General Instructions: Put the entire app directory into one zip file and submit as an attachment under Content → Homework 12 for this course on the Blackboard Learn system by the above deadline. Note that you may submit multiple times, but only the most recent entry submitted before the above deadline will be graded.

For this homework you will create a SpriteKit game for bagging up your food items. You can find a video of my version of the game at https://eecs.wsu.edu/~holder/courses/MAD/hw12/FoodGame.mp4. The goal is to catch the falling food items in a grocery bag. You get a point for each item caught. After three misses the game is over. See screenshots below. Specifically,

1. Create a new project of type Game and choose SpriteKit for the Game Technology.
2. You may assume the device will remain in Portrait orientation, but if you want to enforce that, you can modify the “shouldAutorotate” and “supportedInterfaceOrientations” variables in GameViewController.swift.
3. In the scene editor (accessed by clicking on GameScene.sks) add the following five labels.
   a. Your current “Score” in the upper left corner.
   b. The current number of “Misses Left” in the upper right corner.
   c. A “GAME OVER” label, centered on the scene, that appears when the game is over.
   d. A “Your Score” label, centered under the “GAME OVER” label, that appears when the game is over showing your final score.
   e. A “Start” label, centered at the bottom, that appears initially and when the game is over. This label will also serve as a button to start the game.
4. Add at least five different food images to your project. Each image should have size 128x128.
5. Add a grocery bag image to your project. This image should also have a width of 128 and a height around 90. Create a SKSpriteNode based on this image and give it a rectangular physics body. This sprite should be positioned at the bottom of the scene, initially with x = 0. The grocery bag will move left and right based on the touchesMoved method (see below).
6. Initially, the game scene shows the “Score”, “Misses Left” and “Start” labels. When “Start” is tapped, the game starts by hiding the “Start” label, and if shown the “GAME OVER” and “Your Score” labels. Starting the game should also unhide the grocery bag centered at the bottom of the scene.
7. Starting the game should also initiate a Timer that fires every second. For each second, the game should randomly select one of the food images, create an SKSpriteNode for it, add a rectangular physics body to it, and position it in a random location somewhere inside the upper half of the scene, i.e., somewhere in the rectangle defined by (-sceneWidth/2 < x < sceneWidth/2) and (0 < y < sceneHeight/2). Once the sprite is added as a child, gravity will move it downward. Also, be sure to set the physic body’s contactTestBitMask = 1.

8. Add the SKPhysicsContactDelegate to your GameScene and implement the didBegin:contact method. This method should check if the two contact nodes are a “Food” and a “GroceryBag”, and if so, add a point to the score, and remove the “Food” node from its parent. If the two contact nodes are a “Food” and the “Wall”, then the misses left should be decremented, and the “Food” node should be removed from its parent. If misses left equals zero, then the game should be stopped; any food items still on the scene should be removed; and the “GAME OVER”, “Your Score” and “Start” labels should be unhidden.

9. Implement touchesBegan so that if the node touched is the “Start” label, then the game should be reset and started. If the node touched is the “GroceryBag” sprite, then the grocery bags x position should be changed to the x position of the touch.

10. Implement touchesMoved so that the “GroceryBag” sprite’s x position is changed to the x position of the touch. This way, the grocery bag should move with your (or the mouse’s) touch.

11. Your app should run in Xcode 11 using Swift 5 and iOS 13. Your app will be tested using Xcode’s iPhone 8 device simulator, so make sure your labels and sprite positions work for that device.
StoryBoard:
Simulator:

Initial View:  

![Initial View Image]

Game Over:  

![Game Over Image]

During Game:  

![During Game Image]

During Game:  

![During Game Image]