1) Setting up the scene

- Use any ground cover, I use desert
- Drag in these objects:
  - Props: castleGate
  - Biped: put in three pandas
  - Drop in an Object marker right in the doorway
- Put the gate in the back with the doors open.
- Put the pandas all in front of it at different distances away (so you can clearly see who is closest to the gate, next closest, etc.)
- See next slide on where to place them

Placement of objects

- Here is a possible setup

The story

- The pandas all jump up and down one at a time
- The pandas all say hello together, and then all together they do a flip.
- The panda closest to the center of the doorway, walks over to the door and disappears
  - This repeats in the order of pandas closest to the doorway, until all the pandas have walked over to the doorway and walked through it and disappeared (You need to calculate “closest”
2) Create an array of type Biped and put the three pandas in it

• Create a variable of type Biped Array in myFirstMethod

```cpp
Biped[3][1] this.panda = new Biped[3] { this.panda1, this.panda2, this.panda3 }
```

• Your program should still work if later we add in a few more pandas into the array.

3) Write a **Biped** Procedure to do a flip

• The procedure should be written as a Biped procedure

• The panda should move up, flip and then move back down.

• THERE SHOULD BE NO LOOP IN THIS PROCEDURE, just flip once.

4) Teach a panda how to take two steps and move forward a little

• Write a **BIPED Procedure** to take two steps. **Both legs should move twice** and the Panda should **move forward** a small amount 0.25

• Be sure to make the walk happen fast, set timings to moving hips to be 0.1

• THERE SHOULD BE NO LOOP IN THIS PROCEDURE

5) Panda movement – Put in myFirstMethod

• Put the following in myFirstMethod
  – Have each panda one at a time jump up and down.
  – Have the pandas all together say “hello”
  – Have the pandas all together do a flip

  – Use the loop structures that access arrays!
    • For each in …, for all together, …
6) Write a Scene Function to find the panda closest to the gate

- Since the gate is so large, instead it is better to put an object marker (that is tiny) and compare distances with it.
- This function should create a variable to keep track of the panda closest to the gate so far. Start with it equal to just any panda.
- Then compare each panda in the array to this variable to see if any panda is closer.

6) Write a Scene Function to find the panda closest to the gate (part 2)

- Once you have calculated the panda that is closest to the gate, then return it.
- Be sure you compare to every panda before returning.

7) Add more code to myFirstMethod

- Use a count loop three times (since you have three pandas) and repeatedly do:
  - Have the panda closest to the gate turn to face it, say “I’m the closest, my turn”, and then have it walk (use your walk function in a loop) over to the gate and then through the gate. As it walks through the gate have it disappear.
  - ONCE it has disappeared, have it move really far away (100?) so it won’t be chosen next as the panda closest to the gate.
8) Now add in two more pandas to the array, and then click Run!

- Modify the array variable to add two more pandas, the program should just work!