Checking for Collisions: Alternative Method

Motivation

• In the Simple Collision Detection tutorial we used the center of buildings to determine when the humvee was going to collide with them
• This method worked, but was ineffective for preventing collisions with corners of buildings
• This method was particularly ineffective for the townhouse, whose center was up high
• In this tutorial, we’ll try out a different way to check for collisions

Getting Set Up

• Open up a new Alice world, choose the sand layout
• Add in the townHouse object from the buildings folder and the Humvee object from the vehicles folder

Adding in Cubes

• Instead of using the center of the building to check if our humvee is about to collide with it, we’re going to place a series of cubes around the perimeter and check if the humvee is about to hit any of those
• So, go into the shapes folder and begin to add in cubes around the perimeter of the building
• The number of cubes you need will depend on whether you resized your building and how you place them.
Adding in Cubes

• It might be difficult to correctly place the cubes.
• You can switch from single view to quad view for more precise placement.
• Start with the corners, placing the cube partly inside of the building.

Adding in Cubes

• Go ahead and add in cubes all around the side, your top view should look something like this.
• Don’t forget to put a cube in the stairs!

Adding Cubes

• Once you’ve added all of your cubes, you need to do two things:
  – You don’t want to see the cubes when you run the world, so go to their properties > is showing and set it equal to false.
  – If you move the townhouse then you want the cubes to stay in place and move with it, so go to the cube’s properties and set the vehicle to the townhouse.
• Do this for every cube in your world.

Creating a list

• Next go to world > properties > create new variable.
• Now create a list and add all of your cubes to it so we can use it in our code.
Writing world.collision
• Go to the world > methods tab and create a new method and call it “collision”
• We will use this method to check if the humvee is about to collide with any of the cubes and if it is, then we will move the humvee back

Writing world.collision
• First drag up an if/else block and select true
• Then open up the humvee’s functions and find **humvee is within threshold of object**
• Drag this on top of the true, you can choose 5 for the threshold and a cube for the object

Writing world.collision
• 5 meters was picked arbitrarily, so we want a more precise measurement
• Locate the humvee’s width in the **humvee > function** tab and drag it over the 5 meters
• Then click the arrow beside the width, go to math, and then divide the width by 2

Writing world.collision
• Now click the drop down arrow in front of the word “of” and choose math and add and then just pick 1
Writing world.collision

- We want to replace the 1 with the cube’s width divided by 2
- First go to cube > function and drag in cube’s width
- Then click the arrow beside the width and find math, and then divide by 2. Your finished statement should look like this:

![Image of cube and math operation]

Writing world.collision

- Now after the if statement we want the humvee to move back, so drag in a humvee > move > backwards > 5 meters
- Set the duration equal to .1 so this happens quickly

![Image of humvee and duration settings]

Writing world.collision

- Right now this will only work with one cube, but we want it to work with every cube so we will use the list we created
- First drag in a for all together and choose for it to go through your cube list

![Image of for loop and cube list]

Writing world.collision

- Drag in the entire if statement into the for all together block
- Replace every instance of cube with item_from_cubes

![Image of for loop and item replacement]
Finished world.collision

- Your finished method should look like this:

Creating an Event

- In your event space, right click on the event “when the world starts”
- Choose change to > while the world is running

Creating an Event

- Change the <None> beside the during to call the world.collide method that we just wrote

Adding a Final Event

- We want to be able to move our Humvee around the world using the arrow keys.
- In your event space, select Create new event > Let the arrow keys move subject
- Select the humvee object when the event appears
Finishing world

• We also want the camera to follow the humvee, so go to camera > properties > vehicle and choose humvee > the entire humvee

• For this to work correctly make sure the humvee is facing away from the camera towards the townhouse, you may need to reposition your objects

Finished world

• Now you can play your world and see how the humvee jumps back from the townhouse

• This method works better than using an object’s center, but is more tedious and requires many more objects