Nin I

Design Check: Oct 22-23
Due: Oct 27, 11:59 pm EST on GitHub

Introduction

You may have heard of the popular platformer game N; if you haven’t you can play it here. In this assignment, you will be making a simple platformer that implements a subset of the features of N. You’ll add several important features to your engine, including some simple physics, raycasting, and a system of loading levels from files.

Design Check

- How will you expand your existing collision system to allow for physically correct responses?
- How will gravity be handled?
- How will you ensure that the player can only jump when standing on a platform?

Primary Requirements

Primary Engine Requirements

- Your handin must meet all global requirements.
- Your handin only crashes under exceptional circumstances (edge cases)
- Your engine must contain a physics behavior that correctly holds and updates mass, force, impulse, velocity, and acceleration.
- Your engine must handle static objects and restitution.

Primary Game Requirements

- Your game must contain a player-controller unit above an immobile platform.
- The player-controller unit must fall when in the air.
- The player-controller unit must not fall through the platform.
- The player-controller unit must have a constant downward acceleration.
- The player-controller unit must be able to jump, but only when standing on top of a platform.
● Your game must have three objects with visibly different restitution values that can collide with each other.
● Your game never crashes.

Secondary Requirements

Secondary Engine Requirements
● Your engine must meet all primary engine requirements.
● Your engine must support collisions/MTVs with convex polygons.

Secondary Game Requirements
● Your game must meet all primary game requirements.
● The TA debugger must be extended to show collisions and MTVs with convex polygons.
● Your game must include a convex polygon.

Suggested Extras
● Think about fun mechanics you can add to your platformer – allow the player to double jump, slide down walls, jump off of walls, etc.
● Add friction (resistance to motion perpendicular to the MTV).

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

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