.)



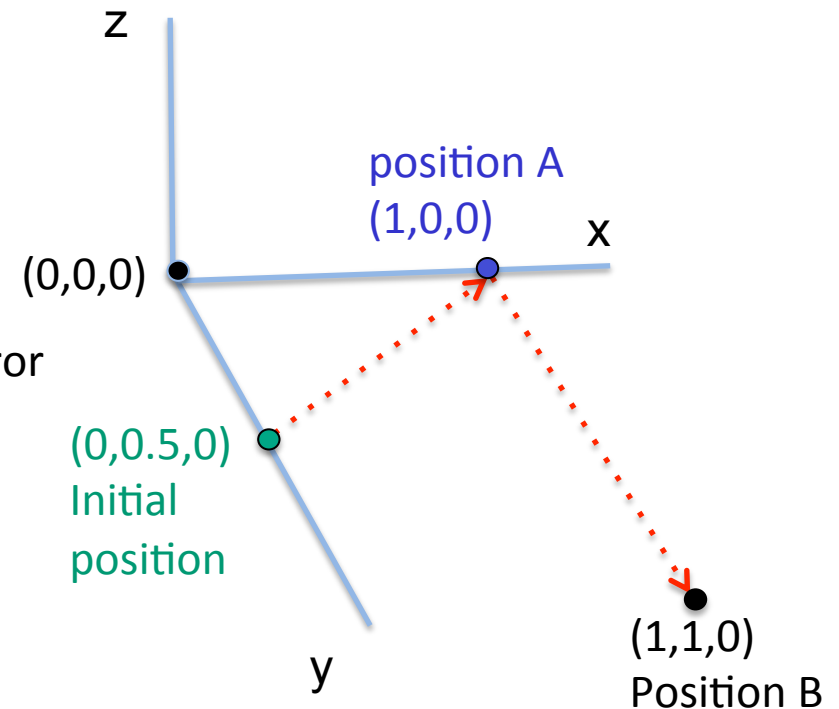
- First some things to consider in the example to the right:
 - Q: How does the x coordinate of the satellite change as it moves from initial position (0,0.5,0) to position A (1,0,0) in the picture?
 - A: The satellite starts with x coordinate=0 and moves towards x coordinate=1
- For this example, we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



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"



Declare variables and arrays



- 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

```
void loop(){
  //This function is called once per second.
  //Use it to control the satellite.

  api.getMyZRState(myZRState);

  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:

If myZRState[0] < target[0]
then...

```
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

```
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-Then-Else using myZRState (cont.)



- 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

```
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);
  }
}
```



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

Position
Velocity
Pointing vector
Rotation rates

| My_ZR_State | | |
|------------------|------------------|------------------|
| X: 0.0 | Y: 0.0 | Z: 0.0 |
| Vx: 0.0 | Vy: 0.0 | Vz: 0.0 |
| Nx: 0.0 | Ny: 0.0 | Nz: 0.0 |
| ωx : 0.0 | ωy : 0.0 | ωz : 0.0 |