Introduction

This week, we’re adding sprites and collision detection to our game.

Design Check

- What behaviors/components are you adding this week to handle sprites and collisions?
- How are you planning on loading and drawing sprites?
- How will you decide which collision method to use?

Primary Requirements

Primary Engine Requirements

- Your handin must meet all global requirements.
- Your handin only crashes under exceptional circumstances (edge cases).
- Your engine must support collision detection between points, circles, and AABs. This includes a collision system and collision behaviors.

Primary Game Requirements

- Your game should display at least 2 “units” in your viewport.
- The player should be able to move the “units” around the screen.
- The “units” should have sprites.
- You must have at least 2 base “units”/elements, which can be combined to form a new element.
- You must be able to add elements to the work-space by dragging them from the menu.
- Your game must complete the debugger. Using the debugger, the TAs should quickly be able to verify collisions between any pair of the following: points, circles, and AABs.
Secondary Requirements

Secondary Engine Requirements
- Your engine must meet all primary engine requirements.
- Your engine must include a sprite component.
-Sprites should only be loaded once.

Secondary Game Requirements
- Your game must meet all primary game requirements.
- You must be able to remove elements from the work-space by dragging them to the menu or some sort of “trash” area.
- You must have at least 4 base “units”/elements and at least 3 “units”/elements the player can make through combinations, one of which must be made by combining two non-base “units”/elements, and one of which must be a final “unit”/element (cannot be combined with anything).

Suggested Extras
- Add labels (UI elements!) to “units”/elements.
- Store newly created “units”/elements in the menu.

Handing In
Handing in works the same way as it did last week! Feel free to refer to the bottom of the Tic assignment handout for a walkthrough of each of these steps. To hand in...
1. Push your final handin commit
2. Create a release for this handin
   a. You should have separate releases for Alc I and Alc II!

Don’t forget to upload a demo video to the demos slack channel!